Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Отчёт**

по предмету «Проектирование программного обеспечения»

Лабораторная работа №1

«Интерфейс электронной почты»

Студент: Лопатнюк П.В.

ФИТ 3 курс 1 группа

Преподаватель: Якубенко К. Д

Минск 2024

# Введение

Проект, посвящённый разработке интерфейса электронной почты, направлен на создание удобного и функционального инструмент. Основная цель — предложить современный интерфейс, который упрощает взаимодействие с почтовыми сервисами и обеспечивает эффективное использование электронной почты для различных категорий пользователей.

Актуальность темы обусловлена возрастающим объёмом электронной почты в бизнесе и повседневной жизни. Удобный и функциональный интерфейс электронной почты позволяет пользователям оптимизировать работу с письмами, эффективно управлять большим количеством сообщений и улучшить общение как внутри организаций, так и между частными лицами. Существует множество аналогичных решений, таких как Gmail, Microsoft Outlook, Yahoo Mail, и ProtonMail, которые предлагают различные возможности для работы с электронной почтой, интеграцию с другими сервисами и инструменты для обеспечения безопасности. Данный проект призван конкурировать с этими решениями, предлагая уникальные функции и улучшенный пользовательский опыт.

Целевой аудиторией проекта являются как частные пользователи, так и сотрудники компаний, которые ежедневно работают с электронной почтой. Интерфейс также будет полезен бизнесу, требующему быстрой и безопасной коммуникации с клиентами и партнёрами.

Цель проекта — создание почтового интерфейса, который бы сочетал удобство использования, высокую производительность и безопасность. Основные задачи для достижения этой цели включают:

* Разработка интуитивно понятного интерфейса для всех типов пользователей.
* Реализация функций управления почтой, таких как фильтрация, поиск и сортировка писем.
* Внедрение механизмов защиты, включая шифрование и фильтрацию спама.
* Поддержка интеграции с другими сервисами и платформами.

Роли пользователей в проекте включают рядовых пользователей, которые будут отправлять и получать почту, системных администраторов для управления настройками почтового сервиса и разработчиков, которые будут поддерживать и развивать систему.

Кратко говоря, архитектура программного обеспечения включает в себя клиент-серверную модель, где сервер обрабатывает запросы и отправку данных, а клиент предоставляет удобный и интерактивный пользовательский интерфейс.

# 1. Основания для разработки

## 1.1. Основание для проведения разработки

Основанием для проведения разработки интерфейса электронной почты является необходимость создания современного, удобного и безопасного решения для работы с цифровой корреспонденцией, которое может удовлетворить потребности как частных пользователей, так и корпоративных клиентов. Существующие решения, такие как Gmail и Outlook, обладают рядом преимуществ, но зачастую либо перегружены дополнительными функциями, либо недостаточно гибки для индивидуальных настроек.

Кроме того, рост объёма электронной почты, используемой в деловых и повседневных коммуникациях, требует улучшения пользовательского опыта для более эффективного управления почтой, включая автоматизацию рутинных операций, улучшенные инструменты сортировки и поиска, а также интеграцию с другими бизнес-инструментами.

Разработка нового интерфейса позволит решить следующие проблемы:

* Перегруженность интерфейсов существующих почтовых сервисов.
* Недостаточная гибкость настроек для индивидуальных пользователей и компаний.
* Ограниченные возможности по интеграции с внешними системами (например, CRM или проектными системами).
* Необходимость повышенной защиты данных в условиях возрастающих угроз.

Таким образом, разработка нового интерфейса электронной почты позволит улучшить удобство работы, повысить продуктивность пользователей и обеспечить более высокий уровень защиты данных, что является важным аспектом для бизнеса и частных лиц.

## 1.2. Наименование и условное обозначение темы разработки

Наименование темы разработки: «Формирование требований к информационной системе. Интерфейс электронной почты».

Условное обозначение темы разработки: «Интерфейс электронной почты».

# 2. Назначение разработки

## 2.1. Функциональное назначение программы

Программа предназначена для организации и упрощения процесса управления электронной почтой. Основные функции программы включают:

* Пользователи могут отправлять электронные письма, прикреплять файлы, управлять отправленной и входящей корреспонденцией.
* Управление почтовыми ящиками: поддержка нескольких почтовых ящиков с возможностью их синхронизации и удобного переключения между ними.
* Сортировка, фильтрация и создание пользовательских папок для хранения сообщений, а также установка меток для важной корреспонденции.
* Возможность быстрого поиска сообщений по ключевым словам, отправителю или дате, а также использование фильтров для автоматической обработки входящих писем.
* Получение push-уведомлений при поступлении новых сообщений или обновлениях в почтовом ящике.
* Удобная работа с контактами и интеграция с календарём для управления задачами и событиями.
* Поддержка протоколов безопасности (SSL/TLS) для защиты данных, а также встроенные средства борьбы со спамом и фишинговыми атаками.

Таким образом, программа обеспечивает полную функциональность для эффективной работы с электронной корреспонденцией, улучшая взаимодействие пользователей с их почтовыми сервисами.

## 2.2. Эксплуатационное назначение программы

Программа «Интерфейс электронной почты» предназначена для повседневного использования с целью отправки, получения и управления электронной корреспонденцией. Она поддерживает многозадачность, автоматизацию обработки писем, работу с несколькими учётными записями, и интеграцию с другими сервисами. Программа обеспечивает безопасность данных через шифрование и фильтрацию спама, подходит как для личного, так и для корпоративного использования.

# 3. Требования к программе

## 3.1. Требования к функциональным характеристикам

### 3.1.1. Целевая аудитория. Описание ролей

Программа «Интерфейс электронной почты» предназначена для широкого круга пользователей, включая индивидуальных пользователей, малый и средний бизнес, а также корпоративные и IT-структуры. В зависимости от специфики использования, система поддерживает несколько ролей, каждая из которых имеет свои особенности взаимодействия с программой.

Основная категория — это пользователи. Основные задачи рядовых пользователей — отправка и получение писем, организация входящей корреспонденции и поддержание эффективной коммуникации с контактами.

К задачам так же относится ежедневная работа с электронной почтой, управление почтовыми ящиками, настройка уведомлений и использование фильтров для сортировки писем.

У пользователей есть доступ к своим почтовым ящикам, создание и редактирование писем, настройка фильтров и управление личными данными.

Системные администраторы отвечают за поддержку инфраструктуры электронной почты, контроль безопасности и администрирование учётных записей. Они обеспечивают корректную работу системы и предотвращают нарушения в работе.

К задачам относится управление правами доступа пользователей, настройка политик безопасности, защита от спама и вирусов, мониторинг производительности системы и резервное копирование данных.

У системных администраторов есть полный доступ к настройкам системы, управление учётными записями и серверами, настройка протоколов безопасности, мониторинг активности.

Разработчики занимаются внедрением и поддержкой системы. Они работают над обновлениями, устранением ошибок и интеграцией с внешними сервисами. Разработчики также тестируют новые функции для обеспечения стабильности работы системы.

К задачам относится разработка новых функций, оптимизация производительности, настройка интеграций с другими приложениями (например, календарь, CRM), и поддержка API для расширения функционала системы.

У разработчиков есть доступ к коду системы, базам данных, тестовым средам и инструментам для разработки и отладки программного обеспечения.

Таким образом, целевая аудитория включает как индивидуальных пользователей, так и бизнес-структуры с различными ролями, обеспечивающими эффективное использование системы, её поддержку и развитие.

### 3.1.2. Требования к составу выполняемых функций

Программа «Интерфейс электронной почты» должна обеспечивать выполнение набора функций для разных категорий пользователей: рядовых пользователей, системных администраторов и разработчиков. Каждая роль требует определённого функционала для работы с системой.

Пользователю доступны следующие функции:

* Отправка и получение писем: пользователи могут создавать, отправлять и получать письма, прикреплять файлы, а также управлять списком контактов.
* Организация почты: возможность сортировать письма по папкам, меткам, архивировать сообщения, а также удалять их.
* Поиск и фильтрация: возможность искать письма по ключевым словам, дате, отправителю, а также настраивать фильтры для автоматической сортировки.
* Настройка уведомлений: возможность включать или отключать push-уведомления о новых сообщениях.

Системному администратору доступны следующие функции:

* Управление пользователями: создание, редактирование и удаление учётных записей пользователей, настройка их прав и доступов.
* Мониторинг активности: просмотр статистики использования системы, контроль за загрузкой почтовых ящиков, а также отслеживание подозрительной активности.
* Настройка безопасности: управление политиками безопасности, настройка шифрования данных, фильтрация спама и вирусов, блокировка нежелательных пользователей и доменов.
* Поддержка инфраструктуры: управление серверами электронной почты, настройка резервного копирования данных, восстановление после сбоев.

При подключении в режиме разработчика ему доступна возможность:

* Обеспечения поддержки системы, устранения ошибок и внедрения новых функций.
* Интеграции с внешними системами: настройка API и интеграций с другими сервисами, такими как календарь или системы управления проектами.
* Обновления и тестирования: реализация новых версий системы, тестирование на стабильность и производительность, оптимизация кода и процессов.
* Управления базой данных: оптимизация работы с базой данных для быстрого поиска и обработки больших объёмов писем.

Каждая категория пользователей должна иметь доступ к соответствующему функционалу, обеспечивая гибкость и безопасность системы, удобство работы с почтовыми ящиками и эффективное управление.

### 3.1.2. Требования к организации входных данных

Данные о пользователях и их почтовых ящиках хранятся в базе данных. Входные данные программы должны вводиться в базу данных в цифровом формате в соответствии с полями таблиц БД. Система должна обеспечивать функции проверки корректности данных при вводе.

При регистрации пользователь должен соблюдать следующие ограничения по длине:

* Логин должен быть уникальным и содержать от 4 до 40 символов.
* Пароль должен иметь длину от 5 до 30 символов.
* Имя должно содержать от 2 до 20 символов.
* Фамилия должна быть длиной от 2 до 25 символов.
* Электронная почта должна быть уникальной, длиной от 5 до 50 символов и соответствовать формату, заданному регулярным выражением.
* Телефон должен быть уникальным, с длиной от 8 до 30 символов и также соответствовать регулярному выражению.

При создании почтового ящика пользователем необходимо вводить следующее:

* Название ящика должно быть уникальным и содержать от 4 до 50 символов.
* Описание ящика может быть пустым и должно иметь длину до 250 символов.

При добавлении нового письма необходимо вводить:

* Заголовок письма должен содержать от 1 до 100 символов.
* Содержание письма может быть длинным, но должно ограничиваться 5000 символами.
* Дата и время отправки должны быть указаны в формате «гггг-мм-дд чч:мм».

При добавлении контактов, пользователи должны указывать:

* Имя контакта, длиной от 2 до 30 символов.
* Электронная почта контакта должна быть в стандартном формате с символами «@» и «.», и допустимыми доменами.

Все входные данные должны проходить валидацию на стороне клиента перед отправкой в базу данных. В случае некорректных данных система должна возвращать соответствующие сообщения об ошибках, обеспечивая пользователям понятные подсказки для исправления.

### 3.1.3. Требования к организации выходных данных

Для обеспечения удобства и эффективности использования системы «Интерфейс электронной почты» были введены следующие требования к организации выходных данных:

* Информация о полученных и отправленных письмах должна содержать отправителя, получателя, тему письма, дату и время отправки, статус письма, и краткое содержание письма в удобном для чтения формате.
* Подтверждение успешной отправки письма должно отображать информацию о теме письма, адресате, дате и времени отправки.
* В личном кабинете пользователя должна отображаться полная история переписки, включая дату, адресата, тему, а также возможность сортировки и фильтрации по разным параметрам (дата, статус, отправитель).
* Контакты пользователя должны выводиться с именем, электронной почтой и возможностью быстро отправить сообщение или просмотреть предыдущую переписку с данным контактом.
* Система должна выводить данные о вложениях в письмах, включая их название, размер и возможность быстрого скачивания.

### 3.1.4. Требования к временным характеристикам

Система «Интерфейс электронной почты» должна удовлетворять следующим требованиям к временным характеристикам:

* Отправка электронного письма должна происходить не более чем за 5 секунд при стабильном интернет-соединении.
* Загрузка списка писем в почтовом ящике должна осуществляться за время не более 3 секунд при наличии до 1000 писем.
* Выполнение поиска по ключевым словам или фильтрам должно занимать не более 2 секунд при запросе в базе данных объёмом до 10 000 писем.
* При открытии письма его содержимое должно отображаться в течение 1–2 секунд, включая вложенные файлы.
* Синхронизация с почтовым сервером должна происходить не реже одного раза в 30 секунд для обновления новых писем и изменений статуса.

## 3.2. Требования к надёжности

### 3.2.1. Требования к обеспечению надёжного (устойчивого) функционирования программы

Для обеспечения надёжного функционирования системы «Интерфейс электронной почты» должны быть выполнены следующие организационно-технические мероприятия:

* Организация бесперебойного питания серверного оборудования и использование источников бесперебойного питания.
* Регулярное резервное копирование данных, включая письма и контактную информацию, с хранением копий на защищённых внешних носителях.
* Использование лицензионного программного обеспечения и своевременное обновление как серверной, так и клиентской частей системы.
* Реализация средств мониторинга и оповещения о сбоях в работе системы с возможностью быстрого реагирования.
* Планирование и регулярное тестирование процессов восстановления после сбоев для минимизации простоев.
* Соблюдение требований к информационной безопасности, включая регулярные проверки системы на наличие уязвимостей и вирусов.
* Внедрение системы аварийного выключения для предотвращения повреждений данных в случае критических ошибок.

### 3.2.2. Время восстановления после отказа

Система должна обеспечивать минимальное время восстановления после отказа для поддержания бесперебойной работы. Время восстановления после критического сбоя не должно превышать 30 минут, включая выполнение резервного восстановления данных.

Для достижения данного показателя необходимо реализовать автоматическую систему резервного копирования, которая позволит восстановить данные с минимальными потерями, внедрить механизмы мониторинга и оповещения, позволяющие оперативно реагировать на сбои и разработать план восстановления, который включает действия по устранению неполадок и восстановлению системы после сбоев.

### 3.2.3. Отказы из-за некорректных действий

Отказы программы могут возникать вследствие некорректных действий пользователя при взаимодействии с системой. Для предотвращения подобных сбоев необходимо ограничить доступ пользователей без административных привилегий, предоставляя только те функции, которые соответствуют их ролям.

При ошибках система должна выдавать понятные сообщения о характере ошибки и предлагать шаги по исправлению. В случае потери или повреждения данных вследствие некорректных действий восстановление должно осуществляться с использованием последних резервных копий не позднее чем через 1 час.

## 3.3. Условия эксплуатации

### 3.3.1. Общие требования к эксплуатации

Программа для интерфейса электронной почты запускается на рабочих станциях сотрудников и пользователей через веб-браузеры или почтовые клиенты. Серверная часть программы размещается на выделенных серверах с обеспечением надёжного соединения и защиты данных между клиентом и сервером.

Программа должна быть интегрирована в операционную систему для автоматического запуска при старте, а также обеспечивать защиту доступа с помощью падежной системы аутентификации. Для предотвращения сбоев и потери данных необходимо организовать регулярное резервное копирование. Техническое оборудование должно работать в стабильных условиях, предотвращающих перегрев.

### 3.3.2. Требования к видам обслуживания

Для обеспечения стабильной работы программы должны проводиться следующие виды обслуживания:

* Регулярные проверки состояния серверов и сетевого оборудования для предотвращения сбоев. Мониторинг температуры и состояния технических устройств для предотвращения перегрева и других аппаратных отказов.
* Периодические обновления программного обеспечения для устранения уязвимостей, добавления новых функций и улучшения производительности. Регулярная проверка программы на наличие ошибок и некорректных данных, исправление обнаруженных дефектов.
* Регулярное создание резервных копий базы данных для возможности восстановления системы в случае сбоя или потери данных.
* Мониторинг работы пользователей и их учётных записей для предотвращения несанкционированного доступа и неправильного использования системы.

### 3.3.3. Требования к численности и квалификации персонала

Для обеспечения надёжной работы программы необходимо привлечение квалифицированного персонала, распределенного по следующим ролям.

Системный администратор — это ключевая фигура, отвечающая за модернизацию, настройку и мониторинг работы серверов, рабочих станций и сети. Он устанавливает, настраивает и обслуживает операционные системы и прикладное программное обеспечение, а также ведёт учётные записи пользователей, настраивая права доступа. Кроме того, системный администратор организует резервное копирование данных и плановое обновление программного обеспечения. Для этой роли требуется высшее профильное образование в области информационных технологий, а также практический опыт работы по установке, настройке и администрированию серверного оборудования и операционных систем. Наличие сертификатов по администрированию систем и безопасности от производителей используемого ПО также является необходимым требованием.

Администратор баз данных отвечает за установку, настройку и обслуживание программного обеспечения СУБД, а также за оптимизацию баз данных для повышения производительности. Он разрабатывает и реализует эффективную политику доступа к информации и мониторит состояние базы данных, обеспечивая её целостность и безопасность. Для этой роли также требуется высшее профильное образование и опыт работы с реляционными СУБД. Знания языков запросов, таких как SQL, и методик работы с большими объёмами данных будут преимуществом.

Администратор информационной безопасности разрабатывает и реализует политику информационной безопасности системы, контролирует и управляет доступом пользователей к информации и функциям системы. Он регулярно мониторит уязвимости системы и применяет обновления безопасности, а также обучает персонал основам информационной безопасности. Для этой роли также необходимо высшее профильное образование, опыт работы в сфере информационной безопасности и знание нормативных требований по защите информации, таких как ISO 27001.

Пользователи программы могут варьироваться в зависимости от объёма эксплуатации системы. Их обязанности заключаются в работе с интерфейсом программы в соответствии с должностными инструкциями, выполняя операции ввода и обработки данных в пределах предоставленных прав доступа. Специальное обучение для пользователей не требуется, но они должны пройти краткий курс по использованию системы, чтобы понимать ключевые функции.

При необходимости может быть привлечён технический специалист по обслуживанию оборудования. Он отвечает за проведение плановых технических осмотров и обслуживания оборудования, а также за устранение неполадок. Специалист взаимодействует с поставщиками оборудования для ремонта и замены неисправных устройств. Для этой роли достаточно среднего специального или высшего технического образования, а также опыта работы с компьютерной техникой и сетевым оборудованием.

Общее требование ко всем ролям заключается в наличии необходимого уровня квалификации. Для системного администратора и администратора баз данных требуется высшее профильное образование, в то время как пользователи и технический специалист могут обойтись средним специальным. Системный администратор и администратор информационной безопасности должны иметь действующие сертификаты от производителей операционных систем и ПО, используемых в организации, а также минимум три года опыта работы для ключевых технических позиций.

## 3.4. Требования к составу и параметрам технических средств

Для обеспечения надеждой и высокопроизводительной работы интерфейса электронной почты, а также для удовлетворения потребностей пользователей, необходимо учитывать следующие требования к составу и параметрам технических средств.

Компьютеры сотрудников, включая администраторов и техническую поддержку, должны быть оснащены процессором x86 с тактовой частотой не менее 2.0 GHz, оперативной памятью объёмом не менее 8 GB и SSD или жёстким диском с объёмом не менее 256 GB для хранения данных и программного обеспечения. Для удобства работы также необходимы мышь, клавиатура и монитор.

Пользовательские компьютеры должны иметь аналогичные характеристики: процессор x86 с тактовой частотой не менее 2.0 GHz, оперативная память объёмом не менее 4 GB и стандартные устройства ввода и вывода.

Сервер, на котором будет размещаться база данных электронной почты, должен быть более производительным, с процессором x86 с тактовой частотой не менее 2.5 GHz, оперативной памятью объёмом не менее 32 GB и SSD объёмом не менее 1 TB для основной базы данных и резервных копий. Также потребуется мышь и клавиатура для администрирования сервера.

В системе должна быть обеспечена высокая скорость сети, не менее 1 Гбит/с, для поддержки быстрой передачи данных между клиентами и сервером. Все технические средства должны быть размещены в условиях, предотвращающих перегрев, и поддерживать регулярное резервное копирование данных для обеспечения возможности восстановления информации в случае сбоев.

## 3.5. Требования к информационной и программной совместимости

### 3.5.1. Требования к информационным структурам и методам решения

Для обеспечения эффективной работы интерфейса электронной почты, необходимо учитывать требования к информационным структурам и методам решения, которые позволят оптимально организовать и управлять данными.

Во-первых, структура данных должна включать таблицы для хранения информации о пользователях, сообщениях, вложениях, контактах и настройках учётных записей. Каждая таблица должна иметь чётко определённые поля и типы данных. Например, таблица пользователей может содержать такие поля, как уникальный идентификатор пользователя, адрес электронной почты, имя, фамилию, дату регистрации и уровень доступа. Таблица сообщений должна включать идентификаторы отправителя и получателя, тему, текст сообщения, дату и время отправки, а также статус сообщения (прочитано/непрочитано).

Во-вторых, необходимо использовать методы решения для управления этими данными. Для обеспечения быстрого доступа к информации следует реализовать индексацию полей, часто используемых в запросах, таких как адрес электронной почты и дата отправки сообщений. Это позволит ускорить выполнение операций поиска и фильтрации.

Для обеспечения безопасности данных следует применить методы шифрования информации при передаче и хранении. Кроме того, необходимо реализовать механизмы аутентификации и авторизации пользователей, чтобы гарантировать доступ к информации только уполномоченным лицам.

Также важно учитывать методы резервного копирования и восстановления данных, которые помогут минимизировать риск потери информации. Регулярные резервные копии должны храниться в безопасном месте и быть доступны для восстановления в случае сбоя системы.

### 3.5.2. Требования к исходным кодам и языкам программирования

Программа должна быть написана на языках программирования, поддерживающих создание интерактивных и адаптивных веб-приложений, таких как C#, JavaScript и HTML/CSS. Использование ASP.NET Core обеспечит надёжную серверную часть, позволяя создавать масштабируемые и производительные приложения. Важно, чтобы исходный код был организован и структурирован, следуя принципам чистого кода и ООП, что упростит его дальнейшую поддержку и развитие.

Кроме того, для реализации функциональности интерфейса электронной почты необходимо интегрировать различные библиотеки и фреймворки, такие как Entity Framework для работы с базами данных, что упростит манипуляции с данными, такими как отправка и получение писем. База данных должна хранить информацию о пользователях, их учётных записях, содержании писем, а также данных для авторизации. Важно, чтобы структура базы данных была нормализована для оптимизации производительности и минимизации избыточности данных.

Программа должна предусматривать возможность работы с реляционными базами данных, такими как MS SQL Server, что позволит эффективно обрабатывать и хранить данные. Код должен включать механизмы обработки ошибок и исключений для обеспечения падежной работы системы, а также проводить валидацию данных перед их записью в базу данных, что повысит устойчивость приложения к некорректным данным.

### 3.5.3. Требования к программным средствам, используемым программой

Требования к программным средствам, используемым в интерфейсе электронной почты, включают в себя набор технологий и инструментов, необходимых для обеспечения функциональности, производительности и безопасности приложения. Важнейшие аспекты этих требований следующие:

* Программа должна быть совместима с современными операционными системами, такими как Windows, macOS и Linux, чтобы обеспечить широкую доступность для пользователей.
* Необходима установка и настройка веб-сервера, например, IIS для Windows или Nginx для Linux, чтобы обеспечить обработку запросов пользователей и поддержку различных протоколов связи, таких как HTTP/HTTPS.
* Для хранения данных о пользователях, письмах и других ресурсах необходимо использовать реляционные базы данных, такие как MS SQL Server, PostgreSQL или MySQL. База данных должна поддерживать механизмы резервного копирования и восстановления данных для обеспечения надёжности.
* Основным языком программирования для разработки серверной части приложения должен быть C#, что обеспечит совместимость с платформой .NET. Для клиентской части следует использовать JavaScript и HTML/CSS для создания интерактивного интерфейса.
* Для ускорения разработки и повышения качества кода рекомендуется использовать такие фреймворки, как ASP.NET Core для серверной части и React или Angular для создания динамичного пользовательского интерфейса. Кроме того, необходимо интегрировать библиотеки для работы с электронной почтой, такие как MailKit, для обеспечения функциональности отправки и получения писем.
* Программа должна разрабатываться с использованием интегрированных сред разработки (IDE), таких как Visual Studio или Visual Studio Code, которые обеспечивают удобные инструменты для написания и отладки кода.
* Для обеспечения качества и надёжности кода необходимо использовать автоматизированные тестовые фреймворки, такие как xUnit или NUnit.
* Для обеспечения устойчивого функционирования системы следует внедрить средства мониторинга и логирования, такие как Serilog или NLog, которые позволят отслеживать состояние приложения и выявлять потенциальные проблемы в реальном времени.

Эти требования помогут обеспечить высокое качество, безопасность и производительность интерфейса электронной почты, удовлетворяющего потребности пользователей.

### 3.5.4. Требования к защите информации и программ

Все данные, включая личную информацию пользователей и содержимое электронной почты, должны быть защищены с помощью алгоритмов шифрования AES-256 как при передаче, так и при хранении.

Доступ к системе должен ограничиваться многофакторной аутентификацией и чётким разграничением ролей пользователей для предотвращения несанкционированного доступа. Необходим мониторинг и аудит системы для отслеживания подозрительной активности и оперативного реагирования на инциденты.

Также обязательным является использование антивирусного программного обеспечения и брандмауэров для защиты от вредоносных программ и сетевых атак. Регулярные проверки безопасности и тестирование на уязвимости помогут выявить слабые места и повысить уровень защиты.

## 3.6. Специальные требования

Разработка интерфейса электронной почты должна учитывать доступность для пользователей с ограниченными возможностями, поддержку мультиязычности и адаптивный дизайн для корректного отображения на различных устройствах. Программа обязана обеспечивать безопасность данных в соответствии с законодательством о защите информации, включая возможность удаления данных по запросу пользователя. Важно также предусмотреть интеграцию с другими сервисами, высокую производительность и поддержку различных форматов файлов для отправки и получения. Соблюдение этих требований обеспечит удобный и безопасный интерфейс для всех категорий пользователей.

# Требования к программной документации

## 4.1. Предварительный состав программной документации

Программная документация для интерфейса электронной почты должна включать несколько ключевых компонентов, обеспечивающих полное описание системы, её функционала и эксплуатационных особенностей. Основными разделами документации являются:

* Техническое задание, в котором детализируются цели и задачи разработки, требования к системе, функциональные возможности и ограничения.
* Пользовательская документация, которая предоставляет инструкции по установке, настройке и использованию программы. В неё входят руководства по работе с интерфейсом, отправке и получению писем, настройке учётной записи и безопасности.
* Программистская документация, описывающая архитектуру системы, структуру исходного кода и требования к языкам программирования и библиотекам. Она необходима для последующей поддержки и модификации системы.
* Документация по безопасности, которая включает описание используемых методов шифрования данных, политик доступа и мер для защиты информации.
* Тестовая документация, содержащая сценарии тестирования функционала программы, отчёты по проведённым тестам и результаты проверки на соответствие требованиям.
* Документация по развёртыванию, описывающая процесс установки и конфигурации программного обеспечения на серверах и клиентских устройствах, а также взаимодействие с базой данных и сетевой инфраструктурой.

## 4.2. Специальные требования к программной документации

Программная документация должна быть понятной и доступной для всех категорий пользователей. Она должна своевременно обновляться при изменениях в системе, обеспечивать инструкциями по защите данных и управлению безопасностью, включая рекомендации по шифрованию и аутентификации.

Особое внимание следует уделить поддержке многопользовательской работы, настройке прав доступа и совместимости с базами данных. Документация должна быть доступна на нескольких языках для интернациональных пользователей, с детальными описаниями всех этапов установки и использования системы.

# 5. Технико-экономические показатели

## 5.1. Ориентировочная экономическая эффективность

Внедрение интерфейса электронной почты может значительно повысить производительность и оптимизировать процессы обмена информацией, что способствует снижению затрат на ручную обработку данных и ускорению коммуникаций. Автоматизация функций электронной почты сократит время, необходимое для выполнения рутинных задач, таких как рассылка писем, управление контактами и обработка вложений.

Ожидается, что улучшение системы безопасности за счёт использования современных методов шифрования и защиты данных минимизирует риски утечек информации, тем самым снижая потенциальные финансовые потери. В долгосрочной перспективе экономическая эффективность также проявляется в снижении затрат на техническое обслуживание и улучшении общей надёжности работы системы.

## 5.2. Предполагаемая годовая потребность

Для полноценного функционирования и поддержки интерфейса электронной почты в течение года потребуется выделение ресурсов на техническое обслуживание серверов, обновление программного обеспечения, обеспечение безопасности и резервное копирование данных. Также необходимо предусмотреть затраты на обучение сотрудников, администрирование системы и решение возникающих технических проблем.

Ожидается, что потребность в ежегодных инвестициях будет включать лицензии на используемые программные продукты, обслуживание серверного оборудования и поддержку системы безопасности.

## 5.3. Экономические преимущества разработки

Разработка интерфейса электронной почты принесёт значительные экономические преимущества за счёт оптимизации рабочих процессов и сокращения времени на обработку сообщений. Внедрение эффективной системы позволит уменьшить затраты на ручную работу и сократить риск ошибок при передаче данных. Автоматизация функций управления электронной почтой улучшит организацию работы и повысит продуктивность сотрудников. Благодаря использованию современных технологий шифрования и защиты данных, будут минимизированы финансовые потери, связанные с утечками информации и атаками.

6. Стадии и этапы разработки

## 6.1. Стадии разработки

Разработка программы «Интерфейс электронной почты» включает следующие стадии:

* Эскизное проектирование. На этой стадии формируются общие идеи и концепция программы «Интерфейс электронной почты». Разрабатываются первичные эскизы пользовательского интерфейса, определяются основные функции системы, такие как отправка, получение и хранение электронных сообщений, работа с контактами, поддержка вложений и прочее. Формулируются требования к производительности и защите данных.
* Техническое проектирование. На данном этапе создаётся детализированная архитектура системы. Определяются технологии, которые будут использоваться в разработке (например, ASP.NET Core для серверной части, HTML/CSS и JavaScript для клиентской части, SQL для базы данных). Разрабатываются схемы взаимодействия между компонентами программы, структура базы данных для хранения сообщений, контактов и других данных, а также прорабатывается система безопасности.
* Реализация (разработка кода). В этой стадии идёт непосредственная разработка программного кода. Программируются все ключевые компоненты, такие как серверная часть (логика обработки запросов, взаимодействие с базой данных), клиентская часть (интерфейс пользователя), а также системы авторизации и аутентификации, защиты данных и их шифрования. Реализуются методы работы с сообщениями, фильтрации, управления контактами и другими элементами функционала.
* Тестирование и отладка. После написания кода проводится тестирование всех модулей системы. Это включает проверку функциональности, совместимости с различными браузерами и устройствами, производительности и защиты данных. Тестируется отправка и получение сообщений, корректность работы с вложениями, управление контактами и уведомления. Выявленные ошибки и баги исправляются, проводится дополнительная отладка для обеспечения стабильной работы программы.
* Ввод в эксплуатацию. На этом этапе программа устанавливается на сервера и запускается в реальной среде для пользователей. Проводится мониторинг работы системы, отслеживаются и устраняются возможные проблемы, возникающие при эксплуатации. Организуется обучение пользователей (например, системных администраторов и техподдержки).
* Поддержка и сопровождение. После ввода программы в эксплуатацию осуществляется её постоянное сопровождение. Это включает исправление багов, которые могут проявиться в процессе использования, обновление системы для улучшения безопасности и производительности, добавление новых функций по мере необходимости.

## 6.2. Этапы разработки

Этапы разработки программы:

* Разработка технического задания.
* Проектирование архитектуры системы и базы данных.
* Разработка пользовательского интерфейса.
* Программирование серверной и клиентской частей.
* Интеграция с базой данных для хранения сообщений, контактов и другой информации.
* Тестирование функционала и безопасности (включая проверку шифрования данных).
* Пилотное внедрение и оценка пользовательского опыта.
* Подготовка и выпуск программной и эксплуатационной документации.

## 6.3. Содержание работ по этапам

Работы, выполняемые на каждом этапе, включают:

* Разработка технического задания. Составляется техническое задание, в котором подробно описываются все требования к программе, её функции, архитектура, интерфейс и требования к безопасности. Важной частью является описание работы с базой данных для хранения почтовых сообщений и контактных данных.
* Проектирование архитектуры системы и базы данных. Детализируется архитектура системы, включая серверную и клиентскую части. Проектируется структура базы данных для хранения писем, вложений, контактов и учётных записей пользователей. Определяется, как будут взаимодействовать компоненты программы (клиент-серверное взаимодействие, интеграция с базой данных).
* Разработка пользовательского интерфейса. На этом этапе разрабатываются макеты интерфейса электронной почты. Интерфейс должен быть интуитивным и удобным для пользователя, включая удобное отображение списка писем, их фильтрацию, удобную работу с контактами и вложениями. Особое внимание уделяется адаптации интерфейса под различные устройства (компьютеры, планшеты, смартфоны).
* Программирование серверной и клиентской частей. Программируется логика работы серверной части: управление почтовыми ящиками, отправка и получение сообщений, работа с базой данных, безопасность данных и их шифрование. Параллельно разрабатывается клиентская часть, отвечающая за отображение интерфейса и взаимодействие с пользователем.
* Интеграция с базой данных. База данных интегрируется с сервером для хранения информации о пользователях, почтовых сообщениях, контактах и вложениях. Определяются методы работы с данными (чтение, запись, обновление, удаление), а также механизмы бэкапа и восстановления информации.
* Тестирование функционала и безопасности. Проводится комплексное тестирование всех компонентов системы. Тестируется корректность работы интерфейса, отправка и получение почты, работа с вложениями, фильтры и другие функции. Также проводится проверка безопасности — тестирование шифрования данных, защищенности от несанкционированного доступа, устойчивости к возможным атакам.
* Пилотное внедрение и оценка пользовательского опыта. Программа внедряется в небольшую группу пользователей для пилотного тестирования. Собираются отзывы и предложения по улучшению работы системы. По итогам этого этапа вносятся правки и оптимизации, если это необходимо.

Подготовка и выпуск программной и эксплуатационной документации. Разрабатываются и выпускаются инструкции для пользователей и администраторов системы. Описываются процедуры работы с программой, её настройки и управления, меры по обеспечению безопасности и восстановлению данных в случае сбоев.

## 6.4. Исполнители

Исполнители проекта по разработке программы «Интерфейс электронной почты» включают в себя команду специалистов, каждый из которых выполняет свою роль для успешной реализации проекта:

* Руководитель проекта — отвечает за общее управление проектом, координацию работы команды и контроль за соблюдением сроков и бюджета. Он также взаимодействует с заказчиком, следит за выполнением технического задания и распределяет ресурсы на разных этапах разработки.
* Аналитик — занимается сбором и анализом требований к программе, изучает пожелания и задачи пользователей. На основе полученной информации он формулирует детализированные требования, которые затем используются для проектирования системы.
* Системный архитектор — разрабатывает архитектуру системы, определяет её структуру и технологические решения. Он отвечает за выбор технологий (например, ASP.NET Core для серверной части, HTML, CSS и JavaScript для клиентской части), проектирование базы данных, а также интеграцию всех компонентов программы.

Программисты:

* Backend-разработчики занимаются разработкой серверной части программы. Они программируют логику обработки почтовых сообщений, взаимодействие с базой данных, обеспечивают безопасность передачи и хранения данных.
* Frontend-разработчики разрабатывают пользовательский интерфейс, обеспечивая удобную работу с почтовыми сообщениями, контактами и вложениями. Они делают процесс взаимодействия с программой максимально комфортным.
* Администратор базы данных – отвечает за проектирование и настройку базы данных, на которой будут храниться все почтовые сообщения, контакты и другая информация. Он следит за производительностью базы данных, оптимизацией запросов и безопасностью хранения данных.
* Специалист по информационной безопасности – разрабатывает и внедряет меры по защите данных, использует методы шифрования, следит за безопасностью аутентификации пользователей и предотвращает утечки данных. Он также контролирует выполнение требований по защите персональных данных.
* Тестировщики – проводят тестирование программы на всех этапах разработки, проверяя её функциональность, корректность работы с почтой, безопасность и производительность. Они выявляют ошибки и уязвимости, после чего передают их разработчикам для исправления. Задача тестировщиков - выявлять ошибки и уязвимости для их последующего исправления.
* Технический писатель – разрабатывает документацию к программе, включая руководства для пользователей и администраторов. Документация описывает процесс работы с программой, настройку системы и меры по обеспечению безопасности.
* Техническая поддержка – после внедрения программы занимается сопровождением пользователей, решением возникающих проблем и обеспечением бесперебойной работы системы.

Важно отметить, что все эти специалисты работают в тесном взаимодействии друг с другом. Результатом их совместной работы становится надежная, удобная и безопасная почтовая программа, которой пользователи могут доверять.

7. Порядок контроля и приёмки

## 7.1. Виды испытаний

Виды испытаний для программы «Интерфейс электронной почты» включают несколько этапов, направленных на проверку функциональности, безопасности и надёжности системы. Основные виды испытаний:

* Функциональное тестирование. Проверяется выполнение всех функциональных требований, описанных в техническом задании. Это включает отправку и получение электронных писем, управление контактами, работу с вложениями, создание папок для организации почты, а также фильтрацию и поиск сообщений.
* Тестирование пользовательского интерфейса (UI). Оценивается удобство и интуитивность интерфейса программы. Проверяется корректное отображение элементов интерфейса на разных устройствах и в разных браузерах, проверяется адаптивность интерфейс.
* Нагрузочное тестирование. Проверяются возможности обработки большого потока писем, число одновременно активных пользователей, а также время отклика системы при высокой нагрузке.
* Безопасность. Проводится тестирование мер защиты данных, включая шифрование сообщений, проверка многофакторной аутентификации и контроль доступа пользователей. Проверяется защита от несанкционированного доступа, утечек данных и других угроз.
* Интеграционное тестирование. Проводится проверка взаимодействия программы с другими системами и компонентами, такими как почтовые сервера, базы данных, сторонние сервисы (например, системы антиспам-фильтрации).
* Модульное тестирование. Проверяются отдельные компоненты системы, такие как обработка писем, отправка вложений, управление контактами. Это помогает выявить ошибки на уровне отдельных функций и модулей программы.
* Регрессионное тестирование. Проводится после внесения изменений в программу для проверки того, что новые изменения не нарушили существующую функциональность.
* Приёмочное тестирование. Проводится перед передачей системы в эксплуатацию заказчику. В ходе приёмочных испытаний проверяется соответствие программы всем требованиям, а также тестируется её производительность в условиях, приближенных к реальной эксплуатации.

Только после успешного прохождения всех видов испытаний программа может быть официально принята и введена в эксплуатацию.

## 7.2. Общие требования к приёмке работы

Общие требования к приёмке работы для программы «Интерфейс электронной почты» направлены на обеспечение качества и соответствия готового продукта заявленным требованиям. Приёмка работ включает несколько этапов и проверок, которые должны подтвердить, что программа полностью готова к эксплуатации и удовлетворяет ожидания заказчика.

Программа должна полностью соответствовать техническому заданию (ТЗ), включая все заявленные функции и характеристики. Приёмка начинается с проверки того, что все функциональные и нефункциональные требования были реализованы.

На этапе приёмки проверяется, что все основные функции программы работают корректно. Это включает отправку, получение и хранение электронных писем, управление контактами и вложениями, а также работу с настройками учётной записи.

Программа проверяется в реальных условиях эксплуатации с использованием данных, максимально приближенных к тем, которые будут использоваться на практике. Это позволяет выявить возможные проблемы с обработкой данных и работой с письмами при реальной нагрузке.

Перед окончательной приёмкой программа должна пройти все необходимые тесты и проверки. Если в процессе тестирования были выявлены ошибки, они должны быть исправлены.

Приёмка включает проверку наличия и качества эксплуатационной и пользовательской документации. Документация должна быть понятной, полной и соответствовать реальной функциональности программы.

Программа должна демонстрировать устойчивую работу без сбоев или ошибок в условиях обычной эксплуатации. Производительность должна соответствовать заявленным показателям, включая время отклика, обработку больших объёмов данных и работу с несколькими пользователями одновременно.

Приёмка программы включает проверку на соответствие требованиям информационной безопасности. Все данные должны быть надёжно защищены при передаче и хранении с использованием современных методов шифрования.

По завершении всех проверок составляется и подписывается акт приёмки-сдачи работ, в котором фиксируется факт соответствия программы всем требованиям, отсутствие критических ошибок и готовность программы к вводу в эксплуатацию.

Приёмка работы проводится только после завершения всех этапов разработки и тестирования, чтобы гарантировать высокое качество конечного продукта и его готовность к использованию.

Заключение

Проект разработки программы «Интерфейс электронной почты» представляет собой комплексное решение, предназначенное для создания удобного и безопасного инструмента для работы с электронной корреспонденцией. В ходе разработки были учтены требования по функциональности, безопасности и удобству использования, что позволило создать продукт, соответствующий современным стандартам.

Основное внимание было уделено защите данных пользователей, высокой производительности системы и удобному интерфейсу, который обеспечивает простой доступ к основным функциям программы, таким как отправка и получение писем, управление вложениями и контактами. Использование шифрования данных и многофакторной аутентификации гарантирует надёжную защиту информации как при её хранении, так и при передаче.

Проект включает несколько стадий разработки, начиная с технического проектирования и заканчивая тестированием и вводом в эксплуатацию. На каждом этапе соблюдались высокие стандарты качества и использовались современные методы разработки и тестирования.

Программа имеет высокую экономическую эффективность благодаря своей доступности для широкого круга пользователей и минимальным затратам на эксплуатацию и поддержку. Это решение удовлетворяет потребности как малых, так и крупных организаций, предоставляя надёжные средства для управления электронной почтой.

Постоянное дополнение функций в соответствии с меняющимися требованиями пользователей. Внедрение новых технологий и инструментов для обеспечения совместимости с современными устройствами и сервисами. Улучшение и обновление программы с учетом изменяющихся требований и тенденций в сфере электронной почты обеспечит актуальность интерфейса.

Таким образом, завершение данного проекта открывает перспективы для дальнейшего развития и совершенствования функционала, адаптации программы под новые требования пользователей и технологические изменения, что делает её актуальной на долгие годы вперёд.