**Содержание**

Изм.

Кол

Лист №док

Подпись

Дата

Лист

2

ТРПО 2-40 01 01.33.41.04.25 ПЗ

Разраб.

Ганисевский

Пров.

Заяц

Н. контр.

Утв.

Разработка интернет-магазина «Мидланд»

Стадия

Листов

50

УО ГГПК

Введение………………………………………………………………….2

1 Постановка задачи .4

* 1. Организационно-экономическая сущность задачи .4
  2. Функциональные требования .5
  3. Описание исходной(входной) информации .6
  4. Описание результатной(выходной) информации…………………7
  5. Описание используемой условно-постоянной информации .9
  6. Эксплуатационные требования……………………………………..10

1. Выбор стратегии разработки и модели жизненного цикла 11
2. Руководство программиста………………………………………….15
   1. Создание сайта………………………………………………………15
   2. Настройка функциональности и дизайна………………………….15
   3. Редактирование страниц……………………………………………16
   4. Настройка Header и Footer………………………………………….16
   5. Редактирование главной страницы сайта………………………….16
3. Тестирование………………………………………………………....18
   1. Тест-кейсы……………………...…………………………………....18
   2. Результаты тестирования…………………………………………...20
4. Руководство пользователя…………………………………………..22
   1. Общие сведения о программном продукте………………………..22
   2. Запуск программы…………………………………………………..23
   3. Инструкции по работе с программой……………………………...24
   4. Завершение работы с программой…………………………………30
5. Заключение…………………………………………………………...32
6. Список использованных источников……………………………….34
7. Приложение А…………………………………………….………….35
8. Приложение Б…………………………………………….…………..42
9. Приложение В…………………………………………….…………..46

**Введение**

Тема проекта – разработка интернет-магазина для продажи свежих продуктов. Основной задачей является создание простой и удобной платформы, где покупатели смогут быстро находить и заказывать свежие овощи, фрукты и другие продукты. Интернет-магазин обеспечит доступ к актуальной информации о наличии товаров, их стоимости и условиях доставки.

Проект направлен на решение базовых задач онлайн-торговли свежими продуктами, таких как предоставление удобного интерфейса для выбора и заказа продукции, упрощение поиска необходимых товаров и минимизация времени на оформление покупки. Интернет-магазин станет инструментом, который сделает процесс приобретения свежих продуктов более доступным, понятным и комфортным для пользователей.

Тема актуальна в рамках проекта, так как разработка такого ресурса позволяет применить на практике знания в области веб-программирования, проектирования пользовательских интерфейсов и работы с базами данных.

Описание разделов пояснительной записки:

Первый подраздел, озаглавленный «Постановка задачи» состоит из определении требований к системе, формировании цели и задач проекта. Уточнении функциональных и нефункциональных требований. Диаграмма вариантов использования – графическое представление взаимодействия пользователей с системой через диаграммы вариантов использования. В выбор стратегии разработки и модели жизненного цикла входит обоснование выбора подхода к разработке. Описание технологий, языков программирования, платформ и сред разработки, которые будут использоваться расположено в подразделе «Инструменты разработки». Разработка плана работы над проектом – создание плана-графика выполнения задач с указанием этапов разработки, сроков и ответственных лиц.

Разработка структуры сайта, системы меню, навигации – планирование информационной архитектуры сайта: структура страниц, логика меню, навигационные цепочки. Разработка UML-диаграмм – подготовка диаграмм UML: диаграммы классов, последовательностей, взаимодействия и активности.

Руководство программиста – разработка инструкций для программистов, включающих описание установки, настройки и модификации кода. Подраздел «Диаграмма компонентов» включает создание диаграммы компонентов для отображения архитектуры системы и взаимодействия между её частями.

Тестирование – проведение тестирования системы, включая описание методологий, этапов и устранение выявленных ошибок с документированием результатов.

Руководство пользователя – разработка инструкций для конечных пользователей, охватывающих регистрацию, оформление заказа и использование личного кабинета.

Заключение – обобщение проделанной работы, достижение поставленных целей и оценка перспектив развития проекта.

**1. Постановка задачи**

**1.1. Организационно-экономическая сущность задачи**

**Наименование задачи:** разработка и запуск интернет-магазина “Мидланд” с системой онлайн-заказов и доставкой домой.

**Цель разработки:** создание удобной и доступной платформы, которая позволит клиентам легко выбирать и заказывать свежие продукты и овощи онлайн, а также обеспечить им быструю и надежную доставку до двери.

**Назначение:** для клиентов: сайт предназначен для всех, кто заинтересован в покупке свежих продуктов и овощей без необходимости посещения магазина.

**Периодичность использования:** расписание доставки: использование ПП будет связано с этим графиком доставки. Частота покупок клиентами: если клиенты часто делают покупки продуктов через ваш сайт, то использование ПП будет регулярным и частым.

**Источники и способы получения информации:** каталог товаров, заказы и покупки, данные клиентов, информация о доставке.

**Информационная связь с другими задачами:** интеграция с системами управления складом, связь с системами обработки платежей.

**Обзор существующих аналогичных ПП:** edostavka ([Доставка продуктов на дом по Минску и Минской области | Едоставка (edostavka.by)](https://edostavka.by/)) – это сервис онлайн-покупок, который предоставляет возможность заказа продуктов питания, свежих овощей и фруктов с доставкой на дом.

Азбука Вкуса ([Азбука вкуса - доставка продуктов на дом в Москве и Санкт-Петербурге, заказать онлайн продукты на дом (av.ru)](https://av.ru/)) – предоставляет услуги доставки свежих продуктов и готовой еды по аналогии с Amazon Fresh. Компания предлагает широкий ассортимент продуктов высокого качества и возможность доставки на дом.

Утконос ([Онлайн-магазин Утконос — доставка продуктов на дом в Москве и области, купить продукты питания с доставкой в интернет-магазине (utkonos.ru)](https://www.utkonos.ru/?utm_referrer=https%3A%2F%2Fwww.google.com%2F)) – это онлайн-сервис, предоставляющий услуги доставки продуктов питания, свежих продуктов и готовой еды. Утконос имеет собственные магазины и склады, а также предлагает широкий выбор товаров для заказа через интернет.

**1.2. Функциональные требования**

1. Пользователи:
   1. Возможность создания учетной записи для новых пользователей;
   2. Аутентификация пользователей при входе в систему;
   3. Просмотр каталога продуктов с детальным описанием, изображениями, категориями, марками и ценами;
   4. Возможность добавления выбранных продуктов в корзину для последующего оформления заказа;
   5. Заполнение формы заказа с указанием адреса доставки, способа оплаты и времени доставки;
   6. Возможность просмотра и редактирования содержимого корзины перед оформлением заказа.
2. Поставщик:
   1. Добавление, редактирование и удаление информации о продуктах(товарах);
   2. Планирование и организация процесса поставок товаров;
   3. Анализ данных о спросе на товары и эффективности работы;
   4. Возможность связаться с администраторами системы для решения вопросов или обратной связи.
3. Администраторы и менеджер:
   1. Настройка параметров и функциональности программного продукта в соответствии с требованиями организации;
   2. Отслеживание работы системы и производительности;
   3. Решение проблем и вопросов пользователей, возникающих в процессе работы с системой;
   4. Предоставление технической поддержки и консультаций;
   5. Мониторинг угроз безопасности и реагирование на них;
   6. Внедрение новых функций и улучшений на основе обратной связи пользователей.
4. Курьеры и служба доставки:
   1. Присвоение заказов курьерам с учетом местоположения и расписания работы;
   2. Возможность генерации отчетов о выполненных доставках и заработке;
   3. Возможность обратиться в службу поддержки для решения проблем или получения помощи;
   4. Функционал для учета времени доставки и оплаты за заказы.

**1.3. Описание исходной(входной) информации**

**Перечень исходной информации:** информация о продуктах, информация о ценах и акциях, данные о пользовательских предпочтениях, информация о заказах и доставке, данные о пользователя, информация о зонах доставки, финансовая информация, данные обратной связи и отзывы, печатная сводка, электронные отчеты, уведомления через приложение, онлайн-панель администратора.

**Перечень пользователей исходной информацией (подразделение и персонал):**

- Администраторы и менеджер:

Подразделение: отдел управления продуктами и заказами.

Персонал: администраторы сайта, менеджер по закупкам и продажам.

- Курьеры и служба доставки:

Подразделение: отдел доставки.

Персонал: курьеры.

- Пользователи (клиенты):

Подразделение: отдел обслуживания клиентов.

Персонал: клиенты, покупатели продуктов.

- Поставщик:

Подразделение: отдел закупок и снабжения.

Персонал: поставщик свежих продуктов и овощей.

**1.4.** **Описание результатной(выходной) информации**

**Перечень результатной информации:** подтверждение заказа, статус заказа, детализация заказа, уведомления о доставке, информация об оплате.

**Формы представления(документ) по каждой позиции перечня; примеры заполнения документов:**

- Подтверждение заказа:

Документ: электронное письмо с деталями заказа.

Пример заполнения: Заказ №12345 успешно оформлен. Детали заказа: Яблоки (3 кг), Морковь (1 кг), Бананы (2 кг), Адрес доставки: ул. Пушкина, д. 10, кв. 5, Сумма заказа: 1500 руб.

- Статус заказа:

Документ: страница с информацией о статусе заказа в личном кабинете.

Пример заполнения: Статус заказа №12345: В обработке.

- Уведомления о доставке:

Документ: SMS-сообщение с информацией о времени доставки и курьере.

Пример заполнения: Ваш заказ будет доставлен 20 марта с 14:00 до 16:00. Курьер: Иванов Иван.

**Периодичность и сроки представления:**

- Подтверждение заказа: генерируется автоматически сразу после оформления заказа пользователем;

- Статус заказа: обновляется в режиме реального времени и доступен в личном кабинете или мобильном приложении;

- Уведомления о доставке: отправляются за определенный период до времени доставки, например, за несколько часов;

- Информация об оплате: предоставляется немедленно после завершения транзакции оплаты заказа.

**Перечень пользователей результатной информации (подразделение и персонал):**

- Клиенты (покупатели):

Подразделение: отдел маркетинга и обслуживания клиентов.

Персонал: покупатели, которые делают заказы и ожидают информацию о них.

- Курьеры и служба доставки:

Подразделение: отдел и доставки.

Персонал: курьеры, ответственные за доставку заказов, которые нуждаются в информации о маршруте и деталях заказа.

- Администраторы и менеджер:

Подразделение: отдел управления продуктами и заказами.

Персонал: администраторы сайта и менеджер, которые могут отслеживать статус заказов и управлять процессом продаж.

**1.5.** **Описание используемой условно-постоянной информации**

**Перечень условно-постоянной информации (классификаторов, справочников, таблиц, списков, с указанием их полных наименований):**

- Классификаторы товаров:

Наименование: справочник категорий товаров.

Описание: иерархическая структура категорий товаров для организации каталога.

- Справочник адресов доставки:

Наименование: справочник адресов доставки.

Описание: список адресов, куда могут осуществляться доставки заказов.

- Таблица цен на товары:

Наименование: таблица цен на товары.

Описание: список товаров с их текущими ценами для корректного расчета стоимости заказов.

- Справочник поставщиков:

Наименование: справочник поставщиков.

Описание: информация о компаниях-поставщиках продуктов и овощей.

- Справочник сотрудников:

Наименование: справочник сотрудников.

Описание: информация о сотрудниках компании, их должностях и контактных данных.

**Формы представления:** таблицы и списки, графики и диаграммы, электронные отчеты.

**1.6. Эксплуатационные требования**

**Требования к применению (определяют качество пользовательского интерфейса, документации и учебных курсов):**

- Пользовательский интерфейс (UI): простота использования, эффективность, доступность;

- Документация: понятность и полнота, структурированность, актуальность;

- Учебные курсы: целевая аудитория, интерактивность, практические задания.

**Требования к производительности (накладывают ограничения на функциональные требования, задавая необходимые эффективность использования ресурсов, пропускную способность и время реакции):**

- Эффективное использование ресурсов;

- Пропускная способность;

- Время реакции;

- Ограничения на время выполнения операций.

**Требования к реализации (предписывает использование определенных стандартов, языков программирования, операционной среды и т.д.):**

- Стандарты и протоколы;

- Использование определенных языков программирования и технологий;

- Совместимость с определенной операционной средой;

- Безопасность и защита данных.

**Требования к надежности (обуславливают допустимые частоту и воздействие сбоев на работу ПП, а также возможности восстановления ПП после сбоев):**

* Устойчивость к сбоям;
* Минимизация воздействий сбоев на работу системы;
* Восстановление после сбоев.

**Требования к интерфейсу (определяют внешние сущности (т.е. пользователей и любые внешние устройства), с которыми может взаимодействовать система, и регламент этого взаимодействия:**

* Определение типов пользовательских интерфейсов и их функциональных возможностей;
* Взаимодействие с внешними устройствами;
* Регламент взаимодействия между системой и пользователями.

**2. Выбор стратегии разработки и модели жизненного цикла**

Таблица 1 **–** Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да | Нет | Нет | Нет |
| 2. | Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| 3. | Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Да |
| 4. | Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| 5. | Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |
| 6. | Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |
| 8. | Итого: | 3 | 3 | 4 | 4 | 4 | 4 |

Итог: на основе результатов заполнения табл. 1 подходящей является RAD, инкрементная, быстрого прототипирования и эволюционная модель.

Таблица 2 **–** Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории команды разработчиков проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | Да | Да |
| 2. | Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | Да |
| 3. | Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | Нет | Да | Нет | Нет |
| 5. | Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| 6. | Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Нет | Да | Да | Да |
| 7. | Итого: | 4 | 4 | 3 | 4 | 3 | 2 |

Итог: на основе результатов заполнения табл. 2 подходящими являются каскадная, v-образная, и инкрементная модель.

Таблица 3 **–** Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории коллектива пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да | Нет | Да | Нет | Да |
| 2. | Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет | Да | Да | Да |
| 3. | Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет | Да | Нет | Да | Нет |
| 4. | Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | Да | Да |
| 5. | Итого: | 1 | 1 | 2 | 2 | 3 | 1 |

Итог: на основе результатов заполнения табл. 3 подходящей является модель быстрого прототипирования.

Таблица 4 **–** Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Разрабатывается ли в проекте продукт нового для организации направления? | Нет | Нет | Нет | Да | Да | Да |
| 2. | Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | Нет | Нет |
| 3. | Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет | Да | Да | Да |
| 4. | Ожидается ли длительная эксплуатация продукта? | Да | Да | Нет | Да | Нет | Да |
| 5. | Необходим ли высокий уровень надежности продукта проекта? | Нет | Да | Нет | Да | Нет | Да |
| 6. | Предполагается ли эволюция продукта проекта в течение ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| 7. | Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет | Да | Да | Да |
| 8. | Является ли график сжатым? | Нет | Нет | Да | Да | Да | Да |
| 9. | Предполагается ли повторное использование компонентов? | Нет | Нет | Да | Да | Да | Да |
| 10. | Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? | Нет | Нет | Нет | Нет | Да | Да |
| 11. | Итого: | 5 | 6 | 4 | 6 | 4 | 6 |

Итог: на основе результатов заполнения табл. 4 подходящими являются v-образная, инкрементная и эволюционная модель.

Таблица 5 **–** Общие расчеты по таблицам 1,2,3,4

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № критерия | Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| 1. | Итого: | 13 | 14 | 13 | 16 | 14 | 13 |

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является инкрементная

**3. Руководство программиста**

**3.1 Создание сайта**

Для установки WordPress необходимо предварительно подготовить окружение. Скачайте OpenServer с официального сайта, установите его и настройте параметры. Выберите версию PHP, совместимую с WordPress, активируйте MySQL для работы с базой данных и настройте веб-сервер. После этого запустите OpenServer и через интерфейс откройте phpMyAdmin, чтобы создать новую базу данных, например, с именем site\_db, задав кодировку utf8mb4\_general\_ci. Далее скачайте последнюю версию WordPress с официального сайта и распакуйте её в папку domains внутри OpenServer, например, в директорию C:\OpenServer\domains\midland.

Следуя инструкциям мастера установки, укажите имя базы данных, логин (по умолчанию root) и оставьте поле пароля пустым, если настройки не изменялись.

После завершения установки WordPress настройте основные параметры сайта, такие как название, язык интерфейса и учётные данные администратора. Для расширения функционала установите нужные плагины через административную панель в разделе «Плагины». Например, для интернет-магазина подойдет WooCommerce, где можно настроить способы оплаты, доставки и валюту. На финальном этапе проведите тестирование сайта. Проверьте корректность отображения страниц, работу плагинов и шорткодов, функциональность интернет-магазина, а также поддержку нескольких языков интерфейса, если это предусмотрено. Такой подход гарантирует стабильную работу веб-приложения.

**3.2 Настройка функциональности и дизайна**

После установки WordPress начинается этап настройки дизайна и добавления функциональности. В первую очередь выберите и установите тему для сайта. Это можно сделать через раздел Внешний вид и Темы. Выберите тему из репозитория WordPress или загрузите свою кастомную тему в формате .zip. Активируйте выбранную тему. Для расширения возможностей сайта установите следующие плагины:

* Elementor — визуальный конструктор страниц с поддержкой drag-and-drop редактора;
* Elementor Header & Footer Builder — позволяет настраивать хедер и футер через Elementor;
* WooCommerce — плагин для создания интернет-магазина. Он добавляет возможность управления товарами, заказами и оплатами;
* Login/Signup Popup — плагин для создания личных кабинетов пользователей, авторизации и регистрации;

Для установки плагинов перейдите в раздел Плагины и Добавить новый, введите название плагина в поиске, установите и активируйте его.

**3.3 Редактирование страниц**

Создание и редактирование страниц осуществляется через административную панель WordPress. Чтобы добавить новую страницу, откройте раздел «Страницы» и выберите пункт «Добавить новую». Создайте необходимые разделы, такие как «О нас», «Контакты» или «Главная». Для редактирования страниц используйте редактор Elementor, который позволяет легко настраивать содержимое с помощью виджетов. В Elementor можно добавлять текстовые блоки, изображения, кнопки и другие элементы, а также изменять их параметры. Для упорядочивания содержимого используйте секции и колонки. Настройте внешний вид блоков, включая шрифты, цвета, размеры и отступы, чтобы обеспечить визуальную привлекательность и удобство для пользователей. Каждая страница должна иметь продуманную структуру и полезный контент, соответствующий её назначению.

**3.4 Настройка Header и Footer**

Для создания хедера и футера использован плагин Elementor Header & Footer Builder. В разделе «Внешний вид» откройте Header Footer Builder и создайте отдельные шаблоны для верхнего (хедера) и нижнего (футера) блоков сайта. В хедере разместите логотип, основное меню навигации и контактную информацию. В футере добавьте ссылки на важные страницы, контактные данные и иконки социальных сетей. Редактирование этих блоков выполняется через интерфейс Elementor аналогично работе с другими страницами. После сохранения шаблонов хедер и футер автоматически отобразятся на всех страницах сайта.

**3.5 Редактирование главной страницы сайта**

Главная страница сайта — это первое, что видит посетитель. Она должна быть информативной, привлекательной и удобной. Хорошо спроектированная главная страница помогает удержать внимание пользователя, создаёт положительное впечатление о вашем бизнесе и способствует конверсии.

Для редактирования откройте главную страницу в Elementor через раздел «Страницы». В верхней части разместите яркий заголовок, отражающий суть вашего предложения, и баннер с изображением, дополняющим текст. Используйте виджеты «Заголовок» и «Изображение».

Информационные блоки должны представлять основные услуги или преимущества. Разделите страницу на секции с текстами, иконками и изображениями. Если сайт использует WooCommerce, добавьте блок с товарами.

Для удобства пользователей добавьте функционал личного кабинета и регистрации через плагин Login/Signup Popup. Настройте адаптивность через вкладку «Адаптивность» в Elementor, чтобы сайт был удобен на любых устройствах. Используйте качественные изображения, чёткий текст и кнопки с призывами к действию, например «Подробнее», «Связаться» или «Купить».