

Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б. Н. Ельцина» (УрФУ)

Институт радиоэлектроники и информационных технологий – РТФ

ОТЧЕТ

о проектной работе

по теме: Аналитика проекта по созданию бота с искусственным интеллектом в социальной сети Telegram

по дисциплине: Проектный практикум

Команда: ШрэксИнтертеймент

Тимлид: Власов Леонид РИ-140945

Аналитик: Дмитриенко Константин РИ-140945

Датасетер: Абзалов Эдуард РИ-140945

Бэкенд-разработчик: Ладыгин Никита РИ-140945

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc165369900)

[1. Целевая аудитория 5](#_Toc165369901)

[2. Определение проблемы 6](#_Toc165369902)

[3. Подходы к решению проблемы 8](#_Toc165369903)

[4. Анализ аналогов 10](#_Toc165369904)

[5. Календарный план проекта 12](#_Toc165369905)

[6. Сценарии использования 14](#_Toc165369906)

[7. Требования к продукту и к MVP 15](#_Toc165369907)

[8. Стек для разработки 18](#_Toc165369908)

[ЗАКЛЮЧЕНИЕ 19](#_Toc165369911)

ВВЕДЕНИЕ

**Актуальность**

В условиях активного внедрения цифровых технологий в повседневную жизнь всё больше людей стремятся следить за своим питанием, вести здоровый образ жизни и контролировать рацион. Особенно это актуально для молодежи, находящейся в учебной среде: школьников, студентов и преподавателей.

Однако большинство сервисов для подсчета калорий и анализа питания либо платные, сложные в использовании, либо не обеспечивают должной точности. Ручной ввод данных отнимает много времени и вызывает раздражение у пользователей. Telegram - является одним из самых популярных мессенджеров в России, и решение, интегрированное в него, обладает большим потенциалом охвата и вовлечения.

Проект FoodSherlock предлагает инновационное решение - Telegram-бот с элементами искусственного интеллекта, который способен по фотографии блюда автоматически определять его название и рассчитывать калорийность. Это простое и удобное решение для повседневного использования, особенно в условиях ограниченного времени и высокой нагрузки.

**Цель и задачи проекта**

Целью данного проекта является разработка Telegram-бота, способного распознавать блюда по изображению и предоставлять пользователю точную информацию о калорийности и составе пищи, с акцентом на простоту использования и доступность для образовательной аудитории. Для достижения этой цели необходимо решить следующие задачи:

1. **Изучить потребности целевой аудитории** - провести анализ студентов, школьников и преподавателей как потенциальных пользователей: какие у них есть боли и задачи в плане контроля питания, как они используют Telegram и чего ждут от подобных сервисов.
2. **Провести анализ существующих решений и конкурентов** - изучить Telegram-боты и мобильные приложения, выполняющие схожие функции, выявить их преимущества и недостатки, определить, какие особенности нужно сохранить, а какие - улучшить или избежать.
3. **Разработать требования к функциональности и интерфейсу бота** - определить ключевые функции MVP: загрузка фото, распознавание блюда, расчёт калорийности, персонализация под предпочтения пользователя. Сформировать технические и UX-требования с учётом специфики Telegram.
4. **Реализовать MVP-версию Telegram-бота и провести её тестирование** - собрать базу данных изображений, обучить модель, разработать интерфейс бота, внедрить базовые функции и протестировать их с реальными пользователями. Проверить точность распознавания и отклик системы.
5. **Оценить результаты и спланировать развитие проекта** - на основе тестирования и отзывов определить направления для доработки: улучшение модели, расширение функционала, добавление образовательных и социальных модулей.

Реализация этих задач позволит не только создать работающий прототип бота, но и обеспечить основу для его масштабирования, улучшения модели и интеграции с другими сервисами.

1. Целевая аудитория

Целевая аудитория проекта определяется по методике 5W Марка Шеррингтона:

* What (Что?)

Мы предлагаем Telegram-бота, который распознаёт еду по фотографии и выдаёт точные данные по калорийности и составу блюда. Это не просто информативный сервис, а интерактивный помощник, встроенный в привычную среду мессенджера.

* Who (Кто?)

Нашими пользователями являются:

* Студенты (18–25 лет) - активно используют Telegram, питаются вне дома, стремятся следить за здоровьем и расходами.
* Школьники (12–17 лет) и их родители - интересуются рационом в школьной столовой.
* Преподаватели и научные сотрудники (25–60 лет) - могут использовать бота в исследовательских целях или для организации питания на мероприятиях.
* Why (Почему?)

Потребность - быстро и без усилий получить информацию о питании. Особенно важна точность данных и удобство. Проблема - отсутствие точных, бесплатных и адаптированных под молодежную аудиторию решений.

* When (Когда?)

Бот может использоваться в любой момент: во время обеда, в столовой, кафе, дома, между парами - главное, чтобы была возможность сделать фото еды.

* Where (Где?)

Telegram - основная платформа, где проходит взаимодействие. Большинство точек входа — это студенческие и школьные чаты, группы в мессенджере, а также каналы, посвящённые ЗОЖ.

1. Определение проблемы

Разработка Telegram-бота для распознавания блюд по фотографии и расчета их калорийности сопровождается рядом специфических проблем, каждая из которых влияет как на точность конечного результата, так и на пользовательский опыт.

**Одна из основных проблем заключается в низком качестве входных данных.**

Пользователи часто присылают изображения блюд, сделанные при плохом освещении, в тени, с размытым фокусом или под неверным углом. Кроме того, распространены случаи загрузки перевёрнутых или частично обрезанных изображений, где блюдо плохо различимо. Всё это приводит к снижению точности распознавания и может вызывать неверные расчёты калорийности.

**Ограничения Telegram как платформы.**

Изображения, передаваемые через мессенджер, подвергаются сжатию, теряют в качестве и размере, что затрудняет анализ деталей. Telegram также не предоставляет предпросмотра изображения перед отправкой, из-за чего пользователи не всегда осознают, насколько качественным является отправленное фото.

**Ещё одна значимая проблема касается самой технологии компьютерного зрения.**

Современные модели компьютерного зрения, даже при хорошем обучении, слабо справляются с распознаванием однородных по текстуре блюд (например, каш, пюре, супов) или сложных композиций, включающих сразу несколько ингредиентов на одной тарелке. Распознавание также осложняется в случае национальных и авторских рецептов, не представленных в базах данных. Даже при корректном определении блюда существует риск ошибки в оценке порции и, как следствие, - в расчете калорийности, что особенно критично для пользователей, имеющих ограничения по питанию.

**Важной проблемой является нехватка качественных размеченных данных.**

Для обучения нейросетевой модели требуется большое количество изображений блюд с точной маркировкой (названием, составом, калорийностью и весом). Однако сбор, разметка и обновление таких данных - процесс очень трудоёмкий. При этом часть открытых датасетов не адаптирована под особенности российской кухни, что ограничивает точность модели в локальном контексте.

Таким образом, проект сталкивается сразу с несколькими уровнями проблем: техническими (качество изображений, ограничения Telegram), алгоритмическими (точность ИИ-модели), пользовательскими (ошибки ввода и завышенные ожидания). Все эти проблемы требуют комплексного подхода и продуманной системы решений. В противном случае бот не будет востребован или не сможет выполнять свои функции на должном уровне, что поставит под угрозу эффективность и жизнеспособность проекта.

1. Подходы к решению проблемы

Для преодоления указанных выше проблем в проекте FoodSherlock предусмотрен ряд решений, направленных на повышение точности, надёжности и удобства работы бота.

1. **Технические решения**

* Модель машинного обучения обучается с учётом искажений, возникающих в Telegram (сжатие, шум, тени). Это позволяет частично компенсировать потерю качества изображения.
* Применение предварительной обработки фото - автоориентация, коррекция яркости, кадрирование и увеличение контраста. Эти процедуры позволяют "улучшить" изображение ещё до анализа.
* Алгоритмы уточняющих вопросов - если бот не уверен в распознавании, он может задать уточняющий вопрос (например, “Это овсянка или пюре?”), тем самым увеличив точность без повторной загрузки.
* Ограничение области анализа на фотографии - позволяет сосредоточиться на ключевом объекте, исключая фон и случайные объекты.

1. **Пользовательские и UX-решения**

* Интерфейс с подсказками по съёмке: при первом использовании бот даёт инструкции: "Фотографируйте при хорошем освещении", "Снимайте блюдо сверху", "Избегайте упаковок".
* Интерактивная обратная связь: пользователь может уточнить, верно ли бот распознал блюдо, и при необходимости вручную скорректировать результат.

1. **Организационные решения**

* Краудсорсинговая база рецептов: пользователи смогут предлагать новые блюда, составы и поправки - что обеспечит рост и актуальность базы данных без чрезмерной нагрузки на команду.
* Открытый исходный код и документация: проект может быть интегрирован в образовательные платформы, где студенты смогут сами вносить вклад в развитие.

1. Анализ аналогов

Перед началом проектирования Telegram-бота FoodSherlock был проведён анализ существующих решений на рынке, выполняющих схожие функции. Основной целью анализа было выявление конкурентных преимуществ, слабых сторон аналогов и понимание пользовательских ожиданий.

Были рассмотрены три популярных Telegram-бота, которые также используют искусственный интеллект для анализа еды по фотографии: @mealschatbot, @Food\_scanner\_bot и @FoodAi. Каждый из них имеет определённый набор функций, а также характерные плюсы и минусы.

@mealschatbot предлагает достаточно широкий функционал: распознаёт популярные блюда, выводит примерную калорийность, позволяет вручную добавлять продукты и вести историю запросов. Однако у него отсутствует персонализация, а основная часть функций доступна только по подписке. Также бот часто ошибается с неочевидными или сложными блюдами, что снижает точность расчётов.

@Food\_scanner\_bot акцентирует внимание на разборе блюда по ингредиентам и даёт советы по замене продуктов на более здоровые аналоги. При этом он не предоставляет точных калорийных данных, ограничиваясь диапазонами. Также весь функционал является платным, что ограничивает доступность для широкой аудитории.

@FoodAi отличается самой высокой скоростью обработки фото (1–3 секунды), минималистичным дизайном и простотой использования. Однако он не рассчитывает калорийность блюд, работает с небольшой базой данных и не предоставляет возможности персонализации или работы с историей запросов.

На основании проведённого анализа можно сделать вывод, что несмотря на наличие нескольких Telegram-ботов с функцией распознавания еды, ни один из них не сочетает в себе одновременно: точный подсчет калорий, персонализацию, доступность без подписки и ориентацию на образовательную аудиторию. Это делает FoodSherlock уникальным решением, особенно для студентов и школьников, заинтересованных в контроле питания и получении научно обоснованной информации о составе пищи.

Таблица 1 – Анализ конкурентов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии оценивания | FoodSherlock | mealschatbot | Food\_scanner\_bot | FoodAi |
| Размерность базы данных | Большая | Ограниченная | Ограниченная | Большая |
| Подсчет калорий | Точное число | Примерное число | Диапазон | Нет |
| Распознавание по фото | Да | Да | Нет | Нет |
| История запросов | Планируется | Да | Да | Нет |
| Платность | Бесплатный | Частично | Полностью | Бесплатный |
| Скорость работы | Средняя | Средняя | Высокая | Очень высокая |

1. Календарный план проекта

Название проекта:

Руководитель проекта:

Таблица 1 – Календарный план

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **№** | **Название** | **Ответственный** | **Длительность** | **Дата начала** | **Временные рамки проекта** | | | |
| 1 нед | 2 нед | 3 нед | 4 нед |
| ***Анализ*** | | | | | | | | |
| *1.1* | *Определение проблемы* | Аналитик | 1 неделя | 03.03 |  |  |  |  |
| *1.2* | *Выявление целевой аудитории* | Аналитик | 1 неделя | 03.03 |  |  |  |  |
| *1.3* | *Конкретизация проблемы* | Аналитик | 1 неделя | 03.03 |  |  |  |  |
| *1.4* | *Подходы к решению проблемы* | Аналитик | 1 неделя | 03.03 |  |  |  |  |
| *1.5* | *Анализ аналогов* | Аналитик | 1 неделя | 06.03 |  |  |  |  |
| *1.6* | *Определение платформы и стека для продукта* | Разработчик | 1 неделя | 06.03 |  |  |  |  |
| *1.7* | *Формулирование требований к MVP продукта* | Аналитик | 1 неделя | 06.03 |  |  |  |  |
| *1.8* | *Определение платформы и стека для MVP* | Разработчик | 1 неделя | 06.04 |  |  |  |  |
| *1.9* | *Формулировка цели* | Аналитик | 1 неделя | 10.03 |  |  |  |  |
| *1.10* | *Формулирование требований к продукту* | Аналитик | 1 неделя | 10.03 |  |  |  |  |
| *1.11* | *Определение задач* | Аналитик | 1 неделя | 11.03 |  |  |  |  |
| ***Проектирование*** | | | | | | | | |
| *2.1* | *Архитектура системы (компоненты, модули системы)* | Дизайнер | 2 недели | 24.03 |  |  |  |  |
| *2.2* | *Разработка сценариев использования системы* | Дизайнер | 2 недели | 27.03 |  |  |  |  |
| *2.3* | *Прототипы интерфейсов* | Дизайнер | 2 недели | 29.03 |  |  |  |  |
| *2.4* | *Дизайн-макеты* | Дизайнер | 2 недели | 31.03 |  |  |  |  |
| *2.5* | *Архитектура системы (компоненты, модули системы)* | Дизайнер | 2 недели | 01.04 |  |  |  |  |
| ***Разработка*** | | | | | | | | |
| *3.1* | *Написание кода* | Разработчик | 1 месяц | 07.04 |  |  |  |  |
| *3.2* | *Тестирование приложения* | Разработчик | 1 месяц | 05.05 |  |  |  |  |
| ***Внедрение*** | | | | | | | | |
| *4.1* | *Оформление MVP* | Тимлид | 2 недели | 19.05 |  |  |  |  |
| *4.2* | *Внедрение MVP* | Тимлид | 2 недели | 23.05 |  |  |  |  |
| *4.3* | *Написание отчета* | Аналитик | 2 недели | 26.05 |  |  |  |  |
| *4.4* | *Оформление презентации* | Аналитик | 2 недели | 26.05 |  |  |  |  |
| *4.5* | *Подготовка к защите проекта* | Вся команда | 2 недели | 0 |  |  |  |  |

1. Сценарии использования

Use-case продукта:

1. Пользователь запускает бота в Telegram;

* Нажимает кнопку /start;
* Получает приветственное сообщение;

1. Пользователь отправляет фото блюда;

* Telegram Bot API передает сообщение с фото экземпляру бота;
* Aiogram-приложение (Python) получает сообщение и извлекает изображение;

1. Обработка изображения;

* Приложение отправляет изображение в Qwen API для анализа
* Qwen API выполняет:
  + Распознавание блюда (классификация)
  + Расчет калорийности
  + API возвращает структурированные данные:

1. Финальное сообщение;

* Пользователь получает краткую сводку;

1. Альтернативный сценарий при ошибке;

* Если распознавание не удалось (confidence < 60%):
  + «❌ Не удалось точно определить блюдо»

1. Требования к продукту и к MVP

Проект Telegram-бота FoodSherlock направлен на создание доступного, удобного и интеллектуального инструмента, который поможет пользователям определять состав и калорийность блюд по фотографии. Основной задачей команды на данном этапе стало создание минимально жизнеспособного продукта (MVP) - версии, включающей ключевые функции, достаточные для запуска, тестирования и сбора обратной связи.

**Общие требования к продукту**

Итоговый продукт должен быть простым в использовании, функциональным, визуально понятным и быстрым в работе. Он должен взаимодействовать с пользователем через Telegram, что определяет требования к интерфейсу: текстовые команды, быстрые ответы, лаконичные описания и инструкции.

Поскольку основная целевая аудитория - студенты и школьники, важно учесть:

* минимализм в интерфейсе;
* отсутствие необходимости регистрации;
* бесплатный доступ к базовым функциям;
* адаптацию под слабую сеть и средние по мощности устройства.

**Цель MVP**

Целью создания MVP является реализация основной идеи продукта - автоматического определения блюда по фотографии и расчёта его калорийности - в простой, но рабочей форме. Это позволит не только протестировать гипотезу, но и собрать реальные отзывы от пользователей для последующего улучшения и масштабирования проекта.

**Основной функционал MVP**

Минимальный жизнеспособный продукт должен включать следующие компоненты:

* Загрузка изображения блюда в чат

Пользователь отправляет фотографию своего блюда напрямую в чат-бот. Бот реагирует только на изображения, без дополнительных кликов или настроек.

* Распознавание блюда по фото

После получения изображения бот запускает модель компьютерного зрения, которая определяет название блюда. В случае низкой уверенности может быть задан уточняющий вопрос (например: “Это гречка или пюре?”).

* Расчёт калорийности и БЖУ

На основании распознанного блюда из базы данных подтягиваются данные о калорийности, белках, жирах и углеводах. Пользователю отображается полная карточка продукта: название, калории, БЖУ.

* Инструкции и базовые команды

В MVP реализуются команды: /start, /help, /info, которые дают информацию о возможностях бота, правилах съёмки и структуре ответа.

**Обработка ошибок и обратная связь**

Если фотография не подходит (пустая, нечёткая, не содержит еды), бот вежливо сообщает об этом и предлагает повторить попытку, предоставляя подсказки по улучшению изображения.

**Технические ограничения MVP**

* Максимальный размер изображения - до 20 МБ (ограничение Telegram).
* Работа только с JPEG и PNG-изображениями.
* Обработка одного блюда на фото - многокомпонентные тарелки пока не поддерживаются.
* Поддержка только самых популярных блюд (около 200 наименований в базе на старте).

**Возможности для масштабирования после MVP**

После запуска и сбора обратной связи MVP может быть дополнен следующими функциями:

* учёт диетических предпочтений (веган, безглютен, диабет);
* расширение базы данных до национальных и авторских блюд;
* добавление истории запросов и трекинга питания;
* создание пользовательских профилей;
* интеграция с образовательными сервисами и ЗОЖ-платформами.

1. Стек для разработки

Для разработки бота для распознавания блюд и подсчёта калорий выбран следующий стек технологий, который обеспечит качественную и эффективную реализацию проекта с учетом всех требований и функционала.

1. **Ultralytics YOLO 11** — последняя версия популярной модели «You Only Look Once» для реального-временной детекции и сегментации изображений с улучшенной точностью и скоростью. Модель оптимизирована под малые и крупные объекты благодаря усовершенствованному извлечению признаков и уменьшенному числу параметров, что ускоряет инференс без потери качества. Поддерживает режимы тренировки, валидации, инференса, экспорта и трекинга, что упрощает весь цикл разработки и развертывания модели.

2. **Python-telegram-bot**— синхронно-асинхронная библиотека на чистом Python для взаимодействия с Telegram Bot API, совместимая с Python 3.9+. Предоставляет высокоуровневые классы (Updater, Dispatcher, Handler) и удобные методы для обработки команд, inline-кнопок, медиа и колбэков.

3. **PyTorch** — оптимизированная библиотека для работы с тензорами и построения нейросетей на GPU/CPU, активно используемая в исследованиях и промышленности. Обеспечивает гибкий интерфейс для определения архитектур моделей, автоматического дифференцирования и эффективного управления вычислительными графами. Имеет богатую экосистему (TorchScript, TorchServe и др.), что упрощает деплой и интеграцию моделей в продакшн-среду.

4. **Torchvision** - включает популярные датасеты, предобученные архитектуры и стандартные трансформации изображений, упрощающие подготовку данных и дообучение моделей компьютерного зрения. Поддерживает такие операции, как изменение размера, нормализация, аугментации (RandomCrop, Flip и т. д.) и загрузку данных через DataLoader, что ускоряет конвейер подготовки данных для инференса и тренировки.

ЗАКЛЮЧЕНИЕ

Проект Telegram-бота **FoodSherlock** находится на завершающем этапе концептуальной и проектной разработки. Его основная цель - создать инструмент, позволяющий пользователям автоматически определять блюда по фотографии и получать точную информацию о калорийности и составе пищи. Бот ориентирован в первую очередь на студентов, школьников, преподавателей и их родителей, которым важно контролировать рацион в условиях высокой занятости и ограниченного времени.

**Основные выводы:**

Созданная концепция проекта позволяет решить актуальную задачу - упростить процесс анализа питания с помощью технологий компьютерного зрения и машинного обучения. В отличие от существующих аналогов, FoodSherlock делает упор на точность, персонализацию и бесплатную доступность. Его можно использовать в повседневной жизни, в образовательной среде и для научных целей, связанных с изучением пищевого поведения.

Целевая аудитория проекта была тщательно проанализирована и сегментирована. На основе этого анализа были сформулированы требования к продукту и его минимально жизнеспособной версии (MVP), включающей распознавание блюд, расчёт калорий и базовые команды Telegram-бота. Учитывались реальные ограничения Telegram, поведенческие особенности пользователей и особенности интерфейсов мессенджеров.

**Функциональные особенности FoodSherlock:**

* Упрощённое взаимодействие через Telegram без установки отдельных приложений.
* Автоматическое распознавание блюд по фотографии с помощью нейросетей.
* Расчёт калорийности и БЖУ по результатам анализа.

**Перспективы развития:**

Таким образом, цели и задачи, поставленные в начале проектной работы, были полностью достигнуты. Выполнение проекта позволило команде применить знания в области анализа потребностей, проектирования интерфейсов, исследования рынка, а также в области современных технологий - машинного обучения и компьютерного зрения. Разработанный концепт Telegram-бота FoodSherlock обладает потенциалом дальнейшего развития.

На следующем этапе проект может быть дополнен функциональностью по ведению истории питания, интеграцией с системами учёта питания в вузах и школах, расширением языковой поддержки, а также реализацией веб-версии. Потенциально бот может использоваться как вспомогательный инструмент в проектах по популяризации здорового образа жизни, а также в исследованиях пищевого поведения студентов.

В целом, реализация проекта FoodSherlock подтверждает актуальность применения технологий машинного обучения и ИИ в повседневной жизни. Разработка позволяет продемонстрировать не только технические знания команды, но и умение выявлять реальные проблемы пользователей и проектировать решения, ориентированные на практическое применение.

**Положительные стороны проекта:**

* Простота и доступность - бот работает на базе привычного Telegram.
* Высокий уровень потенциальной полезности в образовательной среде.
* Перспектива масштабирования и интеграции с цифровыми платформами.
* Возможность открытого развития сообщества и научного применения.

**Отрицательные стороны:**

* На старте возможны ошибки в распознавании из-за ограниченной базы данных и качества фото.
* Требуется дальнейшее обучение модели и расширение функционала для работы с редкими блюдами.
* Пользователи могут предъявлять завышенные ожидания к точности, что требует грамотного UX.

Таким образом, проект Telegram-бота FoodSherlock стал важным шагом в освоении проектной деятельности, применении современных технологий и командной работе. Он позволил не только разработать социально значимый и практически применимый цифровой продукт, но и получить ценный опыт анализа аудитории, формирования требований, проработки MVP и поиска решений реальных пользовательских проблем