

Минобрнауки России  
Юго-Западный государственный университет

Кафедра программной инженерии

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
**ПО ПРОГРАММЕ БАКАЛАВРИАТА**

09.03.04 Программная инженерия

(код, наименование ОПОП ВО: направление подготовки, направленность (профиль))

«Разработка программно-информационных систем»

Интернет-магазин декоративных растений

(название темы)

Дипломный проект

(вид ВКР: дипломная работа или дипломный проект)

Автор ВКР

И. Р. Головина

(подпись, дата)

(инициалы, фамилия)

Группа ПО-026

Руководитель ВКР

Е. И. Аникина

(подпись, дата)

(инициалы, фамилия)

Нормоконтроль

А. А. Чаплыгин

(подпись, дата)

(инициалы, фамилия)

ВКР допущена к защите:

Заведующий кафедрой

А. В. Малышев

(подпись, дата)

(инициалы, фамилия)

Курск 2024 г.

**Минобрнауки России**  
**Юго-Западный государственный университет**

Кафедра программной инженерии

УТВЕРЖДАЮ:  
Заведующий кафедрой

(подпись, инициалы, фамилия)

«\_\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

**ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ  
РАБОТУ ПО ПРОГРАММЕ БАКАЛАВРИАТА**

Студента Головиной И.Р., шифр 20-06-0175, группа ПО-02б

1. Тема « Интернет-магазин декоративных растений» утверждена приказом ректора ЮЗГУ от «04» апреля 2024 г. № 1616-с.
2. Срок предоставления работы к защите «11» июня 2024 г.
3. Исходные данные для создания программной системы:

3.1. Перечень решаемых задач:

- 1) провести анализ предметной области;
- 2) разработать концептуальную модель web-сайта;
- 3) спроектировать web-сайт;
- 4) реализовать сайт средствами web-технологий.

3.2. Входные данные и требуемые результаты для программы:

- 1) Входными данными для программной системы являются: информация о пользователе, предоставляемая им в процессе регистрации в системе; информация о пользователе, предоставляемая им в процессе оформления заказа; информация о товарных позициях (отдельных товарах) заказа; параметры фильтрации товаров; информация о товарах, вводимая администратором; информация о категориях, вводимая администратором.

2) Выходными данными для программной системы являются: стоимость всех товаров в корзине; список заказов; уведомление о добавлении товара в корзину; уведомление об удалении товара из корзины; уведомление об оформлении заказа; сообщения об ошибках; ссылки для переадресации (URL).

4. Содержание работы (по разделам):

4.1. Введение

4.1. Анализ предметной области

4.2. Техническое задание: основание для разработки, назначение разработки, требования к программной системе, требования к оформлению документации.

4.3. Технический проект: общие сведения о программной системе, проект данных программной системы, проектирование архитектуры программной системы, проектирование пользовательского интерфейса программной системы.

4.4. Рабочий проект: спецификация классов программной системы, тестирование программной системы.

4.5. Заключение

4.6. Список использованных источников

5. Перечень графического материала:

Лист 1. Сведения о ВКРБ

Лист 2. Цель и задачи разработки

Лист 3. Диаграмма прецедентов. Часть 1

Лист 4. Диаграмма прецедентов. Часть 2

Лист 5. Диаграмма компонентов

Лист 6. Заключение

Руководитель ВКР

\_\_\_\_\_

(подпись, дата)

Е. И. Аникина

\_\_\_\_\_

(инициалы, фамилия)

Задание принял к исполнению

\_\_\_\_\_

(подпись, дата)

И. Р. Головина

\_\_\_\_\_

(инициалы, фамилия)

## РЕФЕРАТ

Объем работы равен 92 страницам. Работа содержит 32 иллюстрации, 25 таблиц, 54 библиографических источников и 6 листов графического материала. Количество приложений – 2. Графический материал представлен в приложении А. Фрагменты исходного кода представлены в приложении Б.

Перечень ключевых слов: интернет-магазин, растения, база данных, пользователь, администратор, покупатель, товар, web-сайт.

Объектом разработки является интернет-магазин декоративных растений.

Целью выпускной квалификационной работы является создание сайта с основными функциями интернет-магазина и представлением характеристик растений в удобной для пользователя форме.

В процессе создания сайта были выделены основные характеристики растений, на основе которых была построена база данных. Использованы классы и методы модулей, обеспечивающие работу с сущностями предметной области, а также корректную работу интернет-магазина.

При разработке сайта использовался фреймворк Laravel.

Программная система может быть использована для перевода питомника или магазина декоративных растений в онлайн-формат с целью привлечь аудиторию из других городов.

## ABSTRACT

The volume of work is 92 pages. The work contains 32 illustrations, 25 tables, 54 bibliographic sources and 6 sheets of graphic material. The number of applications is 2. The graphic material is presented in annex A. The layout of the site, including the connection of components, is presented in annex B.

List of keywords: online shop, plants, database, user, administrator, buyer, goods, web site.

The object of the development is an online shop of decorative plants.

The purpose of the final qualification work is to create a website with the main functions of an online shop and the presentation of plant characteristics in a user-friendly form.

In the process of creating the site, the main characteristics of plants were highlighted, on the basis of which the database was built. Classes and methods of modules were used to ensure work with the entities of the subject area, as well as the correct operation of the online shop.

The Laravel framework was used in the development of the site.

The software system can be used to convert a nursery or an decorative plant store into an online format in order to attract an audience from other cities.

## СОДЕРЖАНИЕ

<b>ВВЕДЕНИЕ</b>	<b>10</b>
1 Анализ предметной области	12
1.1 История интернет-магазинов	12
1.2 Характеристики растений	13
1.2.1 Многолетние растения	15
1.2.2 Двулетние растения	17
1.2.3 Однолетние растения	17
1.3 Особенности продажи растений	19
1.4 Особенности заказа растений через Интернет	20
2 Техническое задание	23
2.1 Основание для разработки	23
2.2 Цель и назначение разработки	23
2.3 Требования к программной системе	23
2.3.1 Требования к данным	23
2.3.2 Функциональные требования	24
2.3.2.1 Прецедент администратора «Просмотр информации о товаре»	26
2.3.2.2 Прецедент администратора «Редактирование информации о товаре»	26
2.3.2.3 Прецедент администратора «Удаление товара»	26
2.3.2.4 Прецедент администратора «Добавление товара»	26
2.3.2.5 Прецедент администратора «Просмотр информации о заказе»	27
2.3.2.6 Прецедент администратора «Просмотр информации о категории»	27
2.3.2.7 Прецедент администратора «Редактирование информации о категории»	27
2.3.2.8 Прецедент администратора «Удаление категории»	27
2.3.2.9 Прецедент администратора «Добавление категории»	28
2.3.2.10 Прецедент покупателя «Просмотр информации о товаре»	28
2.3.2.11 Прецедент покупателя «Добавление товара в корзину»	28

2.3.2.12	Прецедент покупателя «Просмотр товаров категории»	28
2.3.2.13	Прецедент покупателя «Просмотр информации о заказе»	29
2.3.2.14	Прецедент покупателя «Изменение количества товара»	29
2.3.2.15	Прецедент покупателя «Оформление заказа»	29
2.3.3	Требования пользователя к интерфейсу web-сайта	29
2.3.4	Нефункциональные требования	30
2.3.4.1	Требования к безопасности	30
2.3.4.2	Требования к ПО	31
2.3.4.3	Требования к аппаратному обеспечению	31
2.4	Требования к оформлению документации	32
3	Технический проект	33
3.1	Компоненты приложения	33
3.2	Обоснование выбора технологии проектирования	33
3.2.1	PHP	33
3.2.2	MySQL	34
3.2.3	Blade	35
3.3	Архитектура Laravel	36
3.3.1	Паттерн MVC	36
3.3.2	Жизненный цикл запроса	36
3.3.3	Сервис-провайдеры	38
3.3.4	Сервис-контейнеры	38
3.3.5	Фасады	39
3.4	Структура каталогов	39
3.5	Структура базы данных	40
3.6	Список контроллеров	44
3.6.1	Контроллеры подкаталога Admin	45
3.6.2	Контроллеры подкаталога Auth	45
3.7	Список представлений	45
3.7.1	Представления подкаталога frontend	46
3.7.2	Представления подкаталога orders	46
3.7.3	Представления подкаталога auth	46

3.7.4 Представления подкатаога layouts	46
3.7.5 Представления подкатаога admin	47
3.8 Стили приложения	47
4 Рабочий проект	48
4.1 Маршруты приложения	48
4.2 Спецификация контроллеров приложения	49
4.3 Системное тестирование разработанного web-сайта	53
ЗАКЛЮЧЕНИЕ	68
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	68
ПРИЛОЖЕНИЕ А Представление графического материала	75
ПРИЛОЖЕНИЕ Б Фрагменты исходного кода программы На отдельных листах (CD-RW в прикрепленном конверте)	82
Сведения о ВКРБ (Графический материал / Сведения о ВКРБ.png)	Лист 1
Цель и задачи разработки (Графический материал / Цель и задачи разработки.png)	Лист 2
Диаграмма прецедентов. Часть 1 (Графический материал / Диаграмма прецедентов. Часть 1.png)	Лист 3
Диаграмма прецедентов. Часть 2 (Графический материал / Диаграмма прецедентов. Часть 2.png)	Лист 4
Диаграмма компонентов (Графический материал / Диаграмма компонентов.png)	Лист 5
Заключение (Графический материал / Заключение.png)	Лист 6

## ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

БД – база данных.

ИС – информационная система.

ИТ – информационные технологии.

ПО – программное обеспечение.

РП – рабочий проект.

СУБД – система управления базами данных.

ТЗ – техническое задание.

ТП – технический проект.

UML (Unified Modelling Language) – язык графического описания для объектного моделирования в области разработки программного обеспечения.

## ВВЕДЕНИЕ

Питомник растений – это специально созданный участок, где выращивают и продают различные виды и сорта растений. Они могут быть как крупными промышленными предприятиями, так и небольшими семейными бизнесами. Главная задача питомников — производить качественные растения с хорошо развитой корневой системой, которые могут быстро адаптироваться к новым условиям роста. Растения в питомниках продаются как оптом, так и в розницу.

Из-за больших размеров и требований к окружающей среде, питомники не всегда удобно располагать в местах с большим скоплением людей. Однако они должны быть расположены вблизи населённых пунктов, чтобы обеспечить удобный доступ для покупателей и крупной техники.

Интернет-магазин позволяет привлекать покупателей из разных городов и требует гораздо меньше средств, чем открытие филиалов и привлечение дистрибуторов. Кроме того, покупка через Интернет занимает у покупателя меньше времени, что увеличивает вероятность её совершения. Также есть возможность принимать заказы по ночам, когда физические магазины обычно не работают.

Ещё одним преимуществом создания интернет-магазина является возможность использовать маркетинговые инструменты для продвижения. Благодаря сайту можно реализовать таргетированную рекламу, поисковую оптимизацию (SEO), аналитику и многое другое.

Сайт является лицом компании и может существенно повысить её имидж. Любой пользователь Интернета сможет получить необходимую информацию о компании в любое время. На сайте можно найти контактные телефоны, адрес и электронную почту для связи с компанией. Сейчас большинство клиентов узнают о её существовании именно через сайт. Поэтому сайт можно назвать лучшей рекламой.

*Цель настоящей работы* – разработка web-сайта для продажи растений через Интернет. Для достижения поставленной цели необходимо решить следующие задачи:

- провести анализ предметной области;
- разработать концептуальную модель web-сайта;
- спроектировать web-сайт;
- реализовать сайт средствами web-технологий.

*Структура и объем работы.* Отчет состоит из введения, 4 разделов основной части, заключения, списка использованных источников, 2 приложений. Текст выпускной квалификационной работы равен 11 страницам.

*Во введении* сформулирована цель работы, поставлены задачи разработки, описана структура работы, приведено краткое содержание каждого из разделов.

*В первом разделе* на стадии описания технической характеристики предметной области приводится сбор информации о характеристиках растений и специфике продажи товаров через Интернет.

*В втором разделе* на стадии технического задания приводятся требования к разрабатываемому сайту.

*В третьем разделе* на стадии технического проектирования представлены проектные решения для web-сайта.

*В четвертом разделе* приводится спецификация контроллеров приложения, производится тестирование разработанного сайта.

В заключении излагаются основные результаты работы, полученные в ходе разработки.

В приложении А представлен графический материал. В приложении Б представлены фрагменты исходного кода.

# **1 Анализ предметной области**

## **1.1 История интернет-магазинов**

Прообраз современных интернет-магазинов появился задолго до создания мировой паутины. В 1979 году Михаэль Олдрич, занимавшийся поставками коммуникационных сетей в Великобританию, изобрёл телемагазин «videotex». Эта технология позволяла транслировать каталоги с товарами и контактными данными для связи с продавцом.

Первый интернет-магазин, похожий на современные, появился в 1992 году. Чарльз Стэк создал интернет-магазин книг. В то время книги покупали по бумажным каталогам с ценами и описаниями. Чарльз оцифровал эти каталоги и перенёс их в формат, понятный для интернета. Книги не имели срока годности и не меняли спрос из-за модных тенденций. Многие интернет-магазины, появившиеся в начале 90-х, начинали свою деятельность с продажи печатной продукции.

В 1994 году Джек Бэзос придумал концепцию проекта «Amazon». Изначально в нём можно было купить только книги, но со временем ассортимент сильно расширился. Через интернет стали продавать больше вещей, которые раньше можно было купить только в обычном магазине. Позже «Amazon» стал предлагать постельное бельё, одежду и бытовую технику. Хоть код интернет-ресурса имел изъяны (например, можно было заказать отрицательное количество товара), этот магазин получил всеобщее признание [48].

Первый российский интернет-магазин появился в 1997 году в Москве, он был полностью книжным, как и его предшественники. Через год в стране появились первые интернет-проекты по интернет-банкингу и автоматические интернет-шлюзы для обработки заказов за секунды. Российский интернет-трейдинг сделал резкий рывок вперёд, который определил его дальнейшее развитие.

Появление интернет-магазинов стало настоящей революцией в сфере розничной торговли. Они дали покупателям возможность выбирать из огром-

ного ассортимента товаров, сравнивать цены и делать покупки в любое удобное время.

Кроме того, благодаря интернет-магазинам удалось сократить расходы на аренду торговых площадей и оплату труда персонала. Это позволило снизить цены на товары, делая их более доступными для широкого круга потребителей.

## 1.2 Характеристики растений

Декоративные растения — это разнообразная группа растений, включающая как культивируемые, так и дикие виды деревьев, кустарников, многолетних и однолетних растений, злаков и луковичных растений, которые принадлежат к различным ботаническим семействам. Их уникальные декоративные качества делают их незаменимыми для озеленения садов, парков и скверов, а также для украшения зданий и внутренних помещений.

Внешний облик декоративных растений играет ключевую роль в определении их декоративных характеристик. Важными аспектами являются живописность кроны, изящество силуэта, длительность и обильность цветения, изменчивая окраска в течение года, способность выдерживать неблагоприятные условия и климатические изменения.

Декоративные злаки — это растения семейства злаковых, которые используются для создания групповых посадок и бордюров, газонов, а также для составления букетов. Они обладают привлекательной текстурой и формой, что делает их популярными элементами ландшафтного дизайна [7].

Луковичные растения представляют собой обширную группу декоративных растений, различающихся по цвету, размеру, форме цветков и времени цветения. В открытом грунте они цветут с весны до осени, а благодаря зимней выгонке способны цвести практически круглый год. Эти растения могут использоваться для выращивания в помещении и для создания букетов.

С точки зрения распространения по миру, луковичные растения представляют одну из самых многочисленных групп. Они включают в себя преимущественно лилейные и амариллисовые семейства [44].

Луковичные растения могут размножаться как вегетативным способом, так и семенами. Лилии и гиппеаструмы обычно размножаются семенами, этот метод также используется при создании новых сортов.

Растения, образующие луковицы, требуют особой почвы. Им подходят легкие, проницаемые и влагоемкие грунты с нейтральной или слабошелочной реакцией ( $\text{рН}$  не ниже 6–7). Последнее особенно важно для тюльпанов. Для лилий подходят легкие глинистые почвы с хорошей дренажной системой, практически без извести. Гиацинты и тюльпаны предпочитают песчаные почвы, богатые питательными веществами, с нейтральной кислотностью. Тяжелые глинистые почвы с плохой водопроницаемостью, а также очень бедные песчаные почвы не подходят для луковичных растений [24].

Декоративные деревья и кустарники бывают хвойные и лиственные, вечнозелёные или с опадающей листвой. Их красота зависит от внешних характеристик, таких как форма кроны, цвет и структура листьев, а также цвет и размер цветов и плодов. Подходящие условия для роста помогают сохранить красоту растений, которая меняется вместе с возрастом и сменой времен года в зависимости от генетических особенностей вида.

У древесных растений существуют два способа размножения: семенной и вегетативный. Семенное размножение осуществляется через семена, а вегетативное — через черенки, отводки, корневые отпрыски, прививки или деление куста [10].

При создании групп и игры с контрастными цветами важную роль играет окраска листьев деревьев. Она может усилить общее впечатление от кроны, добавить плотности и объема. В течение сезона меняется интенсивность окраски листьев. Форма и размер листьев также влияют на декоративность растений, особенно если они находятся рядом с дорожками. Растения с красивыми и необычными листьями имеют большую ценность в ландшафтном дизайне, если они не характерны для местной флоры.

В период цветения декоративные растения приобретают особую привлекательность. Они отличаются разнообразием форм, цветов, размеров соцветий, ароматами, продолжительностью и временем цветения. Правильный

выбор цветущих растений поможет создать композицию с длительным цветением.

Жизненный цикл декоративных растений — это онтогенез, или индивидуальное развитие растения от его появления из оплодотворенной яйцеклетки или вегетативной почки (черенка с почками, корневого отпрыска и т.п.) до отмирания.

### **1.2.1 Многолетние растения**

Характерной чертой многолетних травянистых растений является способность переживать зиму на открытой земле, ежегодно проходить через все стадии развития (рост, цветение, плодоношение) и продолжать этот цикл на протяжении многих лет благодаря специальным органам, предназначенным для зимовки (корень, корневище, луковица, ползучие стебли и другие).

У большинства декоративных многолетников надземная часть каждый год отмирает, но на следующий сезон восстанавливается за счет измененного подземного стебля в виде корневищ, луковиц, клубнелуковиц (например, луковичные растения, пионы, флоксы,rudbeckia, акониты и другие) [34].

У других растений надземные побеги не отмирают, а сохраняются зимой, когда все жизненные процессы, как у древесных растений, замедляются. С приходом весны и благоприятных условий из почек начинают развиваться новые вегетативные и генеративные побеги, которые цветут и образуют семена.

У некоторых видов растений стебель и корень, помимо своих основных функций, также выполняют задачи вегетативного размножения и защиты растений зимой.

Существуют три основных типа подземных стеблей: луковица, клубень (клубнелуковица) и корневище.

Луковица — это видоизменённый побег с плотным коротким стеблем и мясистыми листьями, имеющими вид чешуек и запасающими воду и питательные вещества [44].

Клубень – это укороченный, значительно утолщенный подземный побег, который запасает питательные вещества. Клубни, имеющие форму луковицы, называются клубнелуковицами.

Корневище – это долговечное стеблевое образование, внешне напоминающее корень, но отличающееся наличием дополнительных корней и почек. На его поверхности можно видеть следы прикрепления листьев в виде коричневых чешуек и остатки умерших побегов. Корневище может быть горизонтальным, утолщенным (как у ириса, бадана, ландыша), растущим вертикально или наклонно вниз (как у астильбы, примул, функии, диклигры) [45].

У многолетних стержнекорневых растений основной корень не замещается дополнительными, а сохраняется на протяжении всей жизни растения. На его поверхности могут быть небольшие корешки с дополнительными почками (корневые отпрыски), используемые для размножения.

Многолетние растения, благодаря своим биологическим особенностям, способны выдерживать низкие температуры зимой. Молодые экземпляры имеют невысокую степень морозоустойчивости, но по мере их развития эта способность увеличивается. Однако по мере завершения жизненного цикла у растений морозоустойчивость снова снижается. Для сохранения и развития этого ценного биологического свойства растений важна своевременная подготовка к зиме, проведение осенних и ранневесенних посевов в открытый грунт. Не столько низкие температуры, сколько резкие перепады могут нанести вред многолетникам. Особенно опасны для них зимы без снега, с оттепелями и долгими затоплениями поверхности почвы водой. Поэтому для выращивания многолетников, особенно тех, что длительное время находятся на одном месте (например, пионы, флоксы, функии и другие), необходимо выбирать ровные, хорошо дренированные участки без застоя воды.

При создании декоративных миксбордеров (цветников-бордюров из смешанных растений разных размеров и оттенков) важно учитывать морозоустойчивость растений и подбирать ассортимент с учетом конкретных почвенно-климатических условий каждой зоны. Только таким образом мож-

но обеспечить успешное процветание и красоту растений в течение долгого времени [40].

### **1.2.2 Двулетние растения**

В цветоводстве под двулетниками понимаются растения, которые по своей природе являются многолетниками, но достигают пика декоративного эффекта в период своего второго года жизни. К их числу относятся такие виды, как виола, колокольчики, наперстянка, маргаритки и незабудки.

Двулетники обладают разнообразными биологическими и декоративными характеристиками, благодаря чему их применяют для украшения садов и цветников в начале весны, в качестве материала для создания цветочных композиций, для выращивания в горшках, а также для украшения помещений и пристенного озеленения [46].

В отличие от однолетников, двулетники не завершают свой жизненный цикл за один сезон. В первый год они формируют мощную розетку листьев, в то время как во втором году достигают максимального цветущего состояния и дают плоды. В ходе третьего зимнего сезона большая часть растений гибнет, но из семян, оставшихся после увядания, вновь появляются новые растения, которые также достигают пика цветения во втором году своей жизни. Таким образом, участки, занятые двулетниками, способны к самовозобновлению и могут сохранять свою декоративность на протяжении многих лет, со временем изменяя свой облик.

При сильной засухе двулетники могут зацвести и дать небольшое количество семян в первый год жизни.

### **1.2.3 Однолетние растения**

К группе однолетников относятся однолетние цветочные культуры, которые проходят все стадии развития за один сезон. Это астры, ноготки, бархатцы, васильки, алиссумы, маки, флоксы летние и другие растения.

В год посева они цветут, дают семена и погибают. Продолжительность их жизни — всего несколько месяцев в году. Однако по разнообразию форм,

яркости окраски цветков и длительности цветения многие летники превосходят другие цветочные растения. Они обладают исключительным многообразием по всем декоративным особенностям и предоставляют продолжительное время для срезки в букеты — от апреля до самых морозов. Кроме того, летники имеют ещё одно положительное свойство — их можно пересаживать в цветущем состоянии.

К однолетникам причисляют некоторые грунтовые многолетники, которые зацветают в первый же год, но не переносят зиму в наших условиях. К ним относятся львиный зев, петуния, вербена и другие растения, которые на своей родине живут по несколько лет.

По декоративным качествам и применению в озеленении летники можно разделить на цветущие и лиственно-декоративные (орнаментальные), бордюрные и вьющиеся.

Большой видовой и сортовой ассортимент однолетних растений позволяет выбрать множество однолетников, которые можно размножать безрассадным способом — посевом семян в открытый грунт. Этот способ отличается доступностью и дешевизной.

Посев семян летников можно производить осенью — после окончания периода длительных потеплений (вторая половина ноября — начало декабря), весной — до наступления вегетационного периода (вторая половина апреля) или летом. Весенний посев в большинстве случаев даёт лучшие результаты, чем осенний, поэтому надежнее сеять летники весной.

Непосредственно в открытый грунт весной и осенью высевают семена многих однолетних растений, которые достаточно холодостойки. Более теплолюбивые виды следует высевать только после того, как прекратятся сильные весенние заморозки по утрам.

Также в открытый грунт высеваются семена тех растений, которые не переносят пересадку (пикровку) из-за того, что их корневая система не может быстро восстановить всасывающие волоски, которые развиваются только на глубоко уходящих в почву корневых ответвлениях (мак, левкой, резеда, люпин, настурция, эшшольция).

В открытый грунт также высевают семена быстрорастущих видов растений, таких как декоративная фасоль, ноготки, акроклинум, кореопсис и бархатцы.

Для посева однолетних декоративных растений лучше всего использовать крупные отборные семена. Отбор крупных семян имеет огромное преимущество — они содержат большой запас питательных веществ, сеянцы быстро развиваются и образуют сильнорослые устойчивые растения с крупными соцветиями и цветками, которые лучше сопротивляются грибковым заболеваниям и вредителям. Кроме того, в таких случаях наблюдается более раннее цветение [47].

### 1.3 Особенности продажи растений

Продавцы несут ответственность перед покупателями и должны соблюдать определённые правила. Они обязаны предоставить информацию о своём фирменном наименовании, месте нахождения организации и режиме работы. Эта информация должна быть размещена на вывеске магазина.

Если продавец является индивидуальным предпринимателем, он также должен сообщить покупателю о своей государственной регистрации и наименовании органа, который её провёл.

У продавцов должна быть книга отзывов и предложений, которую они обязаны предоставить покупателю по первому требованию. Это помогает улучшить качество обслуживания и создать положительную обратную связь.

Правила продажи различных видов товаров должны быть наглядно и доступно представлены. Это важно для обеспечения прозрачности и доверия со стороны покупателей.

Продавцы обязаны своевременно предоставлять покупателям достоверную информацию о товарах. Эта информация должна содержать наименование товара, сведения о стране происхождения (для импортных товаров), подтверждении соответствия, основных потребительских свойствах товара. Также необходимо указать цену и условия приобретения товара, информа-

цию о разрешении на ввоз определённых видов дикорастущих растений в Российскую Федерацию [49].

Если кассовый чек не содержит видового названия и количества растений, то покупателю должен быть передан товарный чек с этой информацией.

Продавцы обязаны предупредить покупателей о недостатках товаров не только устно, но и письменно. Растения являются непродовольственными товарами, которые не подлежат возврату или обмену на аналогичный товар другого размера, формы, цвета и т. д., если они не имеют недостатков. Перед покупкой растения необходимо тщательно проверить его и убедиться в соответствии вашим требованиям. Некоторые растения могут быть опасными и ядовитыми, особенно для детей и домашних животных. При продаже таких растений продавцы также должны предоставить информацию о возможной опасности, чтобы покупатели были внимательны и предприняли необходимые меры предосторожности.

#### **1.4 Особенности заказа растений через Интернет**

Продажа семян и саженцев по почте — это удобный и популярный способ покупки в России. Больше не нужно обходить множество магазинов в поисках нужных сортов и гибридов: покупатели могут спокойно и внимательно изучить каталог или сайт интернет-магазина, не выходя из дома.

Во многих регионах страны выбор семян и посадочного материала ограничен, а качество и ассортимент уступают тем, что представлены в крупных интернет-магазинах и каталогах компаний. Поэтому жители регионов часто заказывают товары по почте, так как не могут найти нужный сорт или вид растения в обычных магазинах.

Дистанционная торговля семенами и посадочным материалом может осуществляться двумя способами: через каталоги и через интернет-магазины. До недавнего времени основным видом торговли была продажа через каталоги. С развитием интернета в нашей стране, на смену этому виду торговли пришла торговля через интернет-магазины.

Самый популярный и простой товар с точки зрения хранения и доставки — это семена [8]. Их можно заказывать круглый год, многие огородники планируют свои посевы ещё с осени и делают заказы заранее, в октябре-декабре. Такие посылки приходят адресату до Нового года. Однако пик заказов обычно приходится на январь-февраль. Заказывая семена, всегда нужно учитывать сроки посева каждой культуры. Затем отнимается 3-5 дней на комплектацию заказа и 5-20 дней на доставку почтой, в зависимости от региона. В итоге получается, что заказ семян следует делать минимум за месяц до посева [35].

Второй по популярности товар — это луковичные и многолетние растения. Этот вид товара имеет два сезона продаж — весна и осень. Весной предлагаются огромный ассортимент многолетних растений, таких как астильбы, хосты, пионы травянистые, гейхеры, а также луковичных растений, таких как гладиолусы, лилии, амариллисы. Срок приёмки заказов с ноября по апрель, отправка заказов начинается в марте. Осенний сезон более короткий и предлагает тюльпаны, нарциссы, крокусы, гиацинты и различные мелколуковичные растения. Заказы принимаются с июля по август, отправка осуществляется с августа по сентябрь.

Саженцы плодовых и декоративных кустарников и деревьев — следующий по популярности товар. Приём заказов начинается с ноября, а рассылка стартует после 8 марта.

Саженцы кустарников и деревьев, а также луковичные и многолетние растения — это более специфический продукт, чем семена. Для сохранения качества им нужны определённые температурные условия. При выборе оплаты наложенным платежом покупатель оплачивает посылку при получении на почте. Стоимость посылки состоит из нескольких компонентов: стоимости товара, стоимости доставки, которая зависит от веса посылки, расстояния и выбранного способа доставки, а также комиссии за обработку наложенного платежа, которую взимает почтовая служба или курьерская компания.

При выборе предоплаты покупатель заполняет квитанцию в каталоге или получает её на электронную почту. Отправка заказа осуществляется по-

сле поступления денег на расчётный счёт продавца. При получении товара на почте покупатель оплачивает только расходы за почтовую пересылку.

## **2 Техническое задание**

### **2.1 Основание для разработки**

Полное наименование системы: «Интернет-магазин декоративных растений».

Основанием для разработки программы является приказ ректора ЮЗГУ от «04» апреля 2024 г. №1616-с «Об утверждении тем выпускных квалификационных работ».

### **2.2 Цель и назначение разработки**

Программно-информационная система предназначена для продажи растений через Интернет. С системой должны работать следующие группы пользователей:

- администратор интернет-магазина;
- посетитель интернет-магазина.

Администратор должен иметь возможность добавлять, удалять и редактировать товары и свои категории. Например, «акции» или «новинки». Посетитель интернет-магазина должен иметь возможность просматривать товары, добавлять их в корзину и оформлять заказ.

Задачами данной разработки являются:

- проектирование базы данных;
- проектирование пользовательского интерфейса;
- разработка методов отображения данных из базы данных;
- разработка методов управления базой данных администратором.

### **2.3 Требования к программной системе**

#### **2.3.1 Требования к данным**

Входными данными для системы являются:

- информация о пользователе, предоставляемая им в процессе регистрации в системе;

- информация о пользователе, предоставляемая им в процессе оформления заказа;
- информация о товарных позициях (отдельных товарах) заказа;
- параметры фильтрации товаров;
- информация о товарах, вводимая администратором;
- информация о категориях, вводимая администратором.

Выходными данными для системы являются:

- стоимость всех товаров в корзине;
- список заказов;
- уведомление о добавлении товара в корзину;
- уведомление об удалении товара из корзины;
- уведомление об оформленном заказе;
- сообщения об ошибках;
- ссылки для переадресации (URL).

### 2.3.2 Функциональные требования

В процессе разработки сайта была создана модель, наглядно демонстрирующая, как пользователи будут взаимодействовать с сайтом. Для создания диаграммы вариантов использования был использован унифицированный язык визуального моделирования UML. Диаграмма вариантов использования описывает функциональное назначение системы. Это исходное представление системы, которое используется при её проектировании и разработке.

Система представлена в виде набора прецедентов — наборов действий, которые система выполняет для актёров, взаимодействующих с ней. Актёром может быть человек или техническое устройство, которые взаимодействуют с системой извне.

На рисунке 2.1 изображены прецеденты для администратора сайта.

На рисунке 2.2 изображены прецеденты для покупателя.

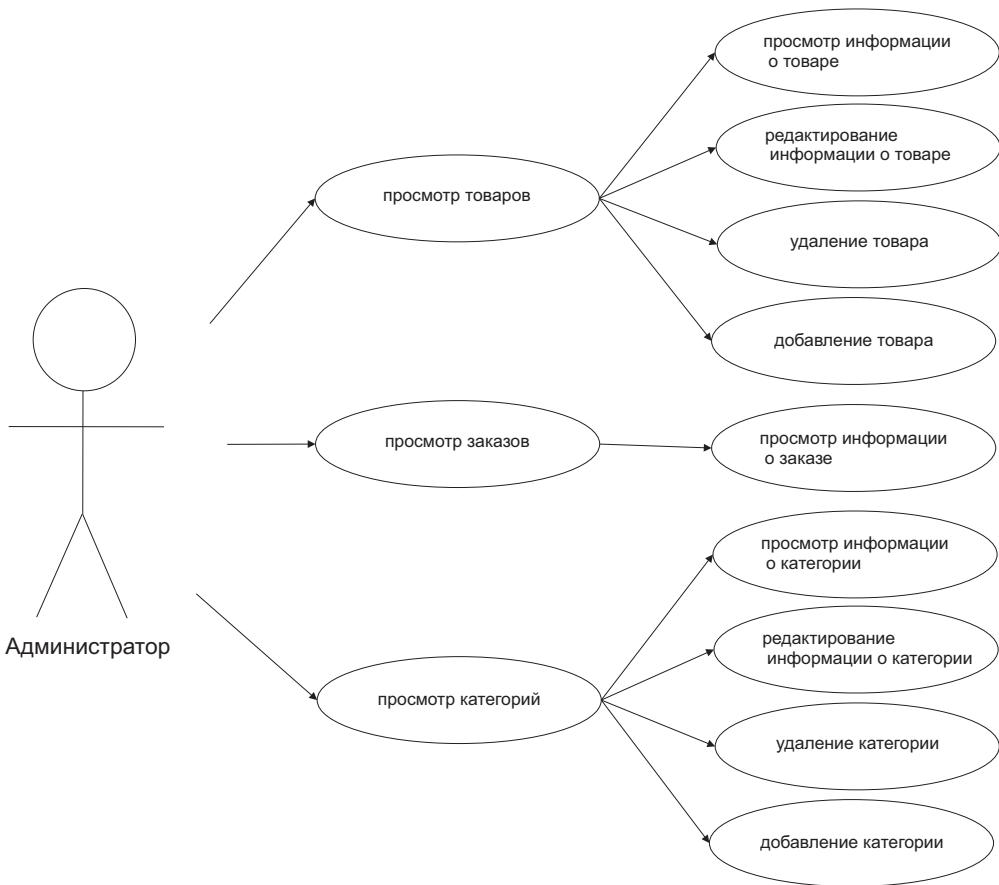


Рисунок 2.1 – Прецеденты для администратора

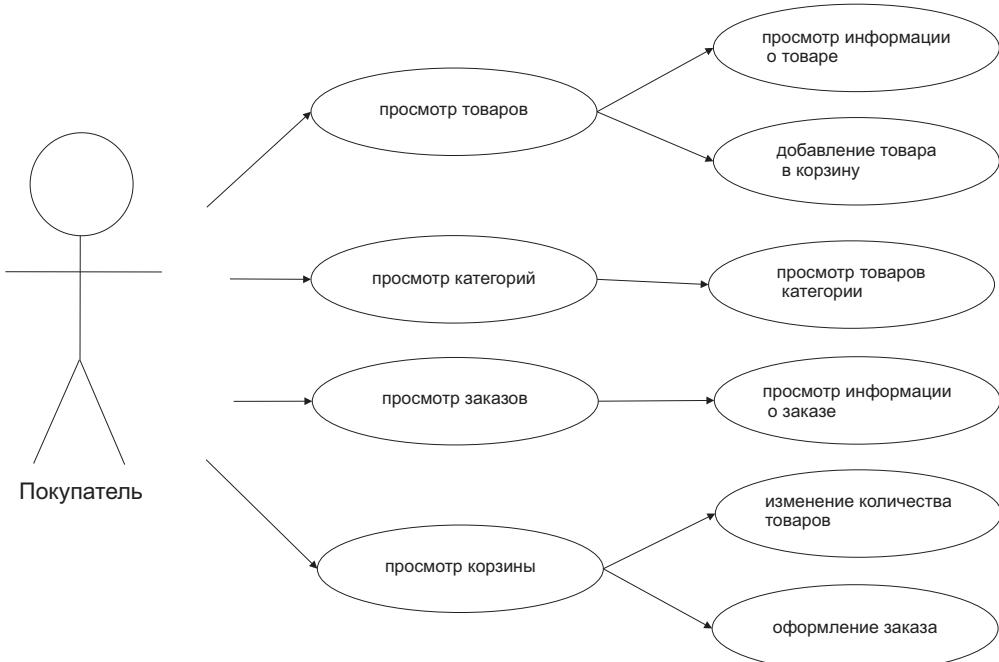


Рисунок 2.2 – Прецеденты для покупателя

### **2.3.2.1   Прецедент администратора «Просмотр информации о товаре»**

Основной успешный сценарий для прецедента «Просмотр информации о товаре»:

1. Открыть страницу товаров.
2. Нажать на кнопку «Открыть» рядом с товаром, о котором нужно узнать информацию.
3. Ознакомиться с представленной информацией.

### **2.3.2.2   Прецедент администратора «Редактирование информации о товаре»**

Основной успешный сценарий для прецедента «Редактирование информации о товаре»:

1. Открыть страницу товаров.
2. Нажать на кнопку «Изменить» рядом с товаром, информацию о котором нужно отредактировать.
3. Переписать информацию в некоторых полях.
4. Нажать на кнопку «Редактировать».

### **2.3.2.3   Прецедент администратора «Удаление товара»**

Основной успешный сценарий для прецедента «Удаление товара»:

1. Открыть страницу товаров.
2. Нажать на кнопку «Удалить» рядом с нужным товаром.

### **2.3.2.4   Прецедент администратора «Добавление товара»**

Основной успешный сценарий для прецедента «Добавление товара»:

1. Открыть страницу товаров.
2. Нажать на кнопку «Добавить» в нижней части страницы.
3. Написать информацию о товаре.
4. Нажать на кнопку «Готово».

### **2.3.2.5   Прецедент администратора «Просмотр информации о заказе»**

Основной успешный сценарий для прецедента «Просмотр информации о заказе»:

1. Открыть страницу заказов.
2. Нажать на кнопку «Открыть» рядом с нужным заказом.
3. Ознакомиться с представленной информацией.

### **2.3.2.6   Прецедент администратора «Просмотр информации о категории»**

Основной успешный сценарий для прецедента «Просмотр информации о категории»:

1. Открыть страницу категорий.
2. Нажать на кнопку «Открыть» рядом с категорией, о которой нужно узнать информацию.
3. Ознакомиться с представленной информацией.

### **2.3.2.7   Прецедент администратора «Редактирование информации о категории»**

Основной успешный сценарий для прецедента «Редактирование информации о категории»:

1. Открыть страницу категорий.
2. Нажать на кнопку «Изменить» рядом с категорией, информацию о которой нужно Отредактировать.
3. Переписать информацию в некоторых полях.
4. Нажать на кнопку «Редактировать».

### **2.3.2.8   Прецедент администратора «Удаление категории»**

Основной успешный сценарий для прецедента «Удаление категории»:

1. Открыть страницу категорий.

2. Нажать на кнопку «Удалить» рядом с нужной категорией.

### **2.3.2.9 Прецедент администратора «Добавление категории»**

Основной успешный сценарий для прецедента «Добавление категории»:

1. Открыть страницу категорий.
2. Нажать на кнопку «Добавить» в нижней части страницы.
3. Написать информацию о категории.
4. Нажать на кнопку «Готово».

### **2.3.2.10 Прецедент покупателя «Просмотр информации о товаре»**

Предусловие: открыта страница всех товаров или товаров категории.

Основной успешный сценарий для прецедента «Просмотр информации о товаре»:

1. Нажать на кнопку «Подробнее» в карточке интересующего товара.
2. Ознакомиться с представленной информацией.

### **2.3.2.11 Прецедент покупателя «Добавление товара в корзину»**

Предусловие: открыта страница всех товаров или товаров категории.

Основной успешный сценарий для прецедента «Добавление товара в корзину»:

1. Нажать на кнопку «Подробнее» в карточке интересующего товара.
2. Ознакомиться с представленной информацией.
3. Нажать на кнопку «Добавить в корзину».

### **2.3.2.12 Прецедент покупателя «Просмотр товаров категории»**

Основной успешный сценарий для прецедента «Просмотр товаров категории»:

1. Открыть страницу «Категории».
2. Нажать на интересующую категорию.
3. Ознакомиться с представленными товарами.

### **2.3.2.13 Прецедент покупателя «Просмотр информации о заказе»**

Предусловие: наличие заказов.

Основной успешный сценарий для прецедента «Просмотр информации о заказе»:

1. Открыть страницу заказов.
2. Нажать на кнопку «Открыть» рядом с нужным заказом.
3. Ознакомиться с представленной информацией.

### **2.3.2.14 Прецедент покупателя «Изменение количества товара»**

Предусловие: наличие товаров в корзине.

Основной успешный сценарий для прецедента «Изменение количества товара»:

1. Открыть корзину.
2. Нажимать на кнопку увеличения или уменьшения количества около нужного товара, пока не получится требуемое число.

### **2.3.2.15 Прецедент покупателя «Оформление заказа»**

Предусловие: наличие товаров в корзине.

Основной успешный сценарий для прецедента «Оформление заказа»:

1. Открыть корзину.
2. Нажать на кнопку «Оформить заказ».
3. Ввести необходимую информацию.
4. Нажать на кнопку «Оформить заказ».

## **2.3.3 Требования пользователя к интерфейсу web-сайта**

Сайт должен иметь следующие страницы:

- главная страница со всеми товарами;
- страница категорий товаров;
- страница товаров, принадлежащих выбранной категории;
- корзина;

- список заказов.

Администратору также необходимы страницы для редактирования, удаления и добавления товаров и категорий.

Товары на главной странице должны быть представлены в виде карточек одинакового размера, содержащих фото, название и цену товара. Пользователь должен иметь возможность фильтровать растения по характеристикам.

Макет страницы с товарами приведён на рисунке 2.3.

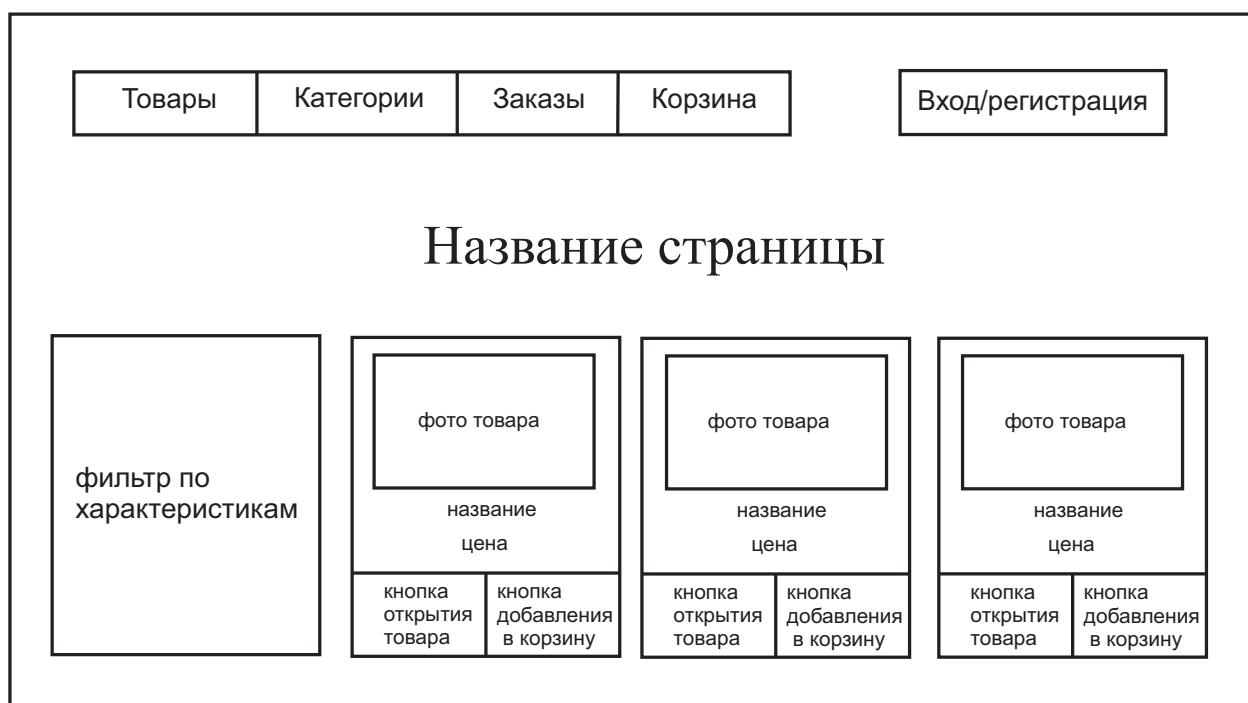


Рисунок 2.3 – Макет страницы с товарами

### 2.3.4 Нефункциональные требования

#### 2.3.4.1 Требования к безопасности

Необходимо устраниТЬ уязвимости, возможные для веб-приложений:

- Межсайтовые скрипты (XSS) — это уязвимость системы безопасности, которая позволяет злоумышленнику размещать клиентские скрипты (обычно JavaScript) на веб-страницах. Когда другие пользователи загружают затронутые страницы, будут выполняться скрипты злоумышленника, что позволяет злоумышленнику украсть cookie-маркеры и маркеры сеанса, изменить данные в базе данных и т.д.

нить содержимое веб-страницы с помощью DOM или перенаправить браузер на другую страницу.

- SQL-инъекция (Внедрение SQL-кода) — это попытка изменить запрос к базе данных. Ввести ее можно через форму или ссылку, которая передает параметры методом GET[15].
- Доступ обычного пользователя к возможностям администратора.
- Межсайтовая подделка запроса (CSRF) — это вид атак на сайт, при котором злоумышленник с помощью мошеннического сайта или скрипта заставляет браузер пользователя выполнять на доверенном сайте действия от его имени: отправлять сообщения, менять пароли, переводить деньги и пр. В атаке используются недостатки протокола HTTP. Чтобы стать жертвой, пользователю достаточно перейти по вредоносной ссылке.

#### **2.3.4.2 Требования к ПО**

Для реализации программной системы необходимо подготовить следующие компоненты:

- пакетный менеджер Composer для управления зависимостями;
- фреймворк Laravel;
- интерпретатор языка программирования PHP;
- СУБД MySQL;
- веб-сервер, поддерживающий PHP.

Laravel совместим со всеми современными операционными системами, включая Windows, macOS и Linux.

#### **2.3.4.3 Требования к аппаратному обеспечению**

Для работы приложения требуется дисковое пространство не менее 2,5 Гб, минимум 2 Гб оперативной памяти и подключение к Интернету. Рекомендуется использовать процессор с 2 или более ядрами и частотой 2 ГГц или выше.

## **2.4 Требования к оформлению документации**

Требования к стадиям разработки программ и программной документации для вычислительных машин, комплексов и систем независимо от их назначения и области применения, этапам и содержанию работ устанавливаются ГОСТ 19.102–77. Программная документация должна включать в себя:

1. Анализ предметной области.
2. Техническое задание.
3. Технический проект.

### 3 Технический проект

#### 3.1 Компоненты приложения

Компоненты приложения и взаимодействия между ними можно увидеть на диаграмме компонентов, приведённой на рисунке 3.1.

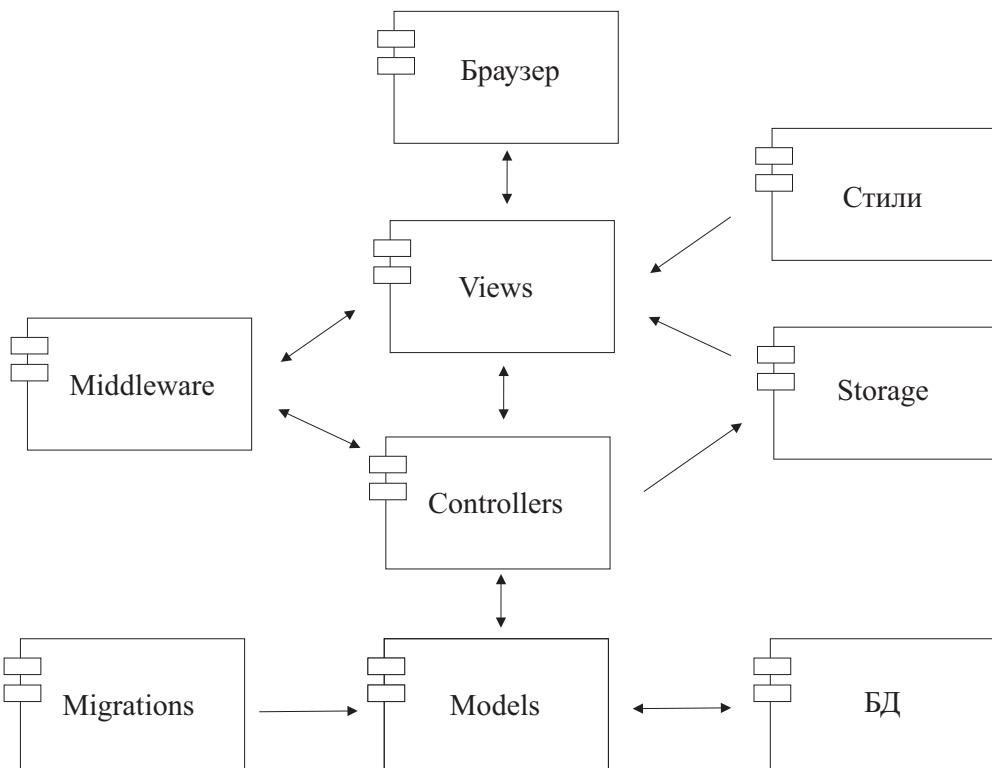


Рисунок 3.1 – Диаграмма компонентов приложения

#### 3.2 Обоснование выбора технологии проектирования

##### 3.2.1 PHP

PHP (Preprocessor Hypertext) — это язык программирования, который широко применяется для разработки веб-приложений и сайтов [3]. Он обладает рядом преимуществ, делающих его популярным среди разработчиков:

- Простота изучения: PHP относительно легко понять и освоить, особенно если у пользователя уже есть базовые знания HTML. Это делает его привлекательным выбором для начинающих программистов.

- Широкая поддержка: PHP имеет большое сообщество разработчиков, которое обеспечивает широкую поддержку и доступ к множеству ресурсов для обучения и решения проблем.
- Гибкость: PHP позволяет легко интегрировать различные технологии, такие как базы данных, XML, JSON и другие веб-технологии.
- Оптимизация производительности: PHP предлагает множество инструментов для оптимизации производительности, включая кэширование, минификацию кода и многопоточность.
- Поддержка MVC (Model-View-Controller): PHP поддерживает архитектуру MVC, которая помогает организовать код и упрощает разработку и поддержку больших приложений.
- Интеграция с другими языками программирования: PHP может быть использован вместе с другими языками программирования, такими как JavaScript, Python и Ruby. Это позволяет разработчикам использовать лучшие практики и инструменты из разных языков.
- Поддержка фреймворков: Существует множество популярных PHP-фреймворков, таких как Laravel, Symfony, CodeIgniter и Yii, которые представляют дополнительные инструменты и структуры для ускорения разработки.

### 3.2.2 MySQL

MySQL - это одна из самых популярных систем управления базами данных (СУБД). Это программа, которая позволяет создавать и управлять базами данных реляционного типа с использованием языка стандартизованных запросов (SQL). MySQL отличается своей доступностью как по финансовой стороне, так и по уровню сложности. Она проста в управлении и имеет развитое сообщество пользователей [1].

С практической точки зрения, система MySQL обладает рядом преимуществ:

- высокая скорость работы благодаря отказу от некоторых стандартов языка;

- простая установка и интуитивно понятный интерфейс, что делает её доступной даже для новичков;
- отсутствие ограничений по количеству пользователей;
- гибкость за счёт открытого исходного кода;
- богатый выбор сторонних инструментов (плагинов), что позволяет настроить систему под конкретные потребности;
- поддержка больших таблиц, содержащих до 50 миллионов строк;
- высокий уровень безопасности;
- широкий функционал, который позволяет использовать систему в различных сферах.

Эта СУБД считается надежной и стабильной уже долгие годы. Она поддерживает разные виды таблиц, быстро выполняет команды и может быть модифицирована под индивидуальные нужды проекта.

### 3.2.3 Blade

При разработке представлений можно использовать PHP либо язык шаблонизатора Blade. Код представлений будет более чистым, если использовать Blade [14].

Преимущества Blade:

- Простота использования. Для вывода данных достаточно использовать двойные фигурные скобки. Это позволяет избежать сложностей с выводом значений, экранированием строк и безопасностью.
- Безопасность. В отличие от PHP, где необходимо экранировать все выводимые данные, в Blade используются автоматические кавычки, что делает код более безопасным.
- Документированность. Blade хорошо документирован, что облегчает изучение и использование этого инструмента.
- Поддержка сообщества. Laravel имеет большое сообщество разработчиков, которые активно используют Blade и делятся своим опытом и решениями проблем.

- Расширяемость. Blade поддерживает пользовательские директивы, что позволяет расширять функциональность шаблонов без необходимости написания большого количества кода.

Выбор в пользу Blade добавляет дополнительный шаг в обработку шаблонов, т.к. все шаблоны Blade будут скомпилированы в PHP-код. Но благодаря кэшированию это не влияет на производительность приложения.

### 3.3 Архитектура Laravel

#### 3.3.1 Паттерн MVC

Архитектура Laravel основана на шаблоне MVC (Model-View-Controller). Это вариант архитектуры, при котором компоненты программы делятся на три части:

- модель (model) предоставляет данные и методы работы с ними: запросы в базу данных, проверка на корректность;
- представление (view) показывает пользователю эти данные;
- контроллер (controller) направляет данные от пользователя к системе и наоборот.

Модель MVC предназначена для разделения логических частей приложения и создания их независимо друг от друга.

#### 3.3.2 Жизненный цикл запроса

Точкой входа всех запросов является файл public/index.php. Он загружает сгенерированное Composer определение автозагрузчика и извлекает экземпляр приложения Laravel из файла bootstrap/app.php.

В зависимости от типа запроса, он отправляется либо в ядро HTTP, либо в ядро Console.

Ядро HTTP находится в файле app/Http/Kernel.php и расширяет класс Illuminate/Foundation/Http/Kernel. Оно определяет массив bootstrappers — загрузчиков, которые будут выполняться перед обработкой запроса. Эти загрузчики настраивают обработку ошибок, настраивают логи, определяют окру-

жение приложения и выполняют другие задачи, которые необходимо выполнить перед фактической обработкой запроса.

Ядро HTTP также определяет список HTTP-middleware, через которые должны пройти все запросы, прежде чем они будут переданы приложению. Эти middleware обрабатывают чтение и запись HTTP-сессии, определяют, находится ли приложение в режиме обслуживания, проверяют CSRF-токены и многое другое.

Одним из наиболее важных действий по начальной загрузке ядра является загрузка сервис-провайдеров — поставщиков услуг приложения. Сервис-провайдеры несут ответственность за начальную загрузку различных компонентов фреймворка, таких как базы данных, очереди, компоненты валидации и маршрутизации. Все сервис-провайдеры приложения настраиваются в массиве providers файла config/app.php.

После создания экземпляров провайдеров для них будет вызываться метод register. Как только все провайдеры будут зарегистрированы, для каждого из них будет вызван метод boot. Метод boot() в Laravel используется для определения кода, который должен быть выполнен при регистрации поставщика услуг. Этот метод обычно используется для регистрации пользовательских привязок, конфигураций и слушателей событий.

Одним из наиболее важных сервис-провайдеров в Laravel-приложении является App/Providers/RouteServiceProvider. Этот сервис-провайдер загружает файлы маршрутов, содержащиеся в каталоге routes.

После загрузки приложения и регистрации всех сервис-провайдеров запрос будет передан для обработки маршрутизатору. Маршрутизатор отправит запрос в маршрут или контроллер, а также запустит указанное middleware для данного маршрута. Если запрос проходит через все middleware соответствующего маршрута, будет выполнен метод маршрута или контроллера, а ответ, возвращённый методом маршрута или контроллера, будет отправлен обратно через цепочку middleware, давая приложению возможность изменить или проверить исходящий ответ.

Когда ответ возвращается через middleware, метод handle ядра HTTP возвращает объект ответа, а файл index.php вызывает метод send, который отправляет содержимое ответа в веб-браузер пользователя.

### 3.3.3 Сервис-провайдеры

Сервис-провайдеры являются центральным местом начальной загрузки всех возможностей Laravel. Под загрузкой подразумевается регистрация нужных сервисов, включая регистрацию привязок к служебному контейнеру, прослушивателей событий, middleware и даже маршрутов.

Все классы сервис-провайдеров хранятся в config/app.php. Провайдеры, имеющиеся там изначально, загружают основные компоненты Laravel, такие как почтовый рассыльщик, очередь, кэш и другие. Многие из этих классов провайдеров являются «отложенными». Это означает, что они будут загружаться не при каждом запросе, а только при необходимости предоставляемых ими услуг. Пользователь может создавать собственные сервис-провайдеры для своего приложения.

Все сервис-провайдеры расширяют класс Illuminate/Support/ServiceProvider. Класс провайдера содержит 2 метода — register и boot. Метод register предназначен для привязки к сервисному контейнеру. Внутри любого из методов класса-провайдера всегда доступно свойство \$app, которое предоставляет доступ к сервисному контейнеру. Метод boot используется для расширения возможностей фреймворка, например, для создания новых blade-директив и запускается только после выполнения всех методов register.

### 3.3.4 Сервис-контейнеры

Сервис-контейнер или контейнер служб – это инструмент для управления зависимостями классов и выполнения внедрения зависимостей. Зависимости классов «вводятся» в класс через конструктор в виде аргументов или, в некоторых случаях, через методы-сеттеры. При создании класса или вызове методов фреймворк смотрит на список аргументов и, если нужно, создаёт

экземпляры необходимых классов и сам подаёт их на вход конструктора или метода.

Все контроллеры регистрируются через сервис-контейнер, поэтому при получении класса контроллера из контейнера автоматически получаются все зависимости, указанные в аргументах конструктора и других методах контроллера.

### 3.3.5 Фасады

Фасады предоставляют «статический» интерфейс для классов, доступных в сервис-контейнере приложения. Все фасады Laravel расширяют базовый класс Illuminate\Support\Facades\Facade. Базовый класс Facade использует магический метод `__callStatic()`, чтобы делегировать вызовы из фасада объекту, извлеченному из контейнера.

Фасады предоставляют краткий и понятный синтаксис, который позволяет пользоваться функциями laravel без запоминания длинных названий классов, которые должны внедряться вручную. В дополнении к фасадам, Laravel предлагает множество глобальных вспомогательных функций, которые упрощают взаимодействие с общими функциями Laravel. Они доступны глобально и не требуют импортирования каких-либо классов для их использования. Многие вспомогательные функции выполняют те же задачи, что и соответствующие фасады.

## 3.4 Структура каталогов

Структура каталогов приложения включает следующие основные каталоги:

1. App — содержит подкаталоги Console, Exceptions, Http, Models и Providers. Каталог Console содержит пользовательские команды Artisan. Каталог Exceptions содержит обработчик исключений. Каталог Http содержит контроллеры, middleware и запросы форм. Каталог Models содержит классы моделей Eloquent. Каталог Providers содержит сервис провайдеры, подготавливающие приложение к входящим запросам.

2. Bootstrap — содержит файл app.php, загружающий фреймворк. Также в папке bootstrap находится папка cache, которая содержит сгенерированные фреймворком файлы для оптимизации производительности, например, кэш-файлы маршрутов и сервисов.
3. Config — содержит файлы конфигурации приложения.
4. Database — содержит миграции базы данных, фабрики моделей и данные для заполнения.
5. Public — содержит index.php, являющийся точкой входа для всех запросов, поступающих в приложение, и ссылку на storage.
6. Resources — содержит представления (view).
7. Routes — содержит все определения маршрутов для приложения.
8. Storage — разделён на подкаталоги storage/app, storage/framework и storage/logs. Каталог storage/framework используется для хранения кэшей. Storage/log хранит файлы журналов (логи) приложения. В каталоге storage/app/public хранятся изображения товаров.
9. Tests — содержит автоматические тесты.
10. Vendor — хранит зависимости, установленные через менеджер пакетов Composer.

### **3.5 Структура базы данных**

В таблице 3.1 показана структура таблицы plant\_categories.

Таблица 3.1 – Таблица plant\_categories

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	id	integer
	да	name	string
	да	description	text

В таблице 3.2 показана структура таблицы plant\_genera.

Таблица 3.2 – Таблица plant\_genera

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	name	string
	да	description	text

В таблице 3.3 показана структура таблицы product\_types.

Таблица 3.3 – Таблица product\_types

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	name	string

В таблице 3.4 показана структура таблицы plant\_colors.

Таблица 3.4 – Таблица plant\_colors

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	name	string

В таблице 3.5 показана структура таблицы climate\_zones.

Таблица 3.5 – Таблица climate\_zones

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	zone_number	integer
	да	lower_temp_limit	integer
	да	upper_temp_limit	integer

В таблице 3.6 показана структура таблицы light\_modes.

Таблица 3.6 – Таблица light\_modes

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	name	string

В таблице 3.7 показана структура таблицы soils.

Таблица 3.7 – Таблица soils

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	name	string

В таблице 3.8 показана структура таблицы life\_cycles.

Таблица 3.8 – Таблица life\_cycles

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	name	string

В таблице 3.9 показана структура таблицы plants.

Таблица 3.9 – Таблица plants

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	id	integer
	да	name	string
	да	description	text
	да	quantity	integer
	да	price	decimal
	нет	image	text
	да	flower_diameter	decimal
foreign key	да	genus	string
foreign key	да	product_type	string

### Продолжение таблицы 3.9

Тип ключа	Обязательность	Имя столбца	Тип данных
foreign key	да	life_cycle	string
foreign key	да	soil	string
foreign key	да	landing_place	string
foreign key	да	climate_zone	integer
foreign key	да	flower_color1	string
foreign key	нет	flower_color2	string
foreign key	нет	flower_color3	string
foreign key	да	leaf_color1	string
foreign key	нет	leaf_color2	string
foreign key	нет	leaf_color3	string

В таблице 3.10 показана структура таблицы users.

Таблица 3.10 – Таблица users

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	id	integer
	да	name	string
	да	email	string
	нет	email_verified_at	timestamp
	да	password	string
	да	is_admin	tinyInteger

В таблице 3.11 показана структура таблицы orders.

Таблица 3.11 – Таблица orders

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	id	integer

## Продолжение таблицы 3.11

Тип ключа	Обязательность	Имя столбца	Тип данных
	да	status	integer
	нет	name	text
	нет	phone	text
	нет	address	text
	нет	email	string
foreign key	нет	user_id	integer

В таблице 3.12 показана структура таблицы order\_plant.

Таблица 3.12 – Таблица order\_plant

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	id	integer
	да	plant_count	integer
foreign key	да	order_id	integer
foreign key	да	plant_id	integer

В таблице 3.13 показана структура таблицы plant\_plant\_category.

Таблица 3.13 – Таблица plant\_plant\_category

Тип ключа	Обязательность	Имя столбца	Тип данных
primary key	да	id	integer
foreign key	да	plant_category_id	integer
foreign key	да	plant_id	integer

## 3.6 Список контроллеров

Каталог Controller содержит следующие контроллеры:

- Controller – родительский контроллер для остальных контроллеров;

- MainController – отвечает за отображение некоторых страниц, в том числе главной страницы сайта;
- HomeController – отвечает за отображение заказов пользователя;
- BasketController – отвечает за добавление товара в корзину и отображение корзины.

Каталог Controller также содержит подкаталоги Admin и Auth.

### **3.6.1 Контроллеры подкаталога Admin**

Подкаталог Admin предназначен для контроллеров, предоставляющих методы для администратора сайта:

- CategoryController – ресурсный контроллер для управления категориями;
- OrderController – отвечает за отображение заказов у администратора;
- PlantController – ресурсный контроллер для управления товарами.

### **3.6.2 Контроллеры подкаталога Auth**

Подкаталог Auth объединяет контроллеры для регистрации и авторизации:

- ConfirmPasswordController – предназначен для обработки подтверждения пароля;
- ForgotPasswordController – отвечает за отправку электронных писем для сброса пароля;
- LoginController – обрабатывает аутентификацию ;
- RegisterController – обрабатывает регистрацию новых пользователей;
- ResetPasswordController – содержит логику сброса пароля;
- VerificationController – отвечает за переход пользователей на другую страницу после верификации.

## **3.7 Список представлений**

Каталог views содержит подкаталоги frontend, orders, auth, layouts и admin.

### **3.7.1 Представления подкаталога frontend**

Подкаталог frontend содержит страницы для просмотра и заказа товаров:

- index.blade.php – главная страница со всеми товарами;
- categories.blade.php – страница категорий товаров;
- category.blade.php – страница категории;
- product.blade.php – страница товара;
- basket.blade.php – корзина;
- order.blade.php – страница оформления заказа;

### **3.7.2 Представления подкаталога orders**

Подкаталог orders содержит страницы заказов пользователя:

- home.blade.php – страница заказов;
- order.blade.php – страница заказа.

### **3.7.3 Представления подкаталога auth**

Подкаталог auth содержит страницы авторизации и регистрации:

- login.blade.php – страница аутентификации;
- register.blade.php – страница регистрации;
- verify.blade.php – страница верификации;

В этом подкаталоге также есть раздел passwords, объединяющий страницы для пароля:

- confirm.blade.php – форма для подтверждения пароля;
- email.blade.php – форма для ввода адреса электронной почты для сброса пароля;
- reset.blade.php – форма для создания нового пароля.

### **3.7.4 Представления подкаталога layouts**

Подкаталог layouts содержит представления, используемые другими представлениями:

- master.blade.php – HTML-каркас для других представлений с подключёнными стилями и меню;
- card.blade.php – шаблон карточки товара;
- filters.blade.php – поле выбора характеристик, по которым можно отфильтровать товары.

### **3.7.5 Представления подкataloga admin**

Подкatalog admin делится на разделы categories, orders и plants.

Раздел categories содержит страницы для работы с категориями:

- index.blade.php – страница категорий;
- show.blade.php – страница категории;
- form.blade.php – страница добавления или редактирования категории.

Раздел orders содержит страницы для просмотра заказов:

- index.blade.php – страница заказов;
- show.blade.php – страница заказа;

Раздел plants содержит страницы для работы с товарами:

- index.blade.php – страница товаров;
- show.blade.php – страница товара;
- form.blade.php – страница добавления или редактирования товара.

## **3.8 Стили приложения**

Стили приложения можно найти по пути public/assets/css. Они представлены двумя файлами: bootstrap.min.css и starter-template.css. Эти стили подключаются в представлении layouts/master.blade.php. Файл bootstrap.min.css позволяет использовать возможности фреймворка Bootstrap для стилизации контента с помощью отдельных классов и компонентов. Одна из ключевых особенностей фреймворка – использование сетки, которая позволяет легко управлять распределением контента на разных размерах экранов.

## 4 Рабочий проект

### 4.1 Маршруты приложения

В таблице 4.1 показаны маршруты приложения.

Таблица 4.1 – Маршруты приложения

URI	HTTP-метод	Метод контроллера
/	GET HEAD	MainController@Index
/admin/categories	GET HEAD	Admin/CategoryController@index
/admin/categories	POST	Admin/CategoryController@store
/admin/categories/create	GET HEAD	Admin/CategoryController@create
/admin/categories/ {category}	GET HEAD	Admin/CategoryController@show
/admin/categories/ {category}	PUT PATCH	Admin/CategoryController@update
/admin/categories/ {category}	DELETE	Admin/CategoryController@destroy
/admin/categories/ {category}/edit	GET HEAD	Admin/CategoryController@edit
/admin/orders	GET HEAD	Admin/OrderController@index
/admin/orders/{order}	GET HEAD	Admin/OrderController@show
/admin/plants	GET HEAD	Admin/PlantController@index
/admin/plants	POST	Admin/PlantController@store
/admin/plants/create	GET HEAD	Admin/PlantController@create
/admin/plants/{plant}	GET HEAD	Admin/PlantController@show
/admin/plants/{plant}	PUT PATCH	Admin/PlantController@update
/admin/plants/{plant}	DELETE	Admin/PlantController@destroy
/admin/plants/{plant}/edit	GET HEAD	Admin/PlantController@edit
/basket	GET HEAD	BasketController@basket
/basket/add/{id}	POST	BasketController@basketAdd
/basket/place	GET HEAD	BasketController@basketPlace

## Продолжение таблицы 4.1

URI	HTTP-метод	Метод контроллера
/basket/place	POST	BasketController@basketConfirm
/basket/remove/{id}	POST	BasketController@basketRemove
/categories	GET HEAD	MainController@categories
/categories/{category}	GET HEAD	MainController@category
/products/{product}	GET HEAD	MainController@product
/home	GET HEAD	HomeController@index
/home/{order}	GET HEAD	HomeController@order
/login	GET HEAD	Auth/LoginController@showLoginForm
/login	POST	Auth/LoginController@login
/logout	POST	Auth/LoginController@logout
/logout	GET HEAD	Auth/LoginController@logout
/password/confirm	GET HEAD	Auth/ConfirmPasswordController@showConfirmForm
/password/confirm	POST	Auth/ConfirmPasswordController@confirm
/password/email	POST	Auth/ForgotPasswordController@sendResetLinkEmail
/password/reset	GET HEAD	Auth/ForgotPasswordController@showLinkRequestForm
/password/reset	POST	Auth/ResetPasswordController@reset
/password/reset/{token}	GET HEAD	Auth/ResetPasswordController@showResetForm
/register	GET HEAD	Auth/RegisterController@showRegistrationForm
/register	POST	Auth/RegisterController@register
/admin/categories	GET HEAD	Admin/CategoryController@index

## 4.2 Спецификация контроллеров приложения

В таблице 4.2 приведено описание методов класса «MainController».

Таблица 4.2 – Спецификация методов класса «MainController»

Название	Доступ	Описание
index	public	Выводит представление для главной страницы
categories	public	Выводит представление для категорий

## Продолжение таблицы 4.2

Название	Доступ	Описание
category	public	Выводит представление для категории
product	public	Выводит представление для товара

В таблице 4.3 приведено описание методов класса «HomeController».

Таблица 4.3 – Спецификация методов класса «HomeController»

Название	Доступ	Описание
__construct	public	Устанавливает доступ только для зарегистрированных пользователей
index	public	Выводит представление для заказов пользователя
order	public	Выводит представление для заказа пользователя

В таблице 4.4 приведено описание методов класса «BasketController».

Таблица 4.4 – Спецификация методов класса «BasketController»

Название	Доступ	Описание
basket	public	Выводит представление для корзины
basketPlace	public	Выводит представление для оформления заказа
basketConfirm	public	Сохраняет данные заказа и переносит пользователя на главную страницу после оформления заказа
basketAdd	public	Добавляет товар в корзину
basketRemove	public	Удаляет товар из корзины

Далее будет представлена спецификация контроллеров, находящихся в папке «Admin».

В таблице 4.5 приведено описание методов класса «CategoryController».

Таблица 4.5 – Спецификация методов класса «CategoryController»

Название	Доступ	Описание
index	public	Выводит представление для категорий в панели администратора
create	public	Выводит представление для создания категории
store	public	Сохраняет категорию в базу данных
show	public	Выводит представление для категории в панели администратора
edit	public	Выводит представление для редактирования категории
update	public	Обновляет информацию о категории в базе данных
destroy	public	Удаляет категорию из базы данных

В таблице 4.6 приведено описание методов класса «OrderController».

Таблица 4.6 – Спецификация методов класса «OrderController»

Название	Доступ	Описание
index	public	Выводит представление для заказов в панели администратора
show	public	Выводит представление для заказа в панели администратора

В таблице 4.7 приведено описание методов класса «PlantController».

Таблица 4.7 – Спецификация методов класса «PlantController»

Название	Доступ	Описание
index	public	Выводит представление для товаров в панели администратора
create	public	Выводит представление для создания товара
store	public	Сохраняет товар в базу данных

## Продолжение таблицы 4.7

Название	Доступ	Описание
show	public	Выводит представление для товара в панели администратора
edit	public	Выводит представление для редактирования товара
update	public	Обновляет информацию о товаре в базе данных
destroy	public	Удаляет товар из базы данных

Далее будет представлена спецификация контроллеров, находящихся в папке «Auth».

В таблице 4.8 приведено описание методов класса «ConfirmPasswordController».

Таблица 4.8 – Спецификация методов класса «ConfirmPasswordController»

Название	Доступ	Описание
__construct	public	Разрешает доступ только авторизованным пользователям

В таблице 4.9 приведено описание методов класса «LoginController».

Таблица 4.9 – Спецификация методов класса «LoginController»

Название	Доступ	Описание
redirectTo	protected	Переносит пользователя на другую страницу после входа
__construct	public	Разрешает доступ только неавторизованным пользователям

В таблице 4.10 приведено описание методов класса «RegisterController».

Таблица 4.10 – Спецификация методов класса «RegisterController»

Название	Доступ	Описание
redirectTo	protected	Переносит пользователя на другую страницу после регистрации
__construct	public	Разрешает доступ только неавторизованным пользователям
validator	protected	Валидирует данные для регистрации пользователя
create	protected	Сохраняет нового пользователя в базе данных

В таблице 4.11 приведено описание методов класса «ResetPasswordController».

Таблица 4.11 – Спецификация методов класса «ResetPasswordController»

Название	Доступ	Описание
redirectTo	protected	Переносит пользователя на другую страницу после сброса пароля

В таблице 4.12 приведено описание методов класса «VerificationController».

Таблица 4.12 – Спецификация методов класса «VerificationController»

Название	Доступ	Описание
redirectTo	protected	Переносит пользователя на другую страницу после верификации
__construct	public	Разрешает доступ только авторизованным пользователям

### 4.3 Системное тестирование разработанного web-сайта

На рисунке 4.1 показана главная страница сайта.

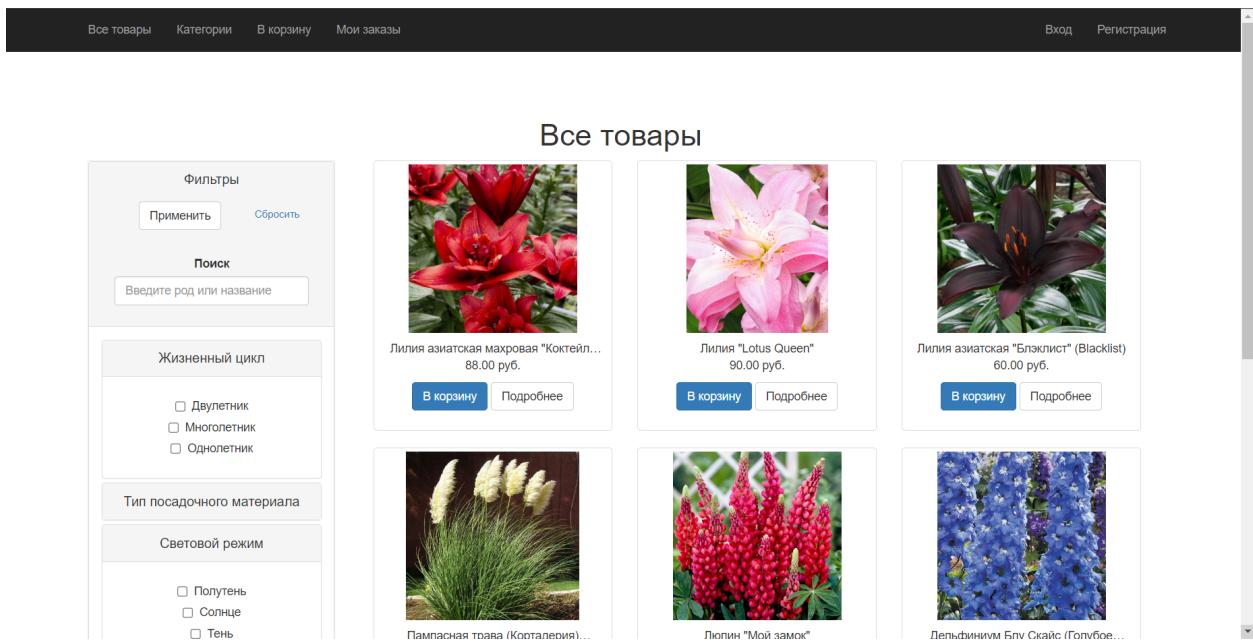


Рисунок 4.1 – Главная страница

На рисунке 4.2 показан результат фильтрации товаров на главной странице по одному параметру.

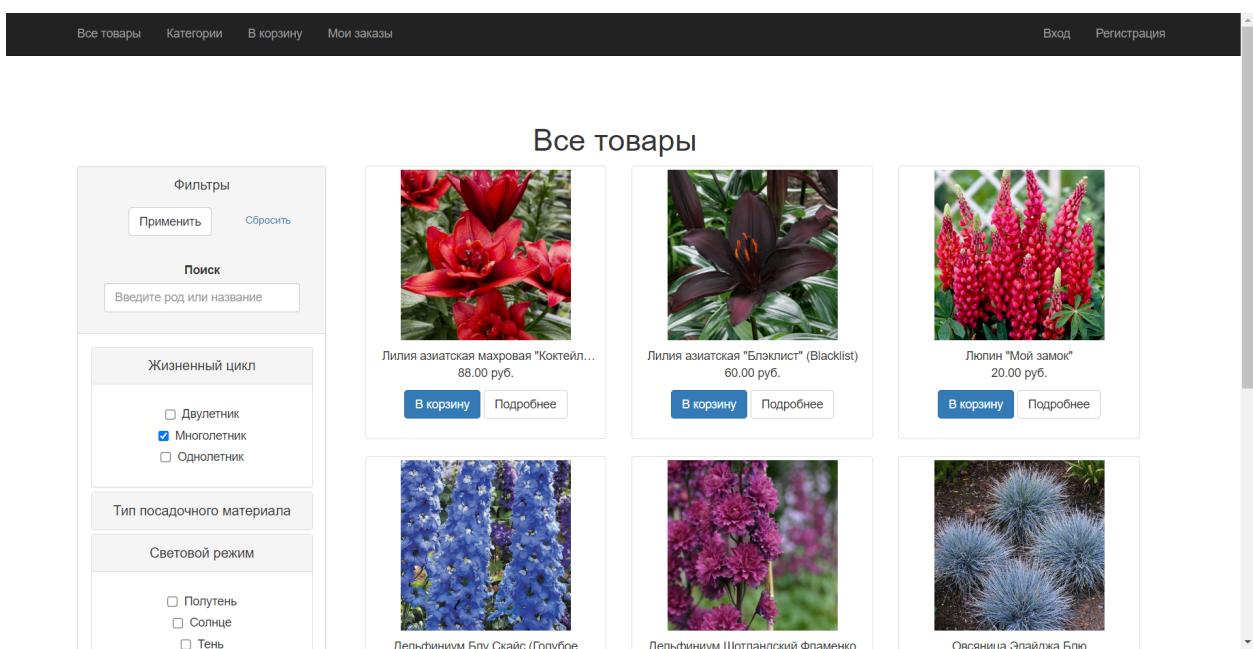


Рисунок 4.2 – Фильтрация по одному параметру

На рисунке 4.3 показан результат фильтрации товаров на главной странице по нескольким параметрам.

Все товары Категории В корзину Мои заказы Вход Регистрация

### Все товары

**Фильтры**

[Применить](#)
[Сбросить](#)

**Поиск**

**Жизненный цикл**

- Двулетник
- Многолетник
- Однолетник

**Тип посадочного материала**

**Световой режим**

**Почва**

- Глинистая
- Песчаная



Лилия азиатская "Блэклист" (Blacklist)  
60.00 руб.

[В корзину](#)
[Подробнее](#)

Рисунок 4.3 – Фильтрация по нескольким параметрам

На рисунке 4.4 показан результат поиска товаров по названию.

Все товары Категории В корзину Мои заказы Вход Регистрация

### Все товары

**Фильтры**

[Применить](#)
[Сбросить](#)

**Поиск**

**Жизненный цикл**

- Двулетник
- Многолетник
- Однолетник

**Тип посадочного материала**

**Световой режим**

- Полутень
- Солнце
- Тень



Лилия азиатская "Блэклист" (Blacklist)  
60.00 руб.

[В корзину](#)
[Подробнее](#)



Дельфиниум Блу Скайс (Голубое...  
560.00 руб.

[В корзину](#)
[Подробнее](#)



Овсяница Эладжа Блю  
220.00 руб.

[В корзину](#)
[Подробнее](#)



Яблоня Валентиновка  
740.00 руб.

[В корзину](#)
[Подробнее](#)

Рисунок 4.4 – Поиск по названию

На рисунке 4.5 показана страница одного из товаров.

## Овсяница Элайджа Блю

Цена: 220.00 руб.



Род	Овсяница
Тип посадочного материала	Корневище
Жизненный цикл	Многолетник
Почва	Песчаная
Световой режим	Солнце
Климатическая зона	3

[В корзину](#)

Овсяница Элайджа Блю - невысокая многолетняя трава с необычными узкими листьями сизого или синевато-стального оттенка. Образует полукруглый пышный куст высотой 25-30 см. Цветет в июне-июле. Соцветия серо-зеленые, в мягких метелках на прямом стебле, после цветения становятся светло-коричневыми.

## Рисунок 4.5 – Страница товара

На рисунке 4.6 показана страница категории.

### Плодовые

Отправка Плодовых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

### Хвойные

Отправка Хвойных саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

### Вьющиеся

Отправка Вьющихся саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

### Злаки и декоративные травы

Отправка Злаковых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

### Цветущие

Отправка Цветущих саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

## Рисунок 4.6 – Страница категорий

На рисунке 4.7 показана страница категории «Злаки и декоративные травы».

Все товары Категории В корзину Мои заказы Вход Регистрация

## Злаки и декоративные травы

Отправка Злаковых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

**Фильтры**

[Применить](#) [Сбросить](#)

**Поиск**

**Жизненный цикл**

Двулетник  
 Многолетник  
 Однолетник

**Тип посадочного материала**

**Световой режим**

Полутень  
 Солнце



Пампасная трава (Кортадерия)...  
150.00 руб.

[В корзину](#) [Подробнее](#)



Овсяница Элайджа Блю  
220.00 руб.

[В корзину](#) [Подробнее](#)

Рисунок 4.7 – Страница категории

После применения фильтрации на странице категории выводятся товары, принадлежащие этой категории и соответствующие выбранным значениям характеристик. Это показано на рисунке 4.8.

Все товары Категории В корзину Мои заказы Вход Регистрация

## Злаки и декоративные травы

Отправка Злаковых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г

**Фильтры**

[Применить](#) [Сбросить](#)

**Поиск**

**Жизненный цикл**

Двулетник  
 Многолетник  
 Однолетник

**Тип посадочного материала**

**Световой режим**

**Почва**

Глинистая  
 Песчаная  
 Сугенистая



Пампасная трава (Кортадерия)...  
150.00 руб.

[В корзину](#) [Подробнее](#)

Рисунок 4.8 – Фильтрация на странице категории

На рисунке 4.9 показана корзина до добавления в неё товаров.

## Корзина

Ваша корзина пуста. Добавьте в неё что-нибудь!

Рисунок 4.9 – Пустая корзина

На рисунке 4.10 показано добавление товара в корзину.

Скриншот интернет-магазина с добавленным товаром в корзину:

Товар Дельфиниум Шотландский Фламенко был добавлен в корзину

### Все товары

Фильтры  
[Применить](#) [Сбросить](#)

Поиск  
Введите род или название

Жизненный цикл  
 Двухлетник  
 Многолетник  
 Однолетник

Тип посадочного материала

Световой режим  
 Полнопанель

Изображение	Название	Цена	Действия
	Лилия азиатская махровая "Коктейль..." 88.00 руб.		<a href="#">В корзину</a> <a href="#">Подробнее</a>
	Лилия "Lotus Queen" 90.00 руб.		<a href="#">В корзину</a> <a href="#">Подробнее</a>
	Лилия азиатская "Блэклист" (Blacklist) 60.00 руб.		<a href="#">В корзину</a> <a href="#">Подробнее</a>

Рисунок 4.10 – Добавление товара в корзину

На рисунке 4.11 показана корзина с товарами.

Корзина				
Название	Кол-во	Цена	Стоимость	
 Люпин Гладиатор	<span>-</span> <span>1</span> <span>+</span>	512.00 руб.	512 руб.	
 Дельфиниум Шотландский Фламенко	<span>-</span> <span>1</span> <span>+</span>	675.00 руб.	675 руб.	
Общая стоимость:	1187 руб.			
				<span>Оформить заказ</span>

Рисунок 4.11 – Корзина с товарами

На рисунке 4.12 показано удаление товара из корзины.

Корзина				
Название	Кол-во	Цена	Стоимость	
 Люпин Гладиатор	<span>-</span> <span>1</span> <span>+</span>	512.00 руб.	512 руб.	
Удален товар Дельфиниум Шотландский Фламенко				
Общая стоимость:	512 руб.			
				<span>Оформить заказ</span>

Рисунок 4.12 – Удаление товара из корзины

На рисунке 4.13 показана страница оформления заказа.



### Подтвердите заказ:

Общая стоимость заказа: 512 руб.

Укажите свои данные для отправки заказа:

Имя:

Номер телефона:

Адрес доставки:

E-mail:

**Подтвердить заказ**

Рисунок 4.13 – Страница оформления заказа

На рисунке 4.14 показана страница входа.



### Вход

E-Mail

Пароль

Запомнить меня

**Войти**    Забыли пароль?

Рисунок 4.14 – Страница входа

На рисунке 4.15 показана страница регистрации.

**Регистрация**

<b>Имя</b>	<input type="text"/>
<b>E-mail</b>	<input type="text"/>
<b>Пароль</b>	<input type="password"/>
<b>Повторите пароль</b>	<input type="password"/>
<b>Зарегистрироваться</b>	

Рисунок 4.15 – Страница регистрации

На рисунке 4.16 показан результат оформления заказа.

Все товары Категории В корзину Мои заказы Выход

Ваш заказ принят в обработку!

**Все товары**

**Фильтры**

[Применить](#) [Сбросить](#)

**Поиск**

Введите род или название

**Жизненный цикл**

Двухлетник  Многолетник  Однолетник

**Тип посадочного материала**

**Световой режим**



Лилия азиатская махровая "Коктейль" 88.00 руб.

[В корзину](#) [Подробнее](#)



Лилия "Lotus Queen" 90.00 руб.

[В корзину](#) [Подробнее](#)



Лилия азиатская "Блэклист" (Blacklist) 60.00 руб.

[В корзину](#) [Подробнее](#)







Рисунок 4.16 – Результат оформления заказа

На рисунке 4.17 показана страница заказов пользователя.

Ваши заказы						
Дата оформления заказа	Имя	Телефон	Адрес доставки	Количество товаров	Сумма	Действия
27/05/2024	Головина Ирина Романовна	+79200000000	г. Курск, ул. Бойцов 9 дивизии	1	512 руб.	<button>Открыть</button>

Рисунок 4.17 – Страница заказов пользователя

На рисунке 4.18 показана страница с деталями заказа пользователя.

Заказ от 27/05/2024			
Заказчик: Головина Ирина Романовна			
Адрес доставки: г. Курск, ул. Бойцов 9 дивизии			
Название	Кол-во	Цена	Стоимость
 Люпин Гладиатор	1	512.00 руб.	512 руб.
Общая стоимость:			512 руб.

Рисунок 4.18 – Страница заказа пользователя

На рисунке 4.19 показана страница заказов в панели администратора.

## Заказы

Id заказа	Имя	Телефон	Адрес доставки	E-mail	Дата оформления заказа	Сумма	Действия
5	Головина Ирина Романовна	+79200000000	г. Курск, ул. Бойцов 9 дивизии	golovina1@yandex.ru	10:44 27/05/2024	512 руб.	<button>Открыть</button>

Рисунок 4.19 – Страница заказов в панели администратора

На рисунке 4.20 показана страница с деталями заказа в панели администратора.

## Заказ №5

Заказчик: Головина Ирина Романовна

Номер телефона: +79200000000

Название	Кол-во	Цена	Стоимость
 Люпин Гладиатор	1	512.00 руб.	512 руб.
Общая стоимость:			512 руб.

Рисунок 4.20 – Страница заказа в панели администратора

На рисунке 4.21 показана страница категорий в панели администратора.

Категории			
ID	Название	Описание	
1	Плодовые	Отправка Плодовых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
2	Хвойные	Отправка Хвойных саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
3	Вьющиеся	Отправка Вьющихся саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
4	Злаки и декоративные травы	Отправка Злаковых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
5	Цветущие	Отправка Цветущих саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>

Рисунок 4.21 – Страница категорий в панели администратора

На рисунке 4.22 показана страница категории «Плодовые» в панели администратора.

Категория Плодовые	
Поле	Значение
ID	1
Название	Плодовые
Описание	Отправка Плодовых саженцев Весна 2024 будет производиться с 15.03.24 г до 15.05.24 г
Кол-во товаров	1

Рисунок 4.22 – Страница категории в панели администратора

На рисунке 4.23 показана страница редактирования категории «Плодовые» в панели администратора.



**Редактировать категорию Плодовые**

**Название:**

**Описание:**

**Сохранить**

Рисунок 4.23 – Страница редактирования категории в панели администратора

На рисунке 4.24 показана страница создания категории в панели администратора.



**Добавить категорию**

**Название:**

**Описание:**

**Сохранить**

Рисунок 4.24 – Страница создания категории в панели администратора

На рисунке 4.25 показана страница товаров в панели администратора.

Товары					
ID	Название	Описание	Количество	Цена	
14	Лилия азиатская махровая "Коктейль Твинз"	Азиатская лилия Коктейль Твинз (Cocktail Twins) - это эффектное растение с крупными махровыми цветками, с слегка закрученными кончиками, открытой чащевидной формы. Сорт имеет лепестки окрашенные в темно-красный цвет, который в некоторых местах переливается то бордовым, то вишневым оттенком. Луковицы яйцевидной или округлой формы. Имеет мощные корни. На стебле располагается обильнокветущее соцветие, которое имеет до 7 цветков. Цветы обращены вверх, возвышаются на сильных стеблях. Листья темного зеленого цвета, прочные и узкие. Они расположены в поочередном порядке. Предпочитает расти в солнечных местах, на плодородной, щелочной, хорошо дренированной почве. Растение морозостойчивое. Можно использовать как в групповых так и в одиночных посадках. Используется для украшения цветников, бордюров. Долго сохраняет свежий вид в букете.	120	88.00	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
15	Лилия "Lotus Queen"	Нежно-розовые цветы "Lotus Queen" привлекают внимание своей утонченностью и поистине королевской красотой. Форма цветка напоминает лотос. Восточная махровая лилия завоевала свое законное место на цветниках и клумбах, создавая необыкновенную красоту на клумбе во второй половине лета.	90	90.00	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
16	Лилия азиатская "Блэклист" (Blacklist)	Лилия азиатская "Блэклист" - выделяется темно-бордовыми цветами и сочной листовой с блеском, которые укрывают высокий стебель. Лепестки и тычинки черно-бордовые, а пыльники ярко-оранжевые. Стебли крепкие и держат несколько крупных бутонов, которые направлены вверх. Цветение будет невероятно эффектным, если высадить 3-5 луковиц на расстоянии 10-15 см. Возможно выращивание в контейнерах, а букеты с черными лилиями особенно цепки среди флористов. В миксбордере сорт Блэклист не потерпится даже среди роз, подойдет ему и яркие флоксы или пестролистные хосты. Эффектно выглядят в групповой посадке.	150	60.00	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>
17	Пампасная трава (Кортадерия) Пумила	Пампасная трава (Кортадерия) Пумила - Многолетнее растение, образующее плотные, крупные кусты. Сорт низкорослый, в высоту достигает 45 см, во время цветения достигает высоты до 1,5 м. Цветет с августа по октябрь. Соцветия имеют кремовый цвет. Листья в основном прикорневые, линейные, широкие, серо-зеленые, оттянутые на конце, по краю мелкозубчатые.	200	150.00	<button>Открыть</button> <button>Изменить</button> <button>Удалить</button>

Рисунок 4.25 – Страница товаров в панели администратора

На рисунке 4.26 показана страница одного из товаров в панели администратора.

Товар Лилия азиатская махровая "Коктейль Твинз"	
Поле	Значение
ID	14
Название	Лилия азиатская махровая "Коктейль Твинз"
Описание	Азиатская лилия Коктейль Твинз (Cocktail Twins) - это эффектное растение с крупными махровыми цветками, с слегка закрученными кончиками, открытой чащевидной формы. Сорт имеет лепестки окрашенные в темно-красный цвет, который в некоторых местах переливается то бордовым, то вишневым оттенком. Луковицы яйцевидной или округлой формы. Имеет мощные корни. На стебле располагается обильнокветущее соцветие, которое имеет до 7 цветков. Цветы обращены вверх, возвышаются на сильных стеблях. Листья темного зеленого цвета, прочные и узкие. Они расположены в поочередном порядке. Предпочитает расти в солнечных местах, на плодородной, щелочной, хорошо дренированной почве. Растение морозостойчивое. Можно использовать как в групповых так и в одиночных посадках. Используется для украшения цветников, бордюров. Долго сохраняет свежий вид в букете.
Род	Лилия
Вид посадочного материала	Луковицы
Жизненный цикл	Многолетник
Почва	Сугнилистая
Место посадки	Солнце
Климатическая зона	4 (от -34 до -29)
Диаметр цветка (см)	12.00

Рисунок 4.26 – Страница товара в панели администратора

На рисунке 4.27 показана страница редактирования товара в панели администратора.

Заказы Категории Товары Выход

### Редактировать товар **Лилия азиатская махровая "Коктейль Твинз"**

**Название:** Лилия азиатская махровая "Коктейль Твинз"

**Относится к категориям:** -- Выберите подходящие значения --  
Плодовые  
Хвойные  
Вьющиеся  
Овощи

**Описание:** Азиатская лилия Коктейль Твинз (Cocktail Twins) - это эффектное растение с крупными махровыми цветками, с легким закрученными кончиками, открытой чашевидной формой. Сорт имеет лепестки окрашенные в темно-красный цвет, который в некоторых местах переливается то бордовым, то вишневым оттенком. Луковицы яйцевидной или округлой формы. Имеет мощные корни.

**Род:** Лилия

**Тип посадочного материала:** Луковицы

**Жизненный цикл:** Многолетник

Рисунок 4.27 – Страница редактирования товара в панели администратора

На рисунке 4.28 показана страница создания товара в панели администратора.

Заказы Категории Товары Выход

### Добавить товар

**Название:** [пустое поле]

**Относится к категориям:** -- Выберите подходящие значения --  
Плодовые  
Хвойные  
Вьющиеся  
Овощи

**Описание:** [пустое поле]

**Род:** -- Выберите одно значение --

**Тип посадочного материала:** -- Выберите одно значение --

**Жизненный цикл:** -- Выберите одно значение --

Рисунок 4.28 – Страница создания товара в панели администратора

## ЗАКЛЮЧЕНИЕ

В процессе выполнения данной работы была создана программно-информационная система для продажи декоративных растений. Были реализованы функции для администратора и покупателей.

Покупатели имеют возможность просматривать товары, используя фильтры по характеристикам и поиск по названию, и оформлять заказы. Администратор может просматривать список заказов, а также контролировать товары и их категории.

Основные результаты работы:

1. Проведен анализ предметной области.
2. Разработана концептуальная модель web-сайта. Определены требования к системе.
3. Осуществлено проектирование web-сайта. Разработана архитектура серверной части. Разработан пользовательский интерфейс web-сайта.
4. Реализован и протестирован web-сайт.

Все требования, объявленные в техническом задании, были полностью реализованы, все задачи, поставленные в начале разработки проекта, были также решены.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бретт, М. PHP и MySQL. Исчерпывающее руководство / М. Бретт. – Санкт-Петербург : Питер, 2016 г. – 544 с. – ISBN 978-5-496-01049-8. – Текст : непосредственный.
2. Веру, Л. Секреты CSS. Идеальные решения ежедневных задач / Л. Веру. – Санкт-Петербург : Питер, 2016 г. – 336 с. – ISBN 978-5-496-02082-4. – Текст : непосредственный.
3. Гизберт, Д. PHP и MySQL / Д. Гизберт. – Москва : НТ Пресс, 2013 г. – 320 с. – ISBN 978-5-477-01174-2. – Текст : непосредственный.
4. Дэкетт, Д. HTML и CSS. Разработка и создание веб-сайтов / Д. Дэкетт. – Москва : Эксмо, 2014 г. – 480 с. – ISBN 978-5-699-64193-2. – Текст : непосредственный.
5. Грофф, Дж. P. SQL: полное руководство / Грофф Дж. Р., Вайнберг П. Н., Оппель Э. Дж. — 3-е изд. — Санкт-Петербург : Диалектика, 2020 г. — 960 с. - ISBN 978-5-907114-26-5. – Текст : непосредственный.
6. Шварц, Б. MySQL. Оптимизация производительности / Шварц Б., Зайцев П., Ткаченко В., Заводны Дж., Ленц А., Бэллинг Д. — 2-е изд. — Санкт-Петербург : Символ-Плюс, 2010 г. — 832 с. - ISBN 978-5-93286-153-0. – Текст : непосредственный.
7. Кингсбери, Н. Основы озеленения сада / Кингсбери Н. — 1-е изд. — Москва : Кладезь-Букс, 2003 г. — 208 с. - ISBN: 5-93395-034-3. – Текст : непосредственный.
8. Кривко, Н. П. Питомниководство садовых культур / Под редакцией Кривко Н. П. — Санкт-Петербург : Лань, 2015 г. — 368 с. - ISBN: 978-5-8114-1761-2. – Текст : непосредственный.
9. Коновалова, Т. Ю. Декоративные кустарники, или 1000 растений для вашего сада. Иллюстрированный справочник / Т. Ю. Коновалова, Н. А. Шевырева — Москва : Фитон+, 2004 г. — 192 с. - ISBN: 5-93457-070-6. – Текст : непосредственный.

10. Берд, Р. Цветущие деревья и кустарники: дизайн вашего сада и советы по выращиванию / Ричард Берд — Москва : АРТ-РОДНИК, 2003 г. — 159 с. - ISBN: 5-88896-089-6. – Текст : непосредственный.
11. Хессайон, Д. Г. Всё о декоративных растениях и кустарниках / Хессайон Д. Г. — 2-е изд.. — Москва : Кладезь-Букс, 2007 г. — 127 с. - ISBN: 978-5-93395-238-1. – Текст : непосредственный.
12. Ганичкина, О. А. Декоративные растения вашего сада: деревья, кустарники, цветы / Ганичкина О.А. — Москва : Эксмо, 2008 г. — 222 с. - ISBN: 978-5-699-28077-3. – Текст : непосредственный.
13. Марковский, Ю. Б. Хвойные растения для декоративного сада / Ю. Б. Марковский, И. В. Успенский. — Москва : Фитон XXI, 2016 г. — 232 с. - ISBN: 978-5-906171-91-7. – Текст : непосредственный.
14. Кириченко, А. В. Laravel для WEB-разработчиков. Практическое руководство по созданию профессиональных сайтов / А. В. Кириченко, Е. В. Дубовик. — Санкт-Петербург :Наука и Техника, 2021 г. - 432 с. - ISBN: 978-5-94387-726-1. – Текст : непосредственный.
15. Дронов, В. А. Laravel. Быстрая разработка современных динамических Web-сайтов на PHP, MySQL, HTML и CSS / В. А. Дронов. — Санкт-Петербург : БХВ-Петербург, 2018 г. - 755 с. - ISBN: 978-5-9775-3845-9. – Текст : непосредственный.
16. Лежнева, Т. Н. Основы декоративного садоводства: учеб. пособие для студ. учреждений сред. проф. образования / Т. Н. Лежнева. — Москва : Академия, 2011 г. - 80 с. - ISBN: 978-5-7695-5899-3. – Текст : непосредственный.
17. Соколова, Т. А. Декоративное растениеводство: Цветоводство: учебник для студ. вузов / Т. А. Соколова, И. Ю. Бочкова. — Москва : Академия, 2017 г. - 432 с. - ISBN: 978-5-7695-6566-3. – Текст : непосредственный.
18. Рак, Ярослав. Энциклопедия комнатных растений / Ярослав Рак. — Москва : Ниола-Пресс, 2009 г. - 192 с. - ISBN: 978-5-366-00427-5. – Текст : непосредственный.

19. Бурлуцкая, Л. А. Комнатные растения / Л. А. Бурлуцкая. — Москва : АСТ, 2009 г. - 238 с. - ISBN: 978-5-17-059056-8. – Текст : непосредственный.
20. Бабин, Д. М. Цветоводство от А до Я / Д. М. Бабин. — Минск : Харвест, 2007 г. - 703 с. - ISBN: 978-985-16-2767-3. – Текст : непосредственный.
21. Капранова, Е. Г. Комнатные растения / Е. Г. Бурлуцкая. — Москва : АСТ, 2010 г. - 320 с. - ISBN: 978-5-17-064127-7. – Текст : непосредственный.
22. Сладкова, О. В. Комнатные растения / О. В. Сладкова. — Москва : АСТ, 2010 г. - 143 с. - ISBN: 978-5-271-26723-9. – Текст : непосредственный.
23. Майер, Ганс-Петер. Цитрусовые растения / Ганс-Петер Майер. — Москва : ОЛМА Медиа Групп, 2012 г. - 123 с. - ISBN: 978-5-373-01682-7. – Текст : непосредственный.
24. Дорошенко, Т. Н. Луковичные растения / Т. Н. Дорошенко. — Москва : Фолио, 2010 г. - 221 с. - ISBN: 978-966-03-4229-3. – Текст : непосредственный.
25. Донская, М. В. Комнатное цветоводство для начинающих / М. В. Донская. — Москва : КМК, 2012 г. - 91 с. - ISBN: 978-5-87317-783-7. – Текст : непосредственный.
26. Головкин, Б. Н. Всё о комнатных растениях / Б. Н. Головкин. — Москва : Айрис-пресс, 2005 г. - 366 с. - ISBN: 5-8112-1143-0. – Текст : непосредственный.
27. Пескова, И. М. Лучшие садовые цветы. Большая иллюстрированная энциклопедия / И. М. Пескова, А. Б. Лысиков. — Москва : Эксмо, 2013 г. - 256 с. - ISBN: 978-5-699-59829-8. – Текст : непосредственный.
28. Колесникова, Е. Н. Простое цветоводство: неприхотливые комнатные растения / Е. Н. Колесникова. — Ростов-на-Дону : Феникс, 2014 г. - 48 с. - ISBN: 978-5-222-21638-5. – Текст : непосредственный.
29. Мовсесян, Л. И. Справочник садовода в вопросах и ответах / Л. И. Мовсесян. — Ростов-на-Дону : Феникс, 2015 г. - 284 с. - ISBN: 978-5-222-25013-6. – Текст : непосредственный.

30. Лепаж, Россин. Садовые цветы. Подробное иллюстрированное руководство / Россин Лепаж, Денис Ретурнард. — Москва : Эксмо, 2015 г. - 240 с. - ISBN: 978-5-699-63469-9. – Текст : непосредственный.
31. Вьюгин, С. М. Цветоводство и питомниководство: учебное пособие для подготовки бакалавров, обучающихся по направлению «Садоводство» / С. М. Вьюгин, Г. В. Вьюгина. — Санкт-Петербург : Лань, 2016 г. - 143 с. - ISBN: 978-5-8114-2116-9. – Текст : непосредственный.
32. Вьюгин, С. М. Цветоводство защищённого грунта. Учебное пособие для вузов / С. М. Вьюгин, Г. В. Вьюгина. — Санкт-Петербург : Лань, 2022 г. - 124 с. - ISBN: 978-5-507-45417-4. – Текст : непосредственный.
33. Вьюгин, С. М. Цветоводство и питомниководство. Учебное пособие / С. М. Вьюгин, Г. В. Вьюгина. — Санкт-Петербург : Лань, 2023 г. - 144 с. - ISBN: 978-5-507-47966-5. – Текст : непосредственный.
34. Вьюгина, Г. В. Цветоводство открытого грунта. Учебное пособие для вузов / Г. В. Вьюгина, С. М. Вьюгин. — Санкт-Петербург : Лань, 2022 г. - 256 с. - ISBN: 978-5-507-45418-1. – Текст : непосредственный.
35. Кайгородцева, А. А. Цветоводство для себя и на продажу. Подробный гайд по выращиванию самых популярных растений / А. А. Кайгородцева. — Москва : Бомбара, 2022 г. - 240 с. - ISBN: 978-5-04-156241-0. – Текст : непосредственный.
36. Бочкова, И. Ю. Цветоводство. Теория и практика / И. Ю. Бочкова. — Москва : Фитон XXI, 2022 г. - 192 с. - ISBN: 978-5-6048603-0-4. – Текст : непосредственный.
37. Белочкина, Ю. В. Лианы / Ю. В. Белочкина. — Москва : Фолио, 2007 г. - 222 с. - ISBN: 966-03-3697-7. – Текст : непосредственный.
38. Бумбеева, Л. И. Розы / Л. И. Бумбеева. — Москва : Кладезь, 2010 г. - 256 с. - ISBN: 978-5-93395-336-4. – Текст : непосредственный.
39. Доналдсон, Стефани. Практическая энциклопедия цветоводства / Стефани Доналдсон. — Москва : Росмэн, 2001 г. - 512 с. - ISBN: 5-8451-0407-6. – Текст : непосредственный.

40. Карпisonова, Р. А. Цветоводство / Р. А. Карпisonова. — Москва : Кладезь, 2007 г. - 256 с. - ISBN: 978-5-93395-152-0. – Текст : непосредственный.
41. Князева, Т. П. Садовые растения. Новейшая энциклопедия / Т. П. Князева, Д. В. Князева. — Москва : ОЛМА Медиа Групп, 2009 г. - 272 с. - ISBN: 978-5-373-02426-6. – Текст : непосредственный.
42. Ладвинская, А. А. Ампельные и вьющиеся растения / А. А. Ладвинская. — Москва : Мир книги, 2006 г. - 240 с. - ISBN: 5-486-00581-4. – Текст : непосредственный.
43. Ладвинская, А. А. Декоративнолистные растения / А. А. Ладвинская. — Москва : Мир книги, 2006 г. - 240 с. - ISBN: 5-486-00501-6. – Текст : непосредственный.
44. Беляевская, Е. К. Луковичные и клубневые растения / Е. К. Беляевская. — Москва : Мир книги, 2006 г. - 240 с. - ISBN: 5-486-00537-7. – Текст : непосредственный.
45. Алдохина, Т. В. Размножение растений / Т. В. Алдохина. — Москва : Мир книги, 2006 г. - 240 с. - ISBN: 5-486-00535-0. – Текст : непосредственный.
46. Алдохина, Т. В. Красиво цветущие растения / Т. В. Алдохина. — Москва : Мир книги, 2006 г. - 240 с. - ISBN: 5-486-00483-4. – Текст : непосредственный.
47. Александрова, М. С. Приусадебное цветоводство / М. С. Александрова. — Москва : Лабиринт, 2002 г. - 336 с. - ISBN: 5-9287-0074-1. – Текст : непосредственный.
48. Стоун, Б. The Everything Store. Джейф Безос и эра Amazon / Б. Стоун. — Москва : Азбука, 2021 г. – 448 с. – ISBN: 978-5-389-15490-2. – Текст : непосредственный.
49. Российская Федерация. Законы. Правила продажи товаров при дистанционном способе продажи товара по договору розничной куплипродажи: Постановление Правительства РФ №2463: принято 31 декабря 2020 г. // КонсультантПлюс: сайт. URL:

[http://www.consultant.ru/document/cons\\_doc\\_LAW\\_373622/](http://www.consultant.ru/document/cons_doc_LAW_373622/) (дата обращения: 19.04.2024). Режим доступа: для зарегистрир. пользователей.

50. Буч, Г. Введение в UML от создателей языка / Г. Буч, И. Якобсон, Д. Рамбо. – Москва : ДМК Пресс, 2015 г. – 498 с. – ISBN: 978-5-457-43379-3. – Текст : непосредственный.
51. Вирт, Н. Алгоритмы и структуры данных / Н. Вирт. – Москва : ДМК Пресс, 2010 г. – 274 с. – ISBN: 978-5-04-034809-1. – Текст : непосредственный.
52. Хоффман, Э. Безопасность веб-приложений / Э. Хоффман. – Санкт-Петербург : Питер, 2021 г. – 336 с. – ISBN: 978-5-4461-1786-4. – Текст : непосредственный.
53. Мартин, Р. Чистая архитектура. Искусство разработки программного обеспечения / Р. Мартин. – Санкт-Петербург : Питер, 2018 г. – 352 с. – ISBN: 978-5-4461-0772-8. – Текст : непосредственный
54. Мандел, Т. Разработка пользовательского интерфейса / Т. Мандел. – Москва : ДМК Пресс, 2019 г. – 420 с. – ISBN: 978-5-04-195060-6. – Текст : непосредственный.

## **ПРИЛОЖЕНИЕ А**

### **Представление графического материала**

Графический материал, выполненный на отдельных листах, изображен на рисунках А.1–А.6.

## Сведения о ВКРБ

Минобрнауки России  
Юго-Западный государственный университет

## Кафедра программной инженерии

# **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА ПО ПРОГРАММЕ БАКАЛАВРИАТА**

«Интернет-магазин декоративных растений»

Руководитель ВКРБ  
к.т.н., доцент  
Аникина Елена Игоревна

Автор ВКРБ  
студент группы ПО-026  
Головина Ирина Романовна

## Рисунок А.1 – Сведения о ВКРБ

## **Цель и задачи разработки**

Цель настоящей работы – разработка web-сайта для продажи растений через Интернет.

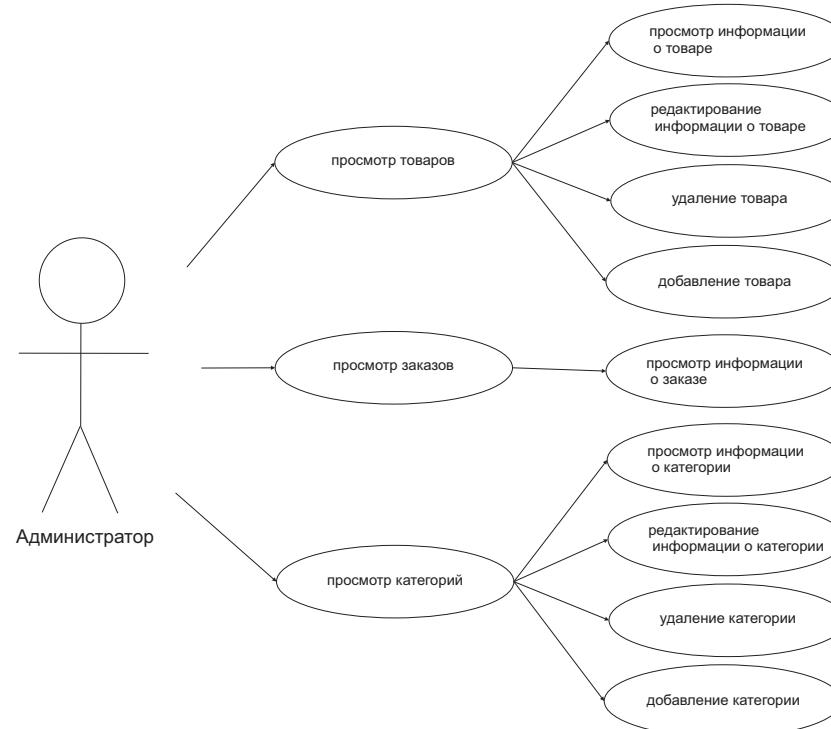
В соответствии с целью работы были сформулированы следующие задачи:

- провести анализ предметной области;
  - разработать концептуальную модель web-сайта;
  - спроектировать web-сайт;
  - реализовать сайт средствами web-технологий.

		ВКРБ 206843.09.03.04.24.001		
		Сведения о ВКРБ		
		Лиц	Неск	Неск
<i>Изве ржени е о резу льтатах исследо ваний</i>	<i>Фамилия И. О.</i>	<i>Полное наименование</i>		
	<i>Григорьев И. Р.</i>	<i>Акционерная компания "Инвест-Медиа"</i>		
<i>Изве ржени е о резу льтатах исследо ваний</i>	<i>Фамилия И. О.</i>	<i>Полное наименование</i>		
	<i>Чечеткин А. А.</i>	<i>ООО "Инвест-Медиа"</i>		
<i>Высокоскоростная информационная платформа "Бизнес-Банк"</i>				
<i>ВЭГУ ПО-2020</i>				

## Рисунок А.2 – Цель и задачи разработки

# Диаграмма прецедентов. Часть 1



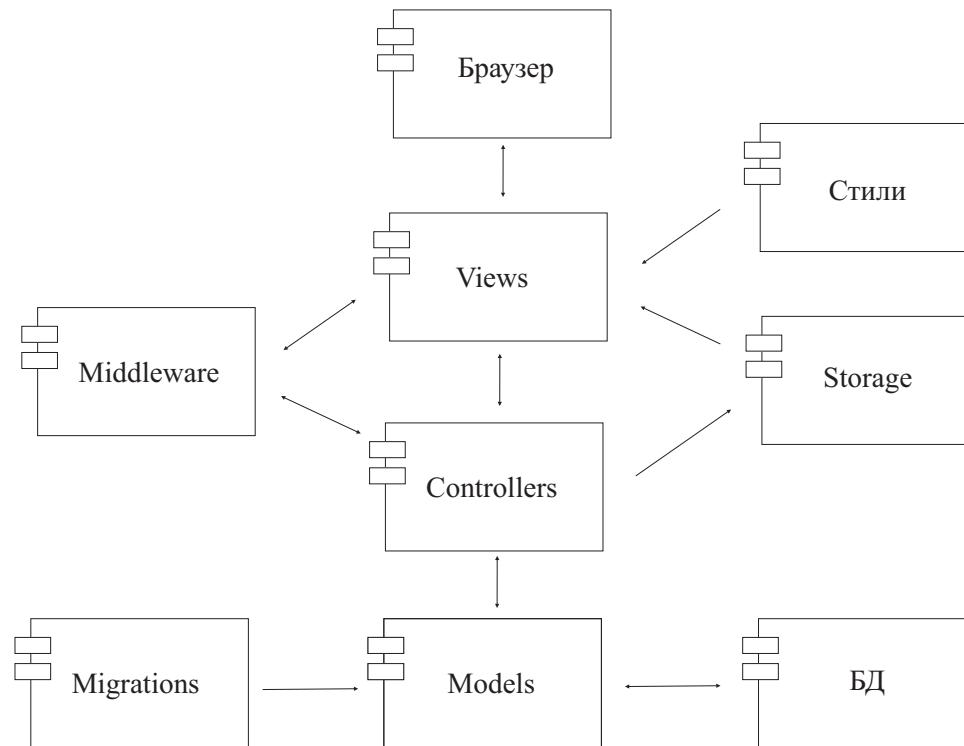
	ВКРБ 2068443.09.03.04.26.001
	Сведения о ВКРБ
Анкета работника	Фамилия И. В. Геннадий Н. Родившийся в 1960 г. в г. Краснодаре Национальность - Чеченец А. А.
Документы	Паспорт № 311 Лицензия № Адм. № 10-026
Виды профессиональной работы бендериста	ИЗГУ № 026

Рисунок А.3 – Диаграмма прецедентов. Часть 1



Рисунок А.4 – Диаграмма прецедентов. Часть 2

## Диаграмма компонентов



	ВКРБ 2068443.09.03.04.26.001
	Сведения о ВКРБ
	Анк. №/код. №/Ном.

### Рисунок А.5 – Диаграмма компонентов

## Заключение

В процессе выполнения данной работы была создана программно-информационная система для продажи декоративных растений. Были реализованы функции для администратора и покупателей.

Покупатели имеют возможность просматривать товары, используя фильтры по характеристикам и поиск по названию, и оформлять заказы. Администратор может просматривать список заказов, а также контролировать товары и их категории.

Основные результаты работы:

1. Проведен анализ предметной области.
2. Разработана концептуальная модель web-сайта. Определены требования к системе.
3. Осуществлено проектирование web-сайта. Разработана архитектура серверной части. Разработан пользовательский интерфейс web-сайта.
4. Реализован и протестирован web-сайт.

Все требования, объявленные в техническом задании, были полностью реализованы, все задачи, поставленные в начале разработки проекта, были также решены.

ВКРБ 2068443.09.03.04.24.001		
Лист	Номер	Номер
Информация о ВКРБ		
Лист 6	Лист 6	
Выпускная квалификационная работа бакалавра		ДЭГУ ПО-026

Рисунок А.6 – Заключение

## ПРИЛОЖЕНИЕ Б

### Фрагменты исходного кода программы

#### BasketController.php

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Http\Controllers\Controller;
6 use App\Models\order;
7 use App\Models\Plant;
8 use Illuminate\Http\Request;
9 use App\Services\CurrencyConversion;
10 use Illuminate\Support\Facades\Auth;
11
12 class BasketController extends Controller
13 {
14     public function basket()
15     {
16         $orderId = session('orderId');
17         if (!is_null($orderId)){
18             $order = Order::findOrFail($orderId);
19             foreach ($order->products as $product)
20                 $product->compareCount();
21         }
22         else $order = array();
23         return view('frontend/basket', compact('order'));
24     }
25
26     public function basketConfirm(Request $request)
27     {
28         $orderId = session('orderId');
29         if (is_null($orderId)) {
30             return redirect()->route('index');
31         }
32         $order = Order::find($orderId);
33         $success = true;
34         foreach ($order->products as $product) {
35             if (!$product->productIsEnough($product->getCountInOrder()))
36                 $success = false;
37         }
38         if ($success){
39             $user = null;
40             if(Auth::check())
41                 $user = Auth::user()->id;
42                 $success = $order->saveOrder($request->name, $request->phone, $request
43                     ->address, $request->email, $user);
44         }
45         if ($success) {
46             foreach ($order->products as $product){
47                 $product->reduceQuantity($product->getCountInOrder());
48             }
49             session()->flash('success', 'Ваш заказ принят в обработку!');
50         }
51     }
52 }
```

```

49 } else {
50     session()->flash('warning', 'Случилась ошибка');
51 }
52 return redirect()->route('index');
53 }

54
55 public function basketPlace()
56 {
57     $orderId = session('orderId');
58     if (is_null($orderId)) {
59         return redirect()->route('index');
60     }
61     $order = Order::find($orderId);
62     return view('frontend/order', compact('order'));
63 }

64
65 public function basketAdd($productId)
66 {
67     $orderId = session('orderId');
68     if (is_null($orderId)) {
69         $order = Order::create();
70         session(['orderId' => $order->id]);
71     } else {
72         $order = Order::find($orderId);
73     }
74     if ($order->products->contains($productId)) {
75         $product = $order->products()->where('plant_id', $productId)->first();
76         $pivotRow = $product->pivot;
77         if ($pivotRow->plant_count < $product->quantity)
78         {
79             $pivotRow->plant_count++;
80             $pivotRow->update();
81         } else
82             session()->flash('warning', 'Невозможно добавить новый товар, так как
83                 на складе нет такого
84                 количества данного товара');
85     } else {
86         $order->products()->attach($productId);
87     }
88     if (Auth::check()) {
89         $order->user_id = Auth::id();
90         $order->save();
91     }
92
93     $product = Plant::find($productId);
94
95     return redirect()->back()->with('success', 'Товар ' . $product->name. ' был
96         добавлен в корзину');
97 }
98
99 public function basketRemove($productId)
100 {
101     $orderId = session('orderId');

```

```

101     if ($orderId == null) {
102         return redirect()->route('basket');
103     }
104     $order = Order::find($orderId);
105     if ($order->products->contains($productId)) {
106         $pivotRow = $order->products()->where('plant_id', $productId)->first()
107             ->pivot;
108         if ($pivotRow->plant_count < 2) {
109             $order->products()->detach($productId);
110         } else {
111             $pivotRow->plant_count--;
112             $pivotRow->update();
113         }
114     }
115     $product = Plant::find($productId);
116
117     session()->flash('warning', 'Удален товар ' . $product->name);
118
119     return redirect()->route('basket');
120 }
121 }
122 }
```

### HomeController.php

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\order;
6 use App\Models\User;
7 use Illuminate\Http\Request;
8 use Illuminate\Support\Facades\Auth;
9
10 class HomeController extends Controller
11 {
12     /**
13     * Create a new controller instance.
14     *
15     * @return void
16     */
17     public function __construct()
18     {
19         $this->middleware('auth');
20     }
21
22     /**
23     * Show the application dashboard.
24     *
25     * @return \Illuminate\Contracts\Support\Renderable
26     */
27     public function index()
28     {
29         $orders = Auth::user()->orders()->where('status', 1)->get();
```

```

30     return view( 'orders/home' , compact( 'orders' ));
31 }
32
33 public function order(Order $order){
34     if (!Auth::user()->orders->contains($order))
35         return back();
36     return view( 'orders/order' , compact( 'order' ));
37 }
38 }
```

## MainController.php

```

1 <?php
2
3 namespace App\Http\Controllers;
4
5 use App\Models\Plant;
6 use App\Models\PlantCategory;
7 use App\Http\Filters\PlantFilter;
8 use App\Http\Requests\PlantRequest;
9
10 class MainController extends Controller
11 {
12     public function Index(PlantRequest $request){
13         $data = $request->validated();
14         $filter = app()->make(PlantFilter::class , [ 'queryParams' => array_filter(
15             $data)]);
16         $products = Plant::filter($filter)->paginate(9);
17         return view( 'frontend/index' , [
18             'products' => $products]);
19     }
20
21     public function categories() {
22         $categories = PlantCategory::paginate(6);
23         return view( 'frontend/categories' , compact( 'categories' ));
24     }
25
26     public function category($id , PlantRequest $request) {
27         $data = $request->validated();
28         $filter = app()->make(PlantFilter::class , [ 'queryParams' => array_filter(
29             $data)]);
30         $category = PlantCategory::where( 'id' , $id)->first();
31         if(!is_object($category))
32             return abort(404);
33         $products = $category->plants()->filter($filter)->paginate(9);
34         return view( 'frontend/category' , compact( 'category' , 'products'));
35     }
36
37     public function product($id) {
38         $product = Plant::where( 'id' , $id)->first();
39         if(!is_object($product))
40             return abort(404);
41         return view( 'frontend/product' , compact( 'product' ));
42     }
43 }
```

## CategoryController.php

```
1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use App\Http\Controllers\Controller;
6 use App\Models\PlantCategory;
7 use Illuminate\Http\Request;
8
9 class CategoryController extends Controller
10 {
11     /**
12      * Display a listing of the resource.
13      */
14     public function index()
15     {
16         $categories = PlantCategory::paginate(5);
17         return view('admin/categories/index', compact('categories'));
18     }
19
20     /**
21      * Show the form for creating a new resource.
22      */
23     public function create()
24     {
25         return view('admin/categories/form');
26     }
27
28     /**
29      * Store a newly created resource in storage.
30      */
31     public function store(Request $request)
32     {
33         PlantCategory::create($request->all());
34         return redirect()->route('categories.index');
35     }
36
37     /**
38      * Display the specified resource.
39      */
40     public function show(PlantCategory $category)
41     {
42         return view('admin/categories/show', compact('category'));
43     }
44
45     /**
46      * Show the form for editing the specified resource.
47      */
48     public function edit(PlantCategory $category)
49     {
50         return view('admin/categories/form', compact('category'));
51     }
52
53     /**
```

```

54 * Update the specified resource in storage.
55 */
56 public function update(Request $request, PlantCategory $category)
57 {
58     $category->update($request->all());
59     return redirect()->route('categories.index');
60 }
61
62 /**
63 * Remove the specified resource from storage.
64 */
65 public function destroy(PlantCategory $category)
66 {
67     $category->delete();
68     return redirect()->route('categories.index');
69 }
70 }
```

### OrderController.php

```

1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use App\Http\Controllers\Controller;
6 use App\Models\order;
7
8 class OrderController extends Controller
9 {
10     public function index()
11     {
12         $orders = Order::where('status', 1)->paginate(10);
13         return view('admin.orders.index', compact('orders'));
14     }
15
16     public function show(Order $order)
17     {
18         return view('admin.orders.show', compact('order'));
19     }
20 }
```

### PlantController.php

```

1 <?php
2
3 namespace App\Http\Controllers\Admin;
4
5 use App\Http\Controllers\Controller;
6 use App\Models\Plant;
7 use App\Models\PlantCategory;
8 use Illuminate\Http\Request;
9 use Illuminate\Support\Facades\Storage;
10
11 class PlantController extends Controller
12 {
13     /**

```

```

14 * Display a listing of the resource.
15 */
16 public function index()
17 {
18     $plants = Plant::paginate(5);
19     return view('admin/plants/index', compact('plants'));
20 }
21
22 /**
23 * Show the form for creating a new resource.
24 */
25 public function create()
26 {
27     return view('admin/plants/form');
28 }
29
30 /**
31 * Store a newly created resource in storage.
32 */
33 public function store(Request $request)
34 {
35     $params = $request->all();
36     if ($path = $request->file('image'))
37     {
38         $img_path = $path->store('public/plants'); //фото будут сохраняться в
39         папку plants
40         $params['image'] = $img_path;
41     }
42     $plant = Plant::create($params);
43     $categories = PlantCategory::find($request->get('plant_category'));
44     if ($categories)
45         $plant->categories()->saveMany($categories);
46     return redirect()->route('plants.index');
47 }
48
49 /**
50 * Display the specified resource.
51 */
52 public function show(Plant $plant)
53 {
54     return view('admin/plants/show', compact('plant'));
55 }
56
57 /**
58 * Show the form for editing the specified resource.
59 */
60 public function edit(Plant $plant)
61 {
62     return view('admin/plants/form', compact('plant'));
63 }
64
65 /**
66 * Update the specified resource in storage.
67 */

```

```

67  public function update(Request $request, Plant $plant)
68  {
69      $params = $request->all();
70      if ($file = $request->file('image'))
71      {
72          $img_path = $file->store('public/plants'); //фото будут сохраняться в
73          //папку plants
74          $params['image'] = $img_path;
75          if ($old_img = $plant->image)
76              Storage::delete($old_img);
77      }
78      else $params['image'] = $plant->image;
79      $plant->update($params);
80      $categories = PlantCategory::find($request->get('plant_category'));
81      if ($categories)
82      {
83          $plant->categories()->detach();
84          $plant->categories()->saveMany($categories);
85      }
86      return redirect()->route('plants.index');
87  }
88 /**
89 * Remove the specified resource from storage.
90 */
91 public function destroy(Plant $plant)
92 {
93     $plant->delete();
94     if (!is_null($plant_image = $plant->image))
95         Storage::delete($plant_image);
96     return redirect()->route('plants.index');
97 }
98 }
```

## Представление layouts/master.blade.php

```

1 <!doctype html>
2 <html lang="en">
3 <head>
4     <meta charset="utf-8">
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7
8     <title>
9         @admin Панель администратора:
10        @else Интернет Магазин:
11        @endadmin
12        @yield('title')
13     </title>
14
15    <link href="https://fonts.googleapis.com/css?family=Raleway:100,600" rel=
16          "stylesheet" type="text/css">
17    <link href="/assets/css/bootstrap.min.css" rel="stylesheet">
18    <link href="/assets/css/starter-template.css" rel="stylesheet">
```

```

19  </head>
20  <body>
21      <nav class="navbar navbar-inverse navbar-fixed-top">
22          <div class="container">
23              <div id="navbar" class="collapse navbar-collapse">
24                  <ul class="nav navbar-nav">
25                      @admin
26                          <li><a href="{{ route('orders') }}>Заказы</a></li>
27                          <li><a href="{{ route('categories.index') }}>Категории</a></li>
28                          <li><a href="{{ route('plants.index') }}>Товары</a></li>
29                      @else
30                          <li><a href="{{ route('index') }}>Все товары</a></li>
31                          <li><a href="{{ route('categories') }}>Категории</a></li>
32                          <li><a href="{{ route('basket') }}>В корзину</a></li>
33                          <li><a href="{{ route('home') }}>Мои заказы</a></li>
34                  @endadmin
35          </ul>
36          <ul class="nav navbar-nav navbar-right">
37              @guest
38                  <li><a href="{{ route('login') }}>Вход</a></li>
39                  <li><a href="{{ route('register') }}>Регистрация</a></li>
40              @endguest
41
42              @auth
43                  <li><a href="{{ route('get-logout') }}>Выход</a></li>
44              @endauth
45          </ul>
46      </div>
47  </div>
48 </nav>
49
50 <div class="container">
51     <div class="starter-template">
52         @if(session()->has('success'))
53             <p class="alert alert-success">{{ session()->get('success') }}</p>
54         @endif
55         @if(session()->has('warning'))
56             <p class="alert alert-warning">{{ session()->get('warning') }}</p>
57         @endif
58         @yield('content')
59     </div>
60 </div>
61 </body>
62 </html>

```

## Представление frontend/index.blade.php

```

1  @extends('layouts.master')
2
3  @section('title', 'Главная')
4
5  @section('content')
6      <h1>Все товары</h1>
7
8      <div class="form-group row">

```

```
9   <div class="col-xs-3">
10     @include( 'layouts.filters' , [ 'route' => route( 'index' )]);
11   </div>
12   <div class="col-xs-9">
13     @foreach($products as $product)
14       @include( 'layouts.card' , compact( 'product' ))
15     @endforeach
16   </div>
17   {{$products->withQueryString()->links( ( 'pagination::bootstrap-4' ))}}
18 </div>
19 @endsection
```

**Место для диска**