**Таблица целей и критериев достижения**

| **Цель** | **Критерии достижения** |
| --- | --- |
| Достигнуть повышения удобства и скорости поиска рецептов для пользователей посредством реализации системы подбора рецептов по имеющимся ингредиентам. | 1. Пользователь может получить подборку релевантных рецептов на основе указанных ингредиентов в течение 3 секунд. 2. Система успешно возвращает результаты поиска для 95% запросов пользователей. |
| Обеспечить персонализацию опыта использования приложения посредством добавления функционала авторизации и сохранения предпочтений пользователя. | 1. Зарегистрированный пользователь может войти в свою учетную запись и получить доступ к сохраненным рецептам. 2. Успешная авторизация пользователя происходит менее чем за 2 секунды. |
| Повысить вовлеченность пользователей благодаря внедрению возможности добавления рецептов в избранное и публикации собственных рецептов. | 1. Пользователь может добавить любой рецепт в избранное одним кликом, и он будет доступен в личном кабинете. 2. Пользователь может опубликовать рецепт, который становится доступным для других пользователей. |
| Повысить качество контента приложения посредством реализации системы отзывов и рейтингов для рецептов. | 1. 80% рецептов имеют отзывы или рейтинги от пользователей спустя месяц после запуска системы. 2. Средний рейтинг рецептов обновляется в режиме реального времени. |
| Увеличить количество активных пользователей благодаря реализации интуитивного интерфейса с фильтрацией рецептов по времени приготовления, количеству порций и калорийности. | 1. Количество ежедневных активных пользователей увеличивается на 20% через 3 месяца после запуска. 2. Пользователь может настроить фильтры и получить релевантные результаты за 2–3 секунды. |

Эти цели связаны с бизнес-задачей создания удобного и востребованного инструмента для поиска рецептов, способного увеличить аудиторию пользователей и их вовлеченность.

**1.3 Область применения**

Разрабатываемое приложение предназначено для использования в следующих информационных системах и технологических средах:

**Список информационных систем:**

1. **Веб-приложение для настольных и мобильных браузеров**
   * Доступ к функционалу через браузеры (Google Chrome, Mozilla Firefox, Safari, Edge).
   * Progressive Web App (PWA) для обеспечения адаптивности и оффлайн-режима.
2. **Серверная инфраструктура**
   * Хостинг API и баз данных на облачных сервисах (например, AWS, Google Cloud Platform, Microsoft Azure).

**Используемые технологии:**

**СУБД**

* **MongoDB**:
  + Документоориентированная база данных для хранения информации о рецептах и пользователях.
  + Позволяет быстро обрабатывать запросы к данным, что важно для фильтрации и поиска.

**Frontend-фреймворк**

* **React.js**:
  + Для создания адаптивного пользовательского интерфейса.
  + Обеспечивает высокую производительность и удобство работы с компонентами.

**Backend-фреймворк**

* **Node.js с использованием Express.js**:
  + Разработка серверного API для обработки запросов от пользователей.
  + Обеспечение взаимодействия с базой данных и сторонними API (например, Spoonacular или Edamam).

**Дополнительные технологии**

1. **JWT (JSON Web Tokens)** — для обеспечения безопасности и авторизации пользователей.
2. **Axios или Fetch API** — для выполнения запросов к сторонним API и серверу.
3. **Cloudinary или AWS S3** — для хранения изображений рецептов.
4. **Redis** — для кэширования часто запрашиваемых данных (например, популярных рецептов).

**Платформы интеграции**

* Разработка приложения с возможностью интеграции с социальными сетями (Facebook, Instagram) для упрощения регистрации и обмена рецептами.
* Возможность расширения в виде мобильного приложения на основе React Native.

Приложение предназначено для конечных пользователей, которые хотят находить рецепты, публиковать свои собственные и делиться ими с другими, используя удобную и быструю технологическую платформу.

1.4 Описание бизнес-процессов

[*IDEF\_0*](https://ru.wikipedia.org/wiki/IDEF0)

**2. Требования к изменяемой подсистеме**

**2.1 Требования к функционалу**

**2.1.1 Общие требования к функционалу**

Разрабатываемое приложение предназначено для пользователей, которые хотят находить рецепты на основе имеющихся ингредиентов, добавлять их в избранное, оставлять отзывы, публиковать свои собственные рецепты и фильтровать их по различным параметрам.

**Бизнес-логика:**

1. **Поиск рецептов по ингредиентам:**  
   Пользователь вводит список имеющихся ингредиентов, система подбирает соответствующие рецепты на основе данных из базы.
2. **Фильтрация рецептов:**  
   Пользователь может уточнять результаты поиска по следующим параметрам:
   * Время приготовления.
   * Количество порций.
   * Калорийность.
3. **Регистрация и авторизация:**  
   Пользователь регистрируется для доступа к персонализированным функциям, таким как сохранение рецептов и создание собственных.
4. **Работа с избранным:**  
   Авторизованный пользователь может добавлять рецепты в избранное и удалять их при необходимости.
5. **Публикация рецептов:**  
   Зарегистрированный пользователь может добавлять свои рецепты, которые затем проходят модерацию перед публикацией.
6. **Система отзывов и рейтингов:**  
   Пользователи могут оставлять отзывы и оценивать рецепты. Средний рейтинг рассчитывается автоматически.

**2.1.2 Поиск рецептов по ингредиентам**

**Описание:**

* Пользователь вводит список ингредиентов через строку поиска (например, «картофель, курица, сыр»).
* Система возвращает список рецептов, содержащих указанные ингредиенты.

**Логика обработки:**

1. Сравнение введенных ингредиентов с данными в базе рецептов.
2. Фильтрация рецептов, где хотя бы 80% ингредиентов совпадают.
3. Сортировка результатов по:
   * Уровню соответствия ингредиентов.
   * Времени приготовления.

**Алгоритм:**

1. Пользователь вводит ингредиенты.
2. Система отправляет запрос в базу данных.
3. Возвращаются рецепты, которые удовлетворяют критериям.
4. Результаты отображаются в интерфейсе с возможностью дальнейшей фильтрации.

**2.1.3 Фильтрация рецептов**

**Описание:**

Пользователь может уточнять результаты поиска, задав параметры фильтрации.

**Детализация:**

1. **Время приготовления** — от 10 до 120 минут.
   * Параметры выбираются из предустановленных диапазонов (например, «до 30 минут», «до 1 часа»).
2. **Количество порций** — от 1 до 10.
   * Указывается точное число.
3. **Калорийность** — от 100 до 2000 ккал.
   * Выбирается с помощью ползунка или числового ввода.

**2.1.4 Регистрация и авторизация**

**Описание:**

Пользователь регистрируется с помощью электронной почты и пароля. Вход в систему осуществляется через логин и пароль или через социальные сети.

**Логика обработки:**

1. Регистрация:
   * Проверка уникальности email.
   * Хэширование пароля.
2. Авторизация:
   * Проверка соответствия email и пароля.
   * Генерация JWT для идентификации пользователя в сессии.

**2.1.5 Работа с избранным**

**Описание:**

Пользователь может добавлять рецепты в избранное, просматривать их в личном кабинете и удалять.

**Логика обработки:**

1. Добавление рецепта в избранное:
   * Кнопка «Добавить в избранное» передает идентификатор рецепта на сервер.
   * Идентификатор сохраняется в поле «favorites» пользователя в базе данных.
2. Удаление из избранного:
   * Аналогичная логика с удалением записи из базы.

**2.1.6 Публикация рецептов**

**Описание:**

Авторизованный пользователь может добавлять собственные рецепты.

**Детализация:**

1. Форма добавления рецепта:
   * Название рецепта.
   * Ингредиенты (список).
   * Шаги приготовления.
   * Время, порции, калорийность.
2. Проверка данных:
   * Валидация на стороне клиента и сервера.
3. Публикация:
   * Рецепт отправляется на модерацию. После подтверждения модератором он становится доступным другим пользователям.

**2.1.7 Система отзывов и рейтингов**

**Описание:**

Пользователь может оставлять отзывы и ставить оценки рецептам.

**Логика обработки:**

1. Добавление отзыва:
   * Поле для текста отзыва и выбора оценки (1–5 звезд).
   * Сохранение данных в базе с привязкой к идентификатору рецепта и пользователя.
2. Обновление среднего рейтинга рецепта:
   * Автоматический пересчет на основе всех оценок.

**2.2 Технические требования**

**2.2.1 Регламентное задание: Обновление базы данных рецептов из стороннего API**

**Описание функциональности:**

* Автоматическая синхронизация базы данных рецептов с внешним API (например, Spoonacular или Edamam) для добавления новых рецептов и обновления информации о существующих.
* Обновление выполняется ежедневно для поддержания актуальности данных.

**Параметры и расписание запуска:**

* Периодичность: 1 раз в сутки (например, в 02:00).
* Расписание задается через Cron или аналогичный планировщик задач.

**Этапы выполнения:**

1. Подключение к API и авторизация с использованием API-ключа.
2. Получение списка новых рецептов, измененных за последние 24 часа.
3. Сравнение полученных данных с текущей базой рецептов по идентификаторам.
4. Добавление новых рецептов в базу MongoDB.
5. Обновление информации в существующих записях (название, ингредиенты, рейтинг, время приготовления и т.д.).

**Принцип организации лога:**

* Создание логов в текстовом формате с фиксацией следующих данных:
  1. Время запуска регламентного задания.
  2. Общее количество полученных рецептов.
  3. Количество новых добавленных записей.
  4. Количество обновленных записей.
  5. Время завершения задания.
* Пример записи в лог:
* [2024-12-16 02:00:00] Начало обновления базы данных рецептов.
* [2024-12-16 02:01:20] Получено 500 рецептов из API.
* [2024-12-16 02:01:30] Добавлено 300 новых рецептов, обновлено 200 записей.
* [2024-12-16 02:02:00] Завершено. Обработка заняла 2 минуты.

**Контрольные процедуры проверки:**

1. Ежедневная проверка успешности выполнения задания:
   * Проверка наличия обновленных данных в базе.
   * Сравнение времени выполнения задания с ожидаемым расписанием.
2. Уведомления об ошибках:
   * При сбое задания (например, отсутствие соединения с API) отправляется email-уведомление ответственному администратору с описанием ошибки.
3. Тестовый запуск:
   * Перед внедрением задания проводится тестирование на выборке данных, чтобы проверить корректность добавления и обновления рецептов.

**2.2.2 Регламентное задание: Удаление устаревших данных из базы**

**Описание функциональности:**

* Удаление рецептов, которые не обновлялись и не использовались пользователями более 1 года.
* Оптимизация объема базы данных для повышения производительности.

**Параметры и расписание запуска:**

* Периодичность: 1 раз в месяц (например, каждое первое число в 03:00).

**Этапы выполнения:**

1. Сканирование базы данных для поиска рецептов с датой последнего обновления > 12 месяцев назад.
2. Удаление найденных записей.
3. Запись информации о количестве удаленных рецептов в лог.

**Принцип организации лога:**

* Фиксация данных:
  1. Время запуска задания.
  2. Количество удаленных записей.
  3. Время завершения.
* Пример записи в лог:
* [2024-12-16 03:00:00] Начало очистки базы данных.
* [2024-12-16 03:01:00] Удалено 1000 устаревших рецептов.
* [2024-12-16 03:01:10] Завершено. Обработка заняла 1 минуту 10 секунд.

**Контрольные процедуры проверки:**

1. Проверка, что рецепты с датой последнего обновления > 12 месяцев отсутствуют в базе.
2. Анализ объема освободившегося места в базе данных.
3. Тестирование задания на тестовой базе перед запуском в продуктивной среде.

Если нужно уточнить или добавить еще регламентные задания, например, для расчета рейтингов или кэширования данных, дайте знать!

**2.3 Требования к ролям**

В рамках данного технического задания предполагается создание и изменение ролей для разграничения прав доступа пользователей к функционалу приложения.

**2.3.1 Список ролей**

1. **Пользователь**
   * **Описание:** Базовая роль для зарегистрированных пользователей приложения.
   * **Уровень доступа:**
     + Просмотр рецептов.
     + Поиск рецептов по ингредиентам.
     + Использование фильтров.
     + Добавление рецептов в избранное.
     + Публикация собственных рецептов.
     + Добавление отзывов и оценок.
     + Просмотр и управление избранным.
   * **Ограничения:**
     + Нет доступа к функциям модерации рецептов.
2. **Модератор**
   * **Описание:** Роль для сотрудников, ответственных за проверку и публикацию пользовательских рецептов.
   * **Уровень доступа:**
     + Просмотр списка рецептов, отправленных на модерацию.
     + Одобрение или отклонение рецептов.
     + Редактирование пользовательских рецептов перед публикацией.
   * **Ограничения:**
     + Нет доступа к изменению системных параметров приложения и пользовательских данных.
3. **Администратор**
   * **Описание:** Роль для управляющего персонала с полным доступом к функционалу приложения.
   * **Уровень доступа:**
     + Полный доступ к базе данных рецептов, пользователей и отзывов.
     + Управление ролями и правами доступа.
     + Настройка параметров регламентных заданий.
     + Удаление и редактирование любых данных (рецептов, отзывов, пользователей).
   * **Ограничения:**
     + Нет ограничений.

**2.3.2 Принципы разграничения доступа**

* Все действия с объектами (например, рецепты, отзывы, избранное) будут проверяться на уровне API с учетом роли текущего пользователя.
* Каждая роль привязана к уникальным уровням доступа:
  + Пользователь может взаимодействовать только с собственными данными (например, добавлять свои рецепты).
  + Модератор ограничен функциями, связанными с модерацией контента.
  + Администратор имеет доступ к управлению всеми объектами и пользователями.

**2.3.3 Изменяемые объекты и роли**

* **Рецепты:**
  + Роль «Модератор» добавляется для работы с модерацией рецептов.
* **Пользователи:**
  + Создание ролей «Пользователь» и «Администратор».
  + Реализация разграничения доступа для управления личными данными и данными других пользователей.

Если изменения в ролях не требуются в будущем, можно будет расширять этот раздел. Пока описанный перечень соответствует целям текущего проекта.

**2.4 Требования к интерфейсам**

В рамках данного технического задания предполагается разработка и изменение интерфейсов приложения для обеспечения удобного доступа к функционалу.

**2.4.1 Основные интерфейсы**

1. **Главная страница**
   * **Раздел:** Пользовательский интерфейс.
   * **Описание:**
     + Строка поиска для ввода ингредиентов.
     + Кнопка для доступа к расширенным фильтрам.
     + Список рекомендованных рецептов с предварительным просмотром (название, время приготовления, рейтинг).
   * **Группа меню:** Главная страница приложения.
2. **Страница поиска рецептов**
   * **Раздел:** Пользовательский интерфейс.
   * **Описание:**
     + Результаты поиска с краткой информацией о рецептах (название, изображение, рейтинг, время приготовления).
     + Фильтры для уточнения поиска (время приготовления, количество порций, калорийность).
     + Сортировка по различным критериям (например, по рейтингу).
   * **Группа меню:** Поиск рецептов.
3. **Личный кабинет**
   * **Раздел:** Пользовательский интерфейс.
   * **Описание:**
     + Раздел «Избранное» с сохраненными рецептами.
     + Раздел «Мои рецепты» с опубликованными пользователем рецептами и кнопкой для добавления нового рецепта.
     + Настройки учетной записи.
   * **Группа меню:** Личный кабинет.
4. **Форма добавления рецепта**
   * **Раздел:** Пользовательский интерфейс.
   * **Описание:**
     + Поля ввода:
       - Название рецепта.
       - Ингредиенты (список с автодополнением).
       - Шаги приготовления (возможность добавления фото).
       - Время приготовления, количество порций, калорийность.
     + Кнопка «Опубликовать».
   * **Группа меню:** Личный кабинет > Мои рецепты.
5. **Страница модерации рецептов**
   * **Раздел:** Интерфейс модератора.
   * **Описание:**
     + Таблица с неопубликованными рецептами (название, дата добавления, автор).
     + Кнопки для подтверждения публикации или отклонения рецепта.
     + Форма для редактирования рецепта перед публикацией.
   * **Группа меню:** Панель модератора.
6. **Панель администратора**
   * **Раздел:** Интерфейс администратора.
   * **Описание:**
     + Раздел «Управление пользователями» с возможностью редактировать роли и права.
     + Раздел «Настройки приложения» для изменения параметров работы (например, расписания регламентных заданий).
     + Раздел «Просмотр логов» для анализа выполнения задач.
   * **Группа меню:** Администратор.

Если изменений в интерфейсе для какого-то функционала не предусмотрено, в соответствующих разделах указано «Изменений не требуется».

**2.4 Требования к интерфейсам**

В рамках данного технического задания предполагается создание и изменение следующих интерфейсов для обеспечения удобного доступа к основному функционалу приложения.

**2.4.1 Наименование интерфейсов и их расположение**

1. **Главная страница**
   * **Наименование интерфейса:** Главная страница.
   * **Раздел/подсистема:** Пользовательский интерфейс.
   * **Группа меню рабочего стола:** Главная.
   * **Описание:**
     + Строка поиска рецептов по ингредиентам.
     + Быстрый доступ к популярным и рекомендованным рецептам.
     + Кнопки для перехода в фильтры и личный кабинет.
2. **Результаты поиска**
   * **Наименование интерфейса:** Поиск рецептов.
   * **Раздел/подсистема:** Пользовательский интерфейс.
   * **Группа меню рабочего стола:** Поиск.
   * **Описание:**
     + Отображение рецептов, соответствующих введенным ингредиентам и фильтрам.
     + Панель сортировки (по рейтингу, времени приготовления и др.).
3. **Личный кабинет пользователя**
   * **Наименование интерфейса:** Личный кабинет.
   * **Раздел/подсистема:** Пользовательский интерфейс.
   * **Группа меню рабочего стола:** Личный кабинет.
   * **Описание:**
     + Раздел «Избранное» для сохраненных рецептов.
     + Раздел «Мои рецепты» для управления собственными рецептами.
     + Настройки учетной записи.
4. **Добавление нового рецепта**
   * **Наименование интерфейса:** Форма добавления рецепта.
   * **Раздел/подсистема:** Пользовательский интерфейс.
   * **Группа меню рабочего стола:** Личный кабинет > Мои рецепты.
   * **Описание:**
     + Поля для ввода названия, ингредиентов, шагов приготовления, времени и порций.
     + Возможность загрузки изображений рецепта.
     + Кнопка «Опубликовать».
5. **Интерфейс модератора**
   * **Наименование интерфейса:** Панель модерации.
   * **Раздел/подсистема:** Интерфейс модератора.
   * **Группа меню рабочего стола:** Модерация.
   * **Описание:**
     + Таблица с рецептами, ожидающими проверки.
     + Кнопки для одобрения, редактирования или отклонения рецептов.
     + Возможность просмотра деталей рецепта перед публикацией.
6. **Интерфейс администратора**
   * **Наименование интерфейса:** Панель администратора.
   * **Раздел/подсистема:** Администратор.
   * **Группа меню рабочего стола:** Администрирование.
   * **Описание:**
     + Раздел для управления ролями и доступами пользователей.
     + Просмотр логов выполнения регламентных заданий.
     + Настройка системных параметров и расписания заданий.

Если некоторые разделы или функционал в дальнейшем не потребуют изменения интерфейсов, можно будет указать для них «Изменений не требуется».

**2.5 Требования к миграции и изменению потоков данных**

В рамках данного технического задания изменения миграции данных и потоков данных не предполагаются.

**Обоснование**

1. **Хранение данных:**
   * Все данные приложения (рецепты, ингредиенты, пользователи, отзывы) хранятся централизованно в базе данных **MongoDB**.
   * Миграция между базами данных или изменение текущих потоков обмена не предусмотрено.
2. **Потоки данных:**
   * Интеграция со сторонними API для получения данных о рецептах будет осуществляться непосредственно в MongoDB.
   * Обмен между внешними источниками данных и базой будет проводиться через единый импортный процесс, без распределенной миграции объектов.
3. **Изменение нагрузки:**
   * В проекте нет необходимости обмена данными между различными базами.
   * Следовательно, дополнительная нагрузка на регламентные процессы и планы обмена отсутствует.

**Заключение**

В связи с вышесказанным данный раздел не требует изменений.

# 3 План внедрения

*Примечание. В данном разделе необходимо перечислить действия, понятные на этапе составления ТЗ, которые потребуется выполнить на этапе внедрения (напоминание о том, что понятно на этапе анализа, но может быть забыто по истечению срока разработки и тестирования).*

*Здесь можно перечислить заинтересованных лиц, кого необходимо оповестить на этапе внедрения функционала.*

*Для внешних инструментов здесь можно указать правила использования внешнего инструмента и то, что встраивать его не требуется.*

*Информация, которую содержит данный раздел, поможет составить итоговый план внедрения, который должен оформляться как отдельный артефакт в рамках проработки запроса.*