СОГЛАСОВАНО УТВЕРЖДАЮ

Руководитель ВКР Студент группы 21202

\_\_\_\_\_\_\_\_\_\_/ Г. Э. Яхъяева \_\_\_\_\_\_\_\_\_\_/ К. П. Кондренко

02.12.2024 02.12.2024

СЕРВЕРНАЯ ЧАСТЬ ИНДИВИДУАЛЬНОГО ПОМОЩНИКА ДЛЯ ПОДБОРА КОСМЕТИЧЕСКИХ СРЕДСТВ «GLAMCHECK»

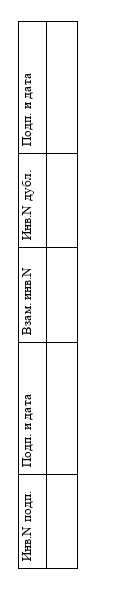
**Техническое задание**

**ЛИСТ УТВЕРЖДЕНИЯ**

**A.B.00001-01 90 01-1-ЛУ**

Представители

предприятия-разработчика:

 Начальник отдела

СОГЛАСОВАНО \_\_\_\_\_\_\_\_/ К.П. Кондренко

Руководитель ВКР 02.12.2024

\_\_\_\_\_\_\_\_\_\_/ Г. Э. Яхъяева

02.12.2024 Руководитель разработки

\_\_\_\_\_\_\_\_/ К.П. Кондренко

02.12.2024

Исполнитель

\_\_\_\_\_\_\_\_/ К.П. Кондренко

02.12.2024

Нормоконтролер

\_\_\_\_\_\_\_\_/ К.П. Кондренко

02.12.2024

2024

П

УТВЕРЖДЕНО

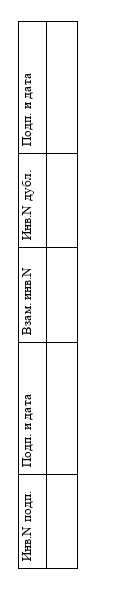
A.B.00001-01 90 01-1-ЛУ

СЕРВЕРНАЯ ЧАСТЬ ИНДИВИДУАЛЬНОГО ПОМОЩНИКА ДЛЯ ПОДБОРА КОСМЕТИЧЕСКИХ СРЕДСТВ «GLAMCHECK»

**Техническое задание**

**A.B.00001-01 90 01-1-ЛУ**

**Бумажный носитель**

**Листов 25**

2024

П

**СОДЕРЖАНИЕ**

[**Определения, обозначения, сокращения 3**](#_heading=h.ple2di3k2m3z)

[**1. Введение 5**](#_heading=h.vn7t8s35acj2)

[1.1. Наименование программы 5](#_heading=h.1fob9te)

[1.2. Краткая характеристика области применения 5](#_heading=h.7a7il7xm2xa3)

[**2. Основания для разработки 6**](#_heading=h.3znysh7)

[2.1. Основание для проведения разработки 6](#_heading=h.2et92p0)

[**3. Назначение разработки 7**](#_heading=h.1t3h5sf)

[3.1. Функциональное назначение 7](#_heading=h.4d34og8)

[3.2. Эксплуатационное назначение 7](#_heading=h.2s8eyo1)

[3.3. Целевая аудитория 7](#_heading=h.17dp8vu)

[**4. Требования к программе или программному изделию 9**](#_heading=h.lnxbz9)

[4.1. Требования к функциональным характеристикам 9](#_heading=h.hs9vdjgieygg)

[4.1.1. Регистрация пользователей 9](#_heading=h.lnvcxqu1n78j)

[4.1.2. Аутентификация пользователей 9](#_heading=h.1efi1jnpu1um)

[4.1.3. Анализ состава косметического средства 10](#_heading=h.4nw0f7lqhk1l)

[4.1.4. Поиск косметических средств 10](#_heading=h.lnvcxqu1n78j)

[4.1.5. Подбор косметических средств 11](#_heading=h.lnvcxqu1n78j)

[4.1.6. Логирование 12](#_heading=h.8ghvjqoze8vc)

[4.2. Требования к надежности 13](#_heading=h.qlq9r4cw5pao)

[4.2.1. Требования к обеспечению надежного функционирования программы 13](#_heading=h.10ij9yoq29ww)

[4.2.2. Время восстановления после отказа 13](#_heading=h.4a7ojto0fbwy)

[4.2.3. Отказы из-за некорректных действий оператора 13](#_heading=h.ovyogpyb3rfu)

[4.2.4. Организационно-технические мероприятия 13](#_heading=h.7c3gdbz1g7xh)

[4.3. Условия эксплуатации 14](#_heading=h.bvqds7a0r1d)

[4.3.1. Климатические условия эксплуатации 14](#_heading=h.q2rz0wolwp4u)

[4.3.2. Требования к количеству и квалификации персонала 14](#_heading=h.9b2lc6zg8ucf)

[4.4. Требования к составу и параметрам технических средств 15](#_heading=h.4kivlvcq8yj0)

[4.5. Требования к информационной и программной совместимости 16](#_heading=h.o7pdjvisoo2m)

[4.5.1. Требования к организации данных 16](#_heading=h.syj58447j684)

[4.5.2. Требования к базам данных 16](#_heading=h.ogl5vr8vl1e1)

[4.5.3. Требования к языкам программирования и операционным система 16](#_heading=h.uf1y5ljtxs9f)

[4.5.4. Требования к библиотекам 17](#_heading=h.b5e7n4yhdntt)

[4.6. Требование к маркировке, упаковке, транспортированию и хранению 17](#_heading=h.cq1aun24zgyf)

[**5. Требования к программной документации 18**](#_heading=h.1ci93xb)

[**6. Технико-экономические показатели 19**](#_heading=h.3whwml4)

[6.1. Ориентировочная экономическая эффективность 19](#_heading=h.d67w7rjxgi7m)

[6.2. Предполагаемая годовая потребность 19](#_heading=h.88jiiexagwwo)

[6.3. Экономические преимущества разработки 19](#_heading=h.9w26h4q9l1oo)

[**7. Стадии и этапы разработки 21**](#_heading=h.u8675gg42w41)

[7.1. Стадии разработки 21](#_heading=h.6kr2br7uw2nl)

[7.2. Этапы разработки 21](#_heading=h.amo8105ww6ku)

[7.3. Сроки разработки 22](#_heading=h.5s4wxkvhgl4x)

[**8. Порядок контроля и приёмки 23**](#_heading=h.ighc8sy3e9ra)

[8.1. Общие положение 23](#_heading=h.bbbnbwir997f)

[8.2. Виды испытаний 23](#_heading=h.knbjlhfwo60j)

[**9. Список использованных источников и литературы 24**](#_heading=h.zec2pusbggga)

# ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

**API** (англ. **A**pplication **P**rogramming **I**nterface) — описание способов взаимодействия программы с другими программами.

**СI/CD** (англ. **C**ontinuous **I**ntegration & **C**ontinuous **D**elivery) — технология автоматизации тестирования и доставки новых модулей разрабатываемого проекта заинтересованным сторонам (разработчикам, аналитикам, инженерам качества, конечным пользователям и др.).

**DevOps** (англ. **Dev**elopment & **Op**eration**s**) — методология автоматизация процессов сборки, настройки и развёртывания программного обеспечения.

**JWT** (англ. **J**SON **W**eb **T**oken) — это компактное, безопасное с точки URL средство представления прав, передаваемое между двумя сторонами [1].

**URI** (англ. **U**niform **R**esource **I**dentifier) — это компактная последовательность символов, которая идентифицирует абстрактный или физический ресурс [2].

**URL** (англ. **U**niform **R**esource **L**ocator) — подвид URI, который, помимо идентификации ресурса, предоставляет средство определения местоположения ресурса путем описания механизма первичного доступа к нему [2].

**Эндпоинт** (англ. endpoint) — URL веб-сервера, к которому происходит обращение для выполнения определённых операций или получения данных.

**Помощник, Приложение** или **Система** — программный комплекс, компонент которого разрабатывается в рамках данного технического задания.

**Программа**, **Серверная часть или Веб-приложение** — программа, разрабатываемая в рамках данного технического задания.

# ВВЕДЕНИЕ

## 1.1. Наименование программы

Полное наименование программы – Серверная часть индивидуального помощника для подбора косметических средств «Glamcheck».

## 1.2. Краткая характеристика области применения

Одной из актуальных проблем современной индустрии ухода за собой является сложность подбора подходящих косметических средств. Люди сталкиваются с нехваткой знаний о правильном уходе, недостатком времени для консультаций со специалистами и неловкостью при обращении к консультантам в магазинах. Это особенно важно для людей с индивидуальными особенностями кожи и волос, такими как сухость, склонность к трещинам или выпадению волос. Неправильно подобранные средства могут не только не решить проблемы, но и усугубить их, что снижает уровень доверия к продукции и увеличивает затраты пользователей.

Индивидуальный помощник для подбора косметических средств является полезным инструментом в данной сфере. Область применения включает такие аспекты, как поддержка пользователей в создании эффективного базового ухода, помощь в изучении состава косметических средств и выборе продукции, соответствующей индивидуальным потребностям. Приложение направлено на удовлетворение запросов широкой аудитории: от новичков в уходе за собой до опытных пользователей, желающих оптимизировать свои бьюти-рутины. Данный помощник разрабатывается в рамках выпускной квалификационной работы совместно со студенткой группы 21203 Егоренко К.В. с основным акцентом на реализацию серверной части помощника.

# ОСНОВАНИЯ ДЛЯ РАЗРАБОТКИ

## 2.1. Основание для проведения разработки

Основанием для проведения разработки является распоряжение № 0438

от 14-го ноября 2024 года на утверждение темы дипломной работы, согласованное Лаврентьевым Михаилом Михайловичем, деканом факультета информационных технологий Новосибирского Государственного Университета. Тема дипломной работы — «Разработка серверной части индивидуального помощника для подбора косметических средств».

Распоряжение, в свою очередь, основано на заявлении на инициативную тему выпускной квалификационной работы, согласованное с К. П. Кондренко (далее – Разработчик) и Г. Э. Яхъяевой (далее – Заказчик).

# 3. НАЗНАЧЕНИЕ РАЗРАБОТКИ

## 3.1. Функциональное назначение

Программа представляет собой серверный модуль, предоставляющий интерфейс, который должен предусматривать следующие возможности:

1. регистрация, аутентификация и авторизация пользователей;
2. анализ состава косметического средства;
3. поиск косметических средств по названию с возможностью фильтрации по цене;
4. подбор косметических средств с учётом индивидуальных особенностей пользователя.

## 3.2. Эксплуатационное назначение

Программа является серверной частью индивидуального помощника для подбора косметических средств. Она представляет собой веб-сервер, предоставляющий определённый API, и обеспечивает работу с используемыми базами данных, а также интеграцию с внешними системами.

## 3.3. Целевая аудитория

Целевая аудитория помощника включает в себя следующие группы пользователей:

1. Люди, начинающие ухаживать за собой: пользователи, которые не имеют опыта в выборе косметических средств и нуждаются в простых и понятных рекомендациях. Для них приложение станет инструментом для знакомства с базовыми аспектами ухода за кожей и волосами.
2. Занятые люди: пользователи, которым важно экономить время на изучение и выбор косметики. Приложение предлагает быстрые рекомендации, упрощая процесс выбора подходящих средств.
3. Люди с индивидуальными особенностями кожи и волос: пользователи, которые имеют индивидуальные особенности (например, сухая кожа, чувствительная кожа, ломкие волосы) и ищут решения, учитывающие их параметры.
4. Покупатели, заинтересованные в осознанном выборе: пользователи, которые хотят изучить состав средств и понять, как компоненты косметики влияют на их кожу и волосы. Это аудитория, которая стремится к осознанному потреблению и заботе о себе.
5. Косметологи и консультанты: профессионалы, которые могут использовать приложение для предоставления клиентам персонализированных рекомендаций, а также для подтверждения своей экспертности.

Возраст целевой аудитории может варьироваться от 18 до 55 лет, однако ключевая группа — это молодые и взрослые люди в возрасте 20–40 лет, которые заинтересованы в уходе за собой и стремятся улучшить качество своей жизни. Приложение ориентировано как на женщин, так и на мужчин, для которых уход за собой является важной частью ежедневной рутины.

# 4. ТРЕБОВАНИЯ К ПРОГРАММЕ ИЛИ ПРОГРАММНОМУ ИЗДЕЛИЮ

## 4.1. Требования к функциональным характеристикам

### 4.1.1. Регистрация пользователей

Веб-сервер должен принимать запрос, содержащий следующие данные для регистрации пользователя:

1. имя пользователя;
2. фамилия пользователя;
3. электронная почта пользователя;
4. пароль пользователя;
5. пол пользователя;
6. возраст пользователя;
7. индивидуальные особенности пользователя (КАКИЕ ИМЕННО — НА ЭТАПЫ ПРОДУМЫВАНИЯ).

После проверки всех полученных данных должно отправиться письмо на указанную электронную почту для подтверждения регистрации и должна создаться запись в базе данных, соответствующая пользователю, указавшему эти данные. Пароли пользователей в базе данных должны храниться в виде хеша Argon2.

### 4.1.2. Аутентификация пользователей

Программа должна поддерживать аутентификацию и авторизацию пользователей на основе механизма OAuth2.0 [3]. Как следствие, должны использоваться два токена: обновления и доступа.

Запрос на аутентификацию должен содержать следующие входные данные:

1. электронная почта пользователя;
2. пароль пользователя.

Ответ данного запроса должен содержать либо сообщение об ошибке, либо выданные пользователю два токена: доступа и обновления, которые нужны для использования помощника.

### 4.1.3. Анализ состава косметического средства

Программа должна поддерживать два эндпоинта, которые соответствует анализу состава косметического средства.

Первый из них анализирует состав по названию средства, а второй — по списку компонент средства, представленных в виде одной строки, в которой компоненты разделены некоторым указываемым при запросе разделителем.

Выходными данными обоих эндпоинтов является агрегированная по всем компонентам косметического средства информация, которая содержит:

1. фактор опасности (0-10);
2. натуральность (натуральный/синтетический);
3. дополнительные параметры средства;
4. косметические свойства средства;
5. для каких типов кожи эффективно средство.

Для получения информации о компонентах косметического средства предлагается использовать API онлайн-сервиса cosmobase.ru.

### 4.1.4. Поиск косметических средств

Программа должна поддерживать эндпоинт, который соответствует поиску косметических средств. В этом эндпоинте должен осуществляться поиск косметических средств по названию с возможность фильтрации по ценам.

Входные параметры эндпоинта:

1. строка для поиска косметического средства по названию;
2. минимальная и максимальная цена искомых косметических средств.

Выходными параметрами эндпоинта является список объектов, каждый из которых содержит следующие поля:

1. уникальный идентификатор косметического средства;
2. название косметического средства;
3. цена косметического средства;
4. ссылка на косметическое средство;
5. ссылка на обложку косметического средства.

### 4.1.5. Подбор косметических средств

Программа должна поддерживать эндпоинт, который позволит рекомендовать, то есть подбирать, косметические средства пользователям.

Входные данные: идентификатор косметического средства (необязательный параметр).

Выходные данные: список объектов (косметических средств), каждый из которых содержит следующие поля:

1. уникальный идентификатор косметического средства;
2. название косметического средства;
3. цена косметического средства;
4. ссылка на косметическое средство;
5. ссылка на обложку косметического средства.

Смысл необязательного параметра в том, что если пользователь уже находится на странице просмотра определенного косметического средства (то есть необязательный параметр передан в запросе), то можно учесть, на странице какого средства он находится, чтобы сделать дальнейшую рекомендацию. А если пользователь не находится на странице просмотра косметического средства (то есть необязательный параметр не передан в запросе), то для рекомендации можно использовать лишь предпочтения пользователя.

Для построения рекомендаций предполагается использовать векторные представления пользователей и косметических средств следующим образом:

1. Каждому косметическому средству и пользователю присваивается некоторое векторное представление.
2. Векторные представления пользователей и косметических средств сохраняются в векторную базу данных для дальнейшего использования.
3. Чтобы порекомендовать пользователю косметическое средство нужно определить векторное представление пользователя (получить это из базы данных) и найти косметические средства, векторное представление которых имеет наименьшее расстояние от векторного представления пользователя.

### 4.1.6. Логирование

Обращение к любому эндпоинту программы должно журналироваться в журнал событий с обязательным указанием следующих параметров:

1. дата и время получения запроса (в часовой зоне UTC);
2. уникальный псевдослучайно генерируемый идентификатор запроса;
3. параметры запроса;
4. ответ программы на запрос.

## 4.2. Требования к надежности

### 4.2.1. Требования к обеспечению надежного функционирования программы

Должен предоставляться непрерывный доступ к программе, расположенной по определённому URL как веб-сервис. Также она не должна непредвиденно прерывать свою работу. Помимо прочего программа должна корректно реагировать на некорректные запросы (например, запросы на несуществующие эндпоинты, передача некорректных данных в эндпоинты).

### 4.2.2. Время восстановления после отказа

В случае отказа работы серверной части и последующей недоступности веб-приложения, время восстановления не должно превышать одни рабочие сутки.

### 4.2.3. Отказы из-за некорректных действий оператора

После запуска программы на сервере, отказ программы вследствие некорректных действий оператора должен быть исключён. В том числе должна быть исключена возможность непреднамеренного выключения программы, не связанного с техническими неполадками сервера.

### 4.2.4. Организационно-технические мероприятия

Надежное функционирование программы должно быть обеспечено выполнением совокупности организационно-технических мероприятий:

1. организация бесперебойного питания технических средств, на которых она запущена;
2. организация бесперебойного доступа в глобальную сеть «Интернет» технических средств, на которых запущена программа;
3. необходимым уровнем квалификации обслуживающего персонала;
4. проверка входных данных для предотвращения возникновения ошибок и отказа работы программы.

## 4.3. Условия эксплуатации

### 4.3.1. Климатические условия эксплуатации

Климатические условия эксплуатации обязаны удовлетворять требованиям к эксплуатации, предъявляемым к техническим устройствам, на которых запущена программа.

### 4.3.2. Требования к количеству и квалификации персонала

Для запуска и поддержки системы необходимо как минимум четыре человека: DevOps-разработчик, инженер по обеспечения качества, системный администратор, как минимум один оператор.

В ходе запуска системный администратор должен подготовить информационную инфраструктуру для развёртывания и тестирования разрабатываемой программы, при этом должны соблюдаться требования к используемым техническим средствам. После запуска он должен поддерживать инфраструктуру в состоянии, пригодном для использования в рамках разрабатываемого программного изделия.

Инженер по обеспечению качества должен предоставить автоматические тесты на разрабатываемую программу с инструкцией по их использованию.

DevOps-инженер, используя инфраструктуру, предоставленную системным администратором, должен настроить CI/CD для разрабатываемой программы с использованием автоматических тестов от инженера по обеспечению качества, при этом должны выполняться требований к информационной и программной совместимости. После запуска он должен поддерживать CI/CD в рабочем состоянии, чтобы при этом выполнялись требования к надёжности.

Операторы не участвуют при запуске программного изделия, однако, в случае возникающих в ходе работы программы ошибок, им нужно анализировать журналы событий и оперативно реагировать на возникающие ошибки для соблюдения требований к надёжности. Операторы должны руководствоваться документом «Руководство оператора».

## 4.4. Требования к составу и параметрам технических средств

Разрабатываемая программа должна работать на устройстве, обладающем следующими минимальными техническими характеристиками:

1. объём ОЗУ не меньше 4 ГБ;
2. объём ПЗУ не меньше 150 ГБ;
3. процессор построен на архитектуре x86 или ARM;
4. есть доступ в глобальную сеть «Интернет».

## 4.5. Требования к информационной и программной совместимости

### 4.5.1. Требования к организации данных

Программа должна представлять собой веб-сервер, предоставляющий API, который совместим с RestAPI. То есть входные и выходные данные должны быть организованы, то есть приниматься и отдаваться, в формате JSON по протоколу HTTP; будут использованы стандартные единицы измерения дат, валют и времени.

### 4.5.2. Требования к базам данных

Хранение данных должно обеспечиваться с использование следующих баз данных:

1. Weaviate — векторная база данных, в которой предполагается хранить векторные представления пользователей и косметических средств. Должна быть использована в виде docker-контейнера из образа cr.weaviate.io/semitechnologies/weaviate:1.27.6.
2. PostgreSQL — реляционная база данных, в которой предполагается хранить информацию о пользователях. Должна быть использована в виде docker-контейнера из образа postgres:17.2.

### 4.5.3. Требования к языкам программирования и операционным система

Разрабатываемая программа должна быть написана на языке программирования Python 3.x. Предполагается, что программа будет запущена на операционной системе Ubuntu 24.x.x. LTS.

### 4.5.4. Требования к библиотекам

Используемые сторонние библиотеки:

1. fastapi 0.115.x — создание API на основе RestAPI;
2. aiohttp 3.x — асинхронная отправка HTTP-запросов;
3. pyjwt 2.x — выпуск и проверка JWT;
4. structlog 24.x — логирование работы серверного модуля;
5. argon2-cffi — хеширование паролей пользователей с использованием алгоритма Argon2;
6. fastapi-mail 1.4.x — для отправки электронной почты (взаимодействие с серверами, поддерживающими протоколы электронной почты);
7. sqlalchemy 2.x — взаимодействие с PostgreSQL;
8. weaviate-client 4.x — взаимодействие с Weaviate.

Для управления используемыми библиотеками должен быть использован инструмент Poetry 1.8.x.

## 4.6. Требование к маркировке, упаковке, транспортированию и хранению

Исходный код хранится в закрытом репозитории на Github. При передаче Заказчику, ему предоставляется доступ к репозиторию с правами на чтение и изменение.

Требования к маркировке и упаковке соответствуют требованиям, которые предъявляются той платформой, на которой будет распространяться программа, то есть системой GitHub.

# 5. ТРЕБОВАНИЯ К ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

Программная документация включает в себя следующие документы:

1. техническое задание;
2. спецификация;
3. текст программы;
4. описание программы;
5. описание применения;
6. ведомость эксплуатационных документов;
7. формуляр;
8. руководство программиста;
9. руководство системного программиста;
10. руководство по техническому обслуживанию;
11. программа и методика испытаний;
12. пояснительная записка.

Пояснительная записка должна включать в себя:

1. описание алгоритма рекомендаций косметических средств;
2. описание алгоритма анализа состава косметических средств;
3. описание механизма аутентификации и авторизации пользователей;
4. используемые базы данных вместе с их схемами;
5. обоснование принятых технических решений предыдущих для предыдущих четырёх пунктов;
6. API разрабатываемой программы как серверного модуля, соответствующий спецификации OpenAPI 3.1.0.

Все документы должны быть доступны как в печатной, так и в электронной форме и сопровождаться актуальной версией программного обеспечения.

# 6. ТЕХНИКО-ЭКОНОМИЧЕСКИЕ ПОКАЗАТЕЛИ

## 6.1. Ориентировочная экономическая эффективность

Предполагается, что создание и использование индивидуального помощника сократит время, необходимое для выбора подходящих продуктов, на 80%.

## 6.2. Предполагаемая годовая потребность

Предполагаемая годовая потребность в разработке и поддержке системы напрямую зависит от успешности маркетинговых и рекламных мероприятий после завершения разработки. В первую очередь, планируется сотрудничество с крупными косметическими брендами, с целью интеграции их продукции в сервис для более точных рекомендаций. Также следует учитывать растущий интерес к онлайн-платформам и приложениям для ухода за собой, что обеспечит спрос на приложение.

## 6.3. Экономические преимущества разработки

Существуют различные онлайн-сервисы и мобильные приложения, которые предоставляют рекомендации по уходу за кожей и подбору косметики, но большинство из них имеют ограниченную персонализацию или не интегрируют данные о составе продуктов. Индивидуальный помощник, с другой стороны, не только предлагает рекомендации на основе индивидуальных предпочтений пользователя, но и включает подробную информацию о составе продукции, что является уникальной чертой. Таким образом, разрабатываемое приложение имеет значительные преимущества перед аналогами, предлагая персональные рекомендации и анализ состава косметических средств. На данный момент аналоги, предоставляющие персонализированные рекомендации в области подбора косметических средств и анализ состава косметики, не были найдены.

# 7. СТАДИИ И ЭТАПЫ РАЗРАБОТКИ

## 7.1. Стадии разработки

Разработка должна быть проведена в три стадии:

1. разработка технического задания;
2. рабочее проектирование;
3. внедрение.

## 7.2. Этапы разработки

На стадии разработки технического задания должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. разработка технического проекта
2. разработка программы;
3. составление программной документации;
4. испытания программы.

На стадии внедрения должна быть выполнена подготовка и передача программы.

## 7.3. Сроки разработки

|  |  |  |
| --- | --- | --- |
| Этапы | Сроки | Исполнители |
| Разработка, согласование и утверждение технического задания | 26.11.2024 - 02.12.2024 | Кондренко К.П. |
| Разработка технического проекта | 02.12.2024 - 15.12.2024 | Кондренко К.П. |
| Разработка программы | 15.12.2024 - 28.12.2024 | Кондренко К.П. |
| Составление программной документации | 01.01.2025 - 01.02.2025 | Кондренко К.П. |
| Испытания программы | 01.02.2025 - 10.02.2025 | Кондренко К.П.  Яхъяева Г.Э. |
| Внедрение | 15.02.2025 - 01.03.2025 | Кондренко К.П.  Егоренко К.В.  Яхъяева Г.Э. |

# 8. ПОРЯДОК КОНТРОЛЯ И ПРИЁМКИ

## 8.1. Общие положение

1. Приемосдаточные испытания программы должны проводиться в соответствии с документом «Программа и методика испытаний».
2. Ход проведения приемо-сдаточных испытаний документируется в протоколе испытаний.
3. Заказчик подписывает акт приёма и передачи при прохождении 100% испытаний, заявленных в документе «Программа и методика испытаний». Передача осуществляется в соответствии с требованиями к транспортировке и хранению.

## 8.2. Виды испытаний

Производится проверка корректного выполнения программой заложенных в нее функций, т.е. осуществляется функциональное тестирование программы.

Также проводится нагрузочное тестирование, которое должно показать, что разрабатываемое программное изделие выдерживает высокие нагрузки, то есть остаётся доступным при них.

# 9. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. RFC 7519 — JSON Web Token (JWT).
2. RFC 3986 — Uniform Resource Identifier (URI): Generic Syntax.
3. RFC 6749 — The OAuth 2.0 Authorization Framework.