МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Руководитель проекта  (подпись) Ерохина Е. А.  (дата) | УТВЕРЖДАЮ  ?? |

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

МОСКОВСКИЙ ИНСТИТУТ ЭЛЕКТРОНИКИ И МАТЕМАТИКИ

ИМ. А.Н.ТИХОНОВА

ДЕПАРТАМЕНТ КОМПЬЮТЕРНОЙ ИНЖЕНЕРИИ

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ ПРОЕКТА**

«Разработка программного обеспечения

«Веб-сервис Умное питание»

с устройством считывания штрих-кодов на Raspberry Pi»

Листов 7

Лидер проекта:

Комаров Александр Михайлович,

ДКИ МИЭМ им. А.Н.Тихонова НИУ ВШЭ,

курс 3, группа БИВ-161

Руководитель проекта:

Ерохина Елена Альфредовна,

старший преподаватель,

ДКИ МИЭМ им. А.Н.Тихонова НИУ ВШЭ

**Введение**

**Аннотация проекта**

Объектом разработки является веб-сайт, предоставляющий пользователям информацию о продуктах, блюдах, рецептах, диетах и устройство сканирования штрих-кодов. Зарегистрированные пользователи сайта будут иметь возможность добавлять в свой список продукты и блюда, а также создавать и добавлять свои собственные рецепты. При помощи штрих-код сканера информация о купленных продуктах будет загружаться на сайт. Кроме того, пользователям сайта будет предоставлена возможность следить за питанием при помощи различных диет. В качестве результата разработки выступит рабочий веб-сайт для персональных компьютеров и мобильных устройств, поддерживающий все современные веб-браузеры, а также устройство, позволяющие считывать данные о продуктах по штрих-коду. Уникальность проекта заключается в подключении и совместной работе технического устройства и веб-сервиса, без проводного подключения. Считывающий сканер должен подключаться к аккаунту пользователя при помощи сервера и привязываться к нему. Сканер и сопровождающее его ПО может быть полезно больным диабетом, спортсменам, а также людям, соблюдающим диету по медицинским показаниям, диетологам, их пациентам и др.

На основе полученного результата проектной работы в будущем возможна реализация версии приложения для мобильных устройств, с единым аккаунтом и базой данных и сканированием штрих-кодов продуктов при помощи камеры смартфона.

**Наименование проекта**

Разработка программного обеспечения «Веб-сервис Умное питание» с устройством считывания штрих-кодов на Raspberry Pi.

**Область применения**

Веб-приложение может быть использовано в повседневной жизни для упрощения учета покупки продуктов и приготовления блюд, а также медицинскими работниками при назначении диет пациентам.

Сканер и сопровождающее его ПО может быть полезно больным диабетом, спортсменам, а также людям, соблюдающим диету по медицинским показаниям, диетологам, их пациентам и др.

**Основания для разработки**

* Приказ № 6.18.1-01/0509-02 от 05.09.2016 «О введении в действие Положения о проектной, научно-исследовательской деятельности и практиках студентов Национального исследовательского университета «Высшая школа экономики».

**Назначение разработки**

Цель проекта состоит в создании веб-сайта, максимально удовлетворяющего потребности пользователя в выборе продуктов, блюд и т.д, а также устройства на Raspberry Pi, для считывания штрих-кодов продуктов. Сайт должен иметь оригинальный интерфейс и дизайн, работать надежно и стабильно на всех устройствах и во всех браузерах. Пользователь сможет сканировать купленные продукты при помощи штрих-код сканера и загружать данные о товарах на сайт. В качестве программной документации должна присутствовать инструкция по эксплуатации ПО.

**Состав задач**

* Создание макетов дизайна страниц сайта.
* Верстка страниц сайта.
* Реализация динамического пользовательского интерфейса.
* Проектирование БД.
* Написание серверной части.
* Тестирование ПО.
* Проектирование сканера штрих-кодов на Raspberry Pi.

**Требования к программе или программному изделию**

* **Требования к функциональным характеристикам**:

Веб-приложение должно содержать в себе следующий функционал:

* Предоставлять пользователю информацию о различных продуктах и рецептах.
* Вести учет продуктов и приготовленных блюд, находящихся в холодильнике.
* Автоматически пополнять списки покупок в соответствии с выбранными настройками.
* Давать возможность выбора диеты и предоставлять информацию о ней.
* Генерировать меню на день на основе цены, срока годности, калорийности и других характеристик продуктов.
* Давать возможность создания аккаунта пользователя.
* Заносить информацию о купленных продуктах в аккаунт на сайт при помощи устройства считывания штрих-кодов.
* Давать возможность создавать и добавлять свои собственные рецепты и продукты.
* **Требования к надежности**:

Ограничения использования веб-приложения напрямую связаны с используемой версией веб-браузера. Веб-сайт не будет поддерживаться в устаревших браузерах:

* Yandex Browser 15.12 и ниже.
* Google Chrome 52 и ниже.
* Internet Explorer 11 и ниже.
* Opera 37 и ниже.
* Safari 9.1 и ниже.
* Firefox 48 и ниже.
* **Условия эксплуатации**:

Условия эксплуатации распространяются на одноплатный компьютер Raspberry Pi. Для корректной работы устройства необходимы следующие условия:

* Температура окружающего воздуха в диапазоне температур от -25℃ до +80℃.
* Относительная влажность до 60%.
* Питание: 5-вольтовый адаптер, с силой тока не менее 2 А.
* **Требования к составу и параметрам технических средств**:

Необходимые ресурсы для реализации проекта:

* Компьютер с выходом в Интернет.
* Raspberry Pi 3 Model B.
* 4 Amp адаптер питания.
* 16GB Micro SD.
* USB сканнер штрих-кодов.
* **Требования к информационной и программной совместимости**:

Технологии и инструменты, используемые в проекте:

1. VSCode, Atom – текстовые редакторы.
2. Google Chrome или Mozilla Firefox – веб-браузеры.
3. Adobe Photoshop – обработка изображения, создание макетов страниц сайта.
4. HTML, CSS – верстка страниц сайта.
5. JS, JQuery – создание интерактивного динамического пользовательского интерфейса.
6. Beget – хостинг.
7. FileZilla – FTP клиент для переноса данных сайта на хостинг.
8. Bootstrap 4 – HTML/CSS фреймворк.
9. Vue.js – фронт-энд JS фреймворк.
10. Git – для контроля версий.
11. GitHub – веб-платформа для управления версиями проекта.
12. SASS – CSS Pre-processor.
13. Node.js – для создания серверной части.
14. MongoDB – для создания БД.
15. Express - веб-фреймворк для приложений Node.js.
16. Heroku - облачная PaaS-платформа.

При работе с сайтом, пользователь будет делать HTTP-запросы, которые в свою очередь будут обрабатываться сервером. В зависимости от вида запроса сервер будет совершать различные манипуляции с БД: показывать, добавлять, изменять и удалять данные.

Используя устройство ввода - сканнер штрих-кодов, пользователь сканирует штрих-код продукта, после чего данный код отправляется на сервер. Сервер находит в БД продукт с соответствующим кодом и добавляет пользователю на сайт.

* **Требования к транспортировке и хранению Raspberry Pi**:

Изделие должно храниться в складских помещениях, защищенных от воздействий атмосферных осадков, на стеллажах в упаковке изготовителя при отсутствии в воздухе паров кислот, щелочей и других веществ, вызывающих коррозию.

Транспортирование изделия разрешается в упаковке предприятия – поставщика всеми видами транспорта, за исключением негерметизированных отсеков самолета, без ограничения расстояния. Транспортирование упакованных изделий может производиться в крытых вагонах и автомашинах, трюмах судов и герметичных кабинах самолетов при температуре воздуха от -20℃ до +70℃.

**Требования к программной документации**

1. Отчет по проекту.
2. Инструкция по эксплуатации.

**Технико-экономические показатели**

Веб-приложение будет отличаться от аналогов своим многообразным функционалом и широкой направленности на большую аудиторию пользователей. В отличии от аналогичных разработок будет включать в себя новейшие технологии и оригинальный дизайн.

Кроме того, ни один подобный проект не содержит в себе устройства считывания штрих-кодов, которое абсолютно стационарно и подключается к сайту без провода.

**Стадии и этапы разработки**

**План работы над проектом:**

1. Этап 1
   1. Сформировать команду, распределить роли.
   2. Сформулировать проектное предложение.
   3. Разработать ТЗ.
   4. Выбрать инструментарий и обосновать его выбор.
   5. Записатт видеофрагмент и выступить с презентацией проекта.
2. Этап 2
   1. Тестирование первой версии программы:
      1. Проверка работы интерфейса сайта.
      2. Тестирование запросов к серверу.
      3. Тестирование прототипа сканера.
   2. Рабочая версия:
      1. Веб-сайт с минимальным функционалом.
      2. Сервер, отвечающий на основные запросы.
      3. Сканер, считывающий штрих-коды.
   3. Документация:
      1. Отчет о работе над проектом.
      2. Инструкция по эксплуатации.
3. Этап 3
   1. Бета-тестирование:
      1. Тестирование сайта на устойчивость, адаптивность и кроссбраузерность.
      2. Полное тестирование работы сервера.
      3. Тестирование работы сканера, его подключения к сайту.
   2. MVR (минимальный жизнеспособный продукт):
      1. Веб-приложение обладает минимальным функционалом, заявленным в ТЗ.
      2. Сервер работает стабильно, отвечая на запрос пользователей.
      3. Сайт размещен на хостинге, в общем доступе.
      4. Сканер подключается к сайту и распознает штрих-коды.
      5. Имеется минимальная техническая документация.
   3. Публичное тестирование:
      1. Тестирование на устойчивость к большому количеству пользователей на сайте.
      2. Тестирование сервера на множество одновременных запросов.
      3. Работа считывающего устройства с сайтом, размещенным в публичном доступе.
   4. Публичный доступ.

**Сроки:**

До 8 ноября: Видеофрагмент с обзором проекта.

До 15 ноября: Техническое задание по проекту, презентация по проекту.

15-31 января: Представление рабочей версии:

* Веб-приложение обладает минимальным функционалом.
* Сервер отвечает на запросы.
* Сканер считывает штрих-коды.

13-24 мая: MVR, публичная защита.

**Порядок контроля и приемки**

Для ПО испытания включают в себя следующие тесты:

* Генерация меню на основе истечения срока годности продуктов.
* Автоматическое заполнение списка покупок нужными продуктами.
* Правильный расчет калорийности блюда.

Испытания сканирующего устройства заключаются в успешном считывании штрих кода на различных поверхностях (пластик, бумага, металл и др.), а также в распознании поврежденного или нечеткого штрих-кода.