Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Разработка и анализ требований

Студентка: Казакова В.В.

ФИТ 2 курс 8 группа

Преподаватель: Ромыш А.С.

Минск 2025

**Лабораторная работа 4**

Спецификация к проекту

Спецификация требований к ПО для приложения “Ресторан”

1. **Ведение**
   1. **Назанчение**

Данный документ является спецификацией требований к программному обеспечению для приложения «Ресторан». В документе описаны основные требования для разработки приложения, тестирования, внедрения и поддержки системы, обеспечивающей эффективное управление ресторанным бизнесом. Он предназначен для разработчиков, менеджеров проектов, тестировщиков, маркетологов, пользователей и составителей документации.

**1.2 Соглашения, принятые в документах**

* Разные размеры шрифтов для разных типов контента.
* Стиль выделения текста(с помощью значков) для создания иерархии и акцентирования важной информации.

**1.3 Границы проекта**

Приложение «Ресторан» предназначено для упрощения работы как для посетителей, так и для сотрудников заведения. Оно позволяет пользователям бронировать столики, просматривать меню, оформлять заказы и оставлять отзывы, а ресторанам — управлять бронированиями, заказами и обслуживанием клиентов. Приложение должно быть доступно на различных платформах, включая веб-интерфейс и мобильные приложения.

Приложение должно предоставлять следующую функциональность:

**Бронирование столиков:** пользователи могут просматривать доступное время, выбирать нужный столик и делать бронирование онлайн.

**Просмотр меню:** отображение актуального меню с описанием блюд, фотографиями и ценами.

**Отзывы и рейтинги:** пользователи могут оставлять отзывы о блюдах и сервисе, а также оценивать их.

**Уведомления:** отправка уведомлений о подтверждении бронирований, статусе заказов, специальных акциях и предложениях.

**1.4 Ссылки**

Руководство по стилю пользовательского интерфейса (UI Style Guide) - <https://learn.microsoft.com/ru-ru/windows-server/manage/windows-admin-center/extend/guides/ui-text-style-guide>

1. **Общее описание**
   1. **Общий взгляд на продукт**

Приложение «Ресторан» является новым продуктом , разработанным для автоматизации процессов управления ресторанным бизнесом. Оно не является лишь новой версией существующей системы или заменой текущего решения, а представляет собой самостоятельное приложение. Приложение интегрируется с внешними системами для управления заказами, платежными системами и аналитикой, обеспечивая обмен данными через четко определённые интерфейсы.

* 1. **Классы и характеристики пользователей**

Приложение ресторана предназначен для следующих классов пользователей:

**Гость на сайте:**  
 Это пользователь, который впервые заходит на сайт или не зарегистрирован в системе. Гость имеет доступ к базовой информации о ресторане: общему описанию, основному меню, контактным данным и информации о местоположении заведения. Ему доступны просмотр отзывов и ознакомление с акциями, однако для полноценного взаимодействия с сервисом (например, бронирования столиков или оформления заказов) требуется регистрация.

**Частый пользователь (клиент):**  
 Этот класс пользователей характеризуется регулярным использованием приложения и может иметь личный кабинет с сохранёнными предпочтениями. Клиент получает доступ к расширенному функционалу: он может просматривать историю своих заказов и бронирований, накапливать бонусные баллы, участвовать в программах лояльности и получать персонализированные предложения. Для него предусмотрены удобные инструменты повторного заказа, быстрый доступ к любимым блюдам и возможность оставлять отзывы с детальным описанием опыта посещения.

**Администратор:**  
 Администратор обладает **наивысшим уровнем привилегий** в системе. Он отвечает за управление содержимым сайта, обновление меню, обработку заказов и бронирований, а также контроль за отзывами пользователей. Администратор имеет доступ к аналитическим данным и статистике, что позволяет оперативно реагировать на изменения в работе ресторана и принимать решения по оптимизации бизнес-процессов.

**2.3 Операционная среда**

* Аппаратная платформа: веб-серверы, настольные компьютеры и мобильные устройства клиентов.
* Операционные системы: Windows, macOS, Linux, Android.
* Браузеры: Google Chrome, Firefox, Safari, Microsoft Edge.
* Географическое местоположение: приложение доступно пользователям в разных географических регионах по всему миру. Расположение серверов и баз данных может быть разным в зависимости от региональных требований и предпочтений.

**2.4 Ограничения дизайна и реализации**

 Бюджет: Финансовые ограничения влияют на выбор технологий, платформы хостинга и объем доступного функционала, поскольку проект должен укладываться в выделенный бюджет.

 Безопасность: Все данные пользователей, включая личные сведения и платежные реквизиты, должны быть надежно защищены, что требует использования современных методов шифрования и строгого контроля доступа.

 Доступность данных: Ограничения на доступ к информации, например, к географическим данным, могут затруднить разработку функций, зависящих от точных данных о клиентах или партнерах, таких как картографические сервисы и навигация.

 Поддержка различных языков: Реализация многоязычной поддержки может создать сложности с переводом и корректным отображением контента, особенно при использовании автоматических переводов, что увеличивает объем работ.

 Сезонные скидки: Проведение акций и скидок в определенные сезоны требует разработки механизма автоматического применения скидок, корректного отображения промо-кодов и обработки увеличенного объема транзакций, что может усложнить логику ценообразования и систему учета продаж.

**2.5** **Предположения и зависимости**

**Предположения:**

* Предполагается, что пользователи приложения имеют стабильное интернет-соединение и используют современные браузеры или мобильные устройства.
* Предполагается, что ресторан своевременно обновляет информацию о меню, бронированиях и акциях, предоставляя актуальные данные.
* Предполагается, что сезонные пики нагрузки будут предсказуемыми, и резервирование серверных мощностей позволит справиться с увеличенным трафиком.
* Базовый функционал приложения рассчитан на использование в современных браузерах и мобильных устройствах, соответствующих минимальным системным требованиям.
* Разработчики предполагают, что архитектура системы будет масштабируемой, что позволит безболезненно увеличивать функционал при необходимости.

**Зависимости:**

* Приложение зависит от наличия поддерживаемой серверной инфраструктуры или облачных платформ, обеспечивающих высокую доступность и масштабируемость.
* Для работы могут потребоваться сторонние библиотеки и фреймворки (например, установка Microsoft .NET Framework 4.5 или его аналога, если приложение разработано с использованием этой технологии).
* Интеграция с внешними системами (платежные шлюзы, CRM, аналитические сервисы) является критической и зависит от их стабильной работы и доступности API.
* Обеспечение безопасности данных зависит от соблюдения внешних стандартов и нормативных требований, влияющих на методы шифрования и контроля доступа.

1. **Функции системы**
   1. **Бронирование**
      1. **Описание**

Функция управления бронированиями обеспечивает возможность создания, изменения и отмены бронирования столиков в ресторане через приложение. Она позволяет пользователям удобно резервировать столики онлайн, а администраторам — отслеживать и управлять всеми бронированиями в режиме реального времени. Приоритет: высокий.

* + 1. **Функциональные требования**

**Создание бронирования**: Пользователь должен иметь возможность

создать бронирование, указав дату, время, количество мест и, при необходимости, дополнительные пожелания. При неверном формате даты или времени система должна выводить сообщение об ошибке с предложением корректировать ввод.

**Редактирование бронирования**: Пользователь должен иметь возможность изменить детали уже созданного бронирования (дату, время, количество гостей и т.д.) до установленного срока подтверждения. При попытке редактирования бронирования, не подлежащего изменению, система должна уведомить пользователя о невозможности внесения изменений.

**Отмена бронирования**: Пользователь должен иметь возможность

отменить бронирование через интерфейс приложения. В случае ошибки при отмене (например, невозможность связи с сервером) система должна информировать пользователя о проблеме и предлагать повторить операцию позже.

**Проверка доступности бронирования**: Система должна проверять доступность выбранного времени и даты, предотвращая двойное бронирование одного и того же места. Если выбранное время занято, система должна предложить альтернативные варианты или сообщить об ошибке.

**Подтверждение бронирования**: После успешного создания или редактирования бронирования система должна отправить подтверждение пользователю (например, по электронной почте или SMS). В случае невозможности отправки подтверждения, пользователь должен быть уведомлен о необходимости повторной попытки.

**Обработка некорректного ввода:** При вводе некорректных данных (например, неправильный формат телефона или даты) система должна выводить понятное сообщение об ошибке с указанием, какие данные необходимо исправить, и не сохранять некорректное бронирование.

**3.2 Обработка отзывов**

**3.2.1 Описание**

Функция обработки отзывов позволяет пользователям оставлять и просматривать отзывы о сервисе ресторана. Это дает возможность собрать обратную связь, своевременно реагировать на проблемы и улучшать качество обслуживания. Приоритет: средний.

**3.2.2 Функциональные требования**

**Оставление отзыва**: Пользователь должен иметь возможность оставить отзыв о ресторане, выбрав оценку (например, от 1 до 5 звезд) и написав текст комментария. При попытке отправить пустой система должна выводить сообщение с предложением дополнить отзыв.

**Просмотр отзывов**: Пользователь должен иметь возможность просматривать список отзывов, сортированный по дате или оценке. Если система не может загрузить отзывы (например, из-за проблем с сервером), пользователю должно отображаться уведомление о временной недоступности этой функции.

**Модерация отзывов**: Администратор должен иметь возможность просматривать, редактировать или удалять отзывы, которые нарушают правила сервиса (например, содержат неприемлемый контент).

**Фильтрация недопустимого контента**: Система должна автоматически фильтровать ключевые слова и фразы, относящиеся к спаму или нецензурной лексике. При обнаружении подозрительного контента отзыв должен быть помечен для дальнейшей проверки, а пользователь — уведомлён о необходимости соблюдения правил.

**3.3 Управление меню**

**3.3.1 Описание**

Функция управления меню обеспечивает создание, редактирование и удаление позиций в меню ресторана. Она позволяет администраторам обновлять список блюд, цены, описания и изображения, а пользователям – просматривать актуальное меню. Приоритет: высокий.

**3.3.2 Функциональные требования**

**Добавление блюда**: Администратор должен иметь возможность добавить новое блюдо в меню, указав его название, описание, цену и загрузив изображение. При неполном заполнении обязательных полей система должна выводить уведомление об ошибке и запретить сохранение.

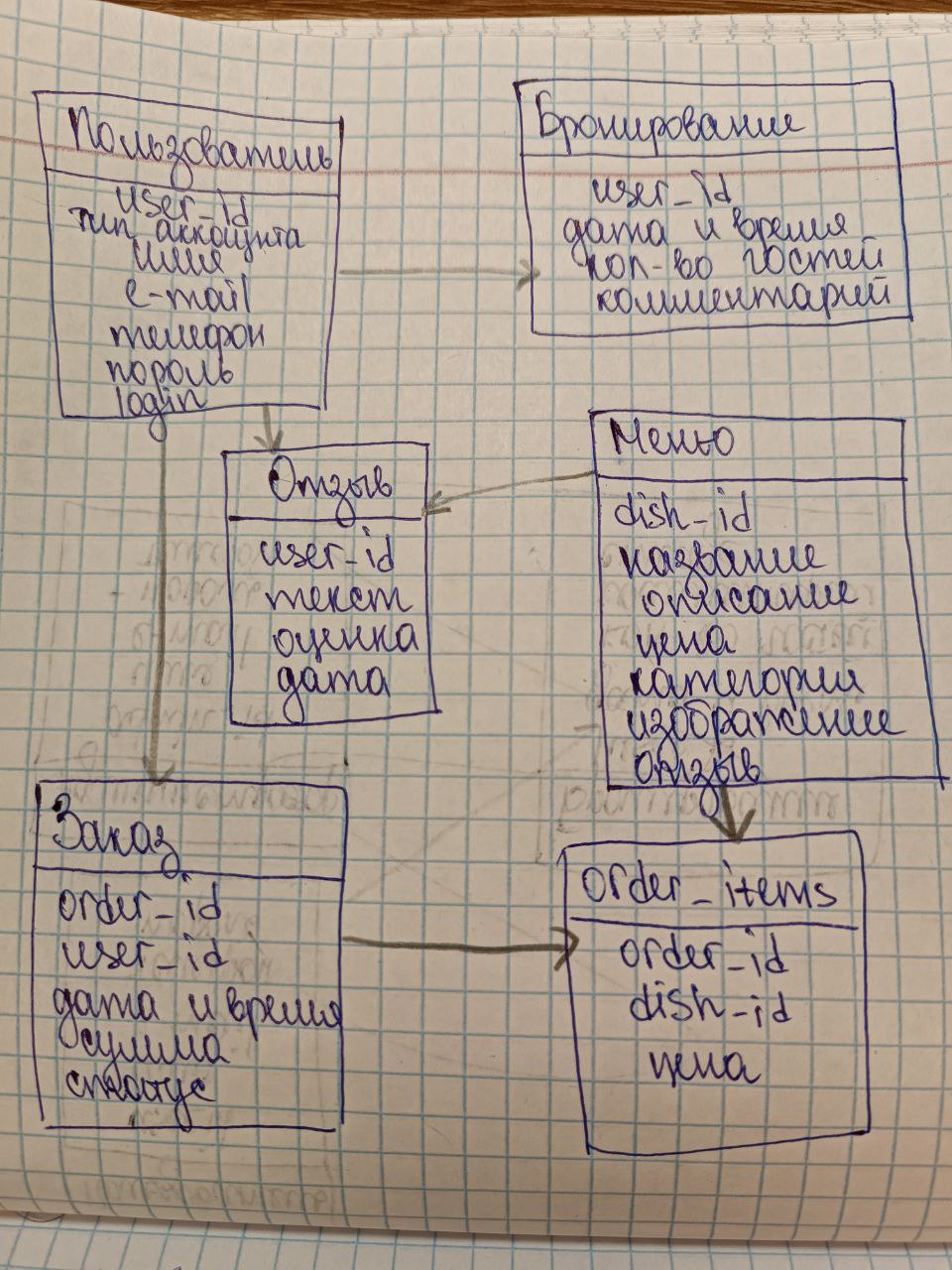
**Редактирование блюда**: Администратор должен иметь возможность изменить информацию о существующем блюде (название, описание, цена, изображение). Если введены некорректные данные, например, отрицательная цена, система должна сообщить об ошибке и не сохранить изменения.

**Удаление блюда:** Администратор должен иметь возможность удалить блюдо из меню. Перед окончательным удалением система должна запрашивать подтверждение действия.

**Просмотр меню:** Пользователи должны иметь возможность просматривать актуальное меню с информацией о блюдах, ценах и изображениях. Если данные не загружаются из-за технической ошибки, пользователю должно отображаться сообщение о временной недоступности функции с предложением попробовать позже.

1. **Требования к данным**
   1. **Логическая модель данных**

Модель данных это визуальное представление объектов и наборов данных, которые будет обрабатывать система, а также отношений между ними.



* 1. **Словарь данных**

Словарь данных определяет основные поля, используемые в системе, их назначение, тип, допустимую длину, формат и ограничения. Данный словарь охватывает ключевые сущности: **Пользователь, Бронирование, Заказ, Меню, Отзыв**, а также служебные поля, обеспечивающие безопасность и структуру данных.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Поле | Описание | Тип данных | Длина | Формат | Допустимые значения |
| user\_id | Уникальный идентификатор пользователя | Integer |  | 1001а | Только положительные числа |
| price | Цена блюда | |  | | --- | | Decimal |  |  | | --- | |  | |  | 350.00 | > 0 |
| dish\_name | Название блюда | String | String | «Цезарь с курицей» | Без специальных символов |
| guests\_count | Количество гостей | Integer | 20 | 2 | От 1 до 20 |
| booking\_date | Дата и время бронирования | DateTime | 16 | 2025-04-20 19:00 | Только будущее время |
| phone | Номер телефона | Char | 15 | +375295556677 | Только цифры, допустим "+" в начале |
| email | Адрес электронной почты | Char | 100 | user@mail.com | Email-формат, уникальное  значение |
| password | Вводимый пароль | Сhar | 8–64 | qwerty123@ | Не менее 8 символов, комбинация букв, цифр и символов |

* 1. **Отчеты**

***Отчёт по заказам***

* **Назначение:** Анализ количества заказов, суммы и популярности блюд за выбранный период.
* **Фильтры:** Дата, категория блюд, способ оплаты, статус заказа.
* **Сортировка:** По дате заказа, по сумме, по количеству заказанных позиций.
* **Суммирование:**
  + Общее количество заказов
  + Общая сумма продаж
  + Средний чек

***Отчёт по бронированиям***

* **Назначение:** Анализ активности клиентов по бронированию столиков.
* **Фильтры:** Дата, день недели, количество гостей, статус бронирования.
* **Сортировка:** По дате, по количеству гостей.
* **Суммирование:**
  + Общее количество бронирований
  + Среднее количество гостей на бронирование
  + Процент отмен/подтверждений

***Финансовый отчёт***

* **Назначение:** Контроль выручки и расходов по заказам.
* **Фильтры:** Дата, тип оплаты, скидки.
* **Сортировка:** По сумме, по дате.
* **Суммирование:**
  + Общая выручка
  + Сумма скидок
  + Сумма по каждому способу оплаты

**4.5 Получение, целостность, хранение и утилизация данных**

***Получение и обслуживание данных:***

* Данные в систему поступают через интерфейсы: веб-приложение.
* Все вводимые пользователем данные проходят **валидацию** на клиенте и сервере для исключения ошибок и вредоносного ввода.

***Целостность данных:***

* Все критически важные поля (например, ID, суммы заказов, даты бронирований) проходят проверку на уникальность, корректность формата и логическую согласованность.

***Хранение данных:***

* Все данные хранятся в централизованной СУБД с регулярным резервным копированием.
* Персональные и платёжные данные пользователей шифруются.
* Конфиденциальная информация (email, телефоны, пароли) хранится в зашифрованном виде и недоступна напрямую.

***Утилизация и удаление данных:***

* **Удалённые записи** помечаются как "удалённые" и могут быть восстановлены в течение ограниченного времени (например, 30 дней).
* Старые и неактивные данные (например, архивные заказы и бронирования) могут быть перемещены в **архивные таблицы** по заданному расписанию.
* **Кэшированные данные** (например, меню, популярные блюда) обновляются с интервалом или при изменениях, и очищаются автоматически.

***Резервное копирование и контрольные точки:***

* Резервные копии базы данных создаются **ежедневно** и хранятся минимум **7 дней**.
* Поддерживаются **контрольные точки** и **журналы транзакций**, позволяющие восстановить систему до последнего стабильного состояния.

***Политики хранения:***

* Данные активных пользователей и операций хранятся в рабочей БД.
* Архивы хранятся в отдельной защищённой структуре не менее 12 месяцев (или согласно требованиям законодательства).

1. **Требования к внешним интерфейсам**
   1. **Пользовательские интерфейсы**

***Общие стандарты интерфейса:***

* Приложение разрабатывается с учётом **адаптивности**: корректная работа на экранах от 360×640 (мобильные устройства) до 1920×1080 (десктопы).

***Стилевые рекомендации:***

* **Цветовая палитра**: соответствует бренду ресторана (например, тёплые оттенки, ассоциирующиеся с едой).
* **Шрифты**:
  + Заголовки — полужирный, 18–24 px.
  + Основной текст — нейтральный, 14–16 px.
* **Иконки и кнопки**: в едином стиле, с использованием SVG или шрифтовых иконок
* **Изображения**: формат JPG/PNG.
* **Кнопки**: единообразные по стилю (скруглённые углы, контрастный текст), стандартные действия — «Заказать», «Забронировать», «Отменить», «Назад», «Сохранить».

**Навигация и взаимодействие:**

* **Фиксированное меню навигации** доступно на всех страницах (главная, меню, бронирование, заказы, профиль).
* **Кнопка справки** и поддержка отображаются в виде иконки «?», доступной в углу экрана.

**Сочетания клавиш (для десктоп-версии):**

| **Действие** | **Сочетание клавиш** |
| --- | --- |
| Открыть главное меню | Alt + M |
| Перейти в профиль | Alt + P |
| Открыть помощь | Alt + H |
| Оформить заказ | Ctrl + Enter |

**Стандарты отображения сообщений:**

* Уведомления о действиях (успех, ошибка, предупреждение) — всплывающие, не перекрывают основной контент.
* Сообщения об ошибках — красные, понятные, указывают на конкретное поле.
* Подсказки — серым текстом внутри полей до ввода.

**Валидация данных:**

* Все формы проходят валидацию как на клиенте, так и на сервере.
* Проверки:
  + Email — формат name@example.com
  + Телефон — только цифры, длина 10–15 символов
  + Пароль — не менее 8 символов, буквы и цифры
  + Обязательные поля — подсвечиваются, если не заполнены

**Локализация и адаптация:**

* Интерфейс изначально проектируется с поддержкой **многоязычности**.
* Поддержка перевода дат и валют по региону пользователя.

**Специальные возможности (доступность):**

* Цветовая схема — контрастная, не зависит только от цвета (для дальтоников).
* Минимальный размер шрифта — 14px, с возможностью масштабирования.
* Навигация по клавиатуре — реализована на всех экранах.

**5.2** **Интерфейсы ПО**

*Опишите связи продукта и других компонентов ПО, в том числе другие приложения, базы данных, операционные системы, средства, библиотеки, веб-сайты и интегрированные серийные компоненты. Укажите назначение, форматы и содержимое сообщений, данных и контрольных значений, обмен которыми происходит между компонентами ПО. Опишите соответствия между входными и выходными данными между системами и все преобразования, которые должны происходить с данными при перемещении между системами. Опишите службы, необходимые внешним компонентам ПО, и природу взаимодействия между компонентами.*

#### ****Связь с платёжной системой:****

* **Назначение:** Обеспечение онлайн-оплаты заказов пользователями.
* **Формат сообщений:** JSON по HTTPS.
* **Пример запроса:**

{"order\_id": 2001,

"amount": 145000,

"currency": "byn"}

Пример ответа:

{"transaction\_id": "tr\_001",

"status": "success",

"paid\_at": "2025-04-20T12:45:00Z"

}

 **Контрольные значения:**

* amount передаётся в копейках/центах.

 **Преобразования:**

* Дата оплаты конвертируется в локальное время для отображения.

#### ****Связь с базой данных (PostgreSQL / MySQL)****

**Назначение:** Хранение и управление всеми данными: пользователи, заказы, блюда, бронирования, отзывы.

**Формат взаимодействия:** SQL-запросы .

**Пример запроса:**

SELECT \* FROM menu WHERE user\_id = 101;

**Контрольные значения:**

* Используются первичные и внешние ключи (user\_id, order\_id, dish\_id).
* Все поля валидируются при записи (тип, длина, ограничения).

**Преобразования:**

* Автоматическое округление числовых данных (например, цена — до 2 знаков).
* Хэширование паролей перед записью.

**Общий доступ:**

* таблицы пользователей, заказов, меню и бронирований;
* кэш популярных блюд для быстрого отображения в клиентском приложении;

**Нефункциональные требования и ограничения**

* **Безопасность:**
  + Вся передача данных между компонентами — только через **HTTPS**.
* **Производительность:**
  + Максимальное время ответа API — **до 2 секунд** для обычных операций.
  + Для запросов с фильтрацией или отчётами — не более **5 секунд**.
* **Надёжность:**
  + Система должна обеспечивать **99.5%** времени доступности (SLA).
  + Все ошибки логируются с уровнем серьезности (INFO/WARN/ERROR).

5.3 **Интерфейсы оборудования**

***ПК, ноутбуки:***

* **Тип устройства:** Настольные или портативные компьютеры.
* **ОС:** Windows 10/11, macOS 12+, Linux.
* **Взаимодействие:** Через веб-браузер с доступом к административной панели (на базе HTML/JS).
* **Протокол:** HTTP/HTTPS.
* **Управление:** Клавиатура, мышь, сенсорный экран.
* **Особенности:** Обеспечение корректного отображения на экранах от 1280×720 и выше.

***Мобильные устройства клиентов и персонала:***

* **Тип устройства:** Смартфоны, планшеты (Android 10+, iOS 13+).
* **Взаимодействие:** Через мобильное приложение или адаптивную веб-версию.
* **Протокол:** HTTPS.
* **Управление:** Сенсорный ввод, голосовой ввод.
* **Поддержка устройств:** Поддерживаются стандартные разрешения от 360×640.

***QR-сканеры / камеры***

* **Назначение:** Сканирование QR-кодов со стороны клиента (например, при оплате, регистрации или бонусах).
* **Тип подключения:** встроенные камеры мобильных устройств.
* **Протокол:** HID или через Web API камеры.
* **Взаимодействие:** Приложение принимает отсканированные данные и автоматически обрабатывает их (например, открытие заказа).

5.4 **Коммуникационные интерфейсы**

**Функции взаимодействия:**

* **Электронная почта:**
  + Отправка уведомлений о бронированиях и заказах пользователям.
  + Формат сообщений: текст или HTML.
  + Поддержка шаблонов и переменных (имя пользователя, дата бронирования и т.п.).
* **Веб-интерфейсы :**
  + Возможность интеграции с веб-браузером для удалённого доступа администратора.
  + Отображение статистики, управление меню, отзывами и бронированиями.
* **Электронные формы:**
  + Формы регистрации, входа, бронирования и добавления отзыва.
  + Проверка данных на клиенте и сервере.
  + Валидация обязательных полей и форматов (email, телефон, количество гостей и т.п.).

**Форматы сообщений:**

* **JSON** — основной формат передачи данных между клиентом и сервером.
* **SQL-запросы** — для взаимодействия между серверной частью и базой данных.

**Особенности безопасности:**

* Авторизация и аутентификация обязательны для всех операций, связанных с личными данными и управлением бронированиями.
* Используется HTTPS для всех соединений.
* Пароли хранятся в виде хэшей.

**Скорость передачи данных:**

* Среднее время ответа сервера на стандартные запросы — до 2 секунд.
* Быстрая синхронизация при работе с формами заказов и бронирований.

**Механизмы согласования и синхронизации:**

* Обновление данных о бронированиях в реальном времени
* Использование временных меток (updated\_at) для разрешения конфликтов при одновременном редактировании данных.

**Ограничения:**

* Форматы данных строго определены — все поля имеют чётко заданные типы и длину.
* Валидация всех входящих данных (на стороне клиента и сервера).
* Поддержка только современных браузеров и мобильных устройств с актуальными версиями ОС.

6.**Атрибуты качества**

**6.1 Удобство использования**

***Лёгкость изучения:***

* Интерфейс должен быть интуитивно понятным даже для новых пользователей.
* Первичное обучение не должно превышать 5 минут.
* Используются знакомые элементы управления (иконки, кнопки, поля) и понятные названия действий.

***Простота использования:***

* Взаимодействие с приложением должно требовать минимального количества шагов (например, бронирование столика — не более 3 кликов).
* Все часто используемые функции (меню, заказ, отзыв) — доступны с главного экрана.
* Интерфейс адаптирован под мобильные и настольные устройства.

***Предотвращение ошибок:***

* Все формы снабжены встроенной валидацией (например, недопустимый email, пустые поля).
* Кнопки "Удалить" или "Отменить" снабжены подтверждением действий.

***Восстановление после ошибок:***

* Пользователь получает понятные сообщения об ошибках (например, “Не удалось сохранить заказ. Попробуйте ещё раз.”).
* В случае сбоя соединения данные сохраняются локально и отправляются повторно.
* Система предоставляет возможность отменить или повторить последние действия (например, откат отмены брони).

***Эффективность взаимодействия***

* Все действия выполняются быстро — интерфейс должен отвечать на действия пользователя менее чем за 0.5 секунды.
* Поддерживается автозаполнение и сохранение пользовательских данных (например, имя и номер телефона при повторном бронировании).

***Специальные возможности***

* Цветовая палитра рассчитана с учётом пользователей с нарушениями цветового восприятия (не только цвет, но и иконки/подписи).
* Шрифты масштабируются и читаемы (размер не менее 14px).
* Навигация возможна с клавиатуры (без мыши).

**6.2 Производительность**

**Время отклика пользовательского интерфейса:**

* Интерфейс должен реагировать на действия пользователя (нажатия кнопок, переключения вкладок) **не более чем за 0.5 секунды.**
* Загрузка основных страниц (например, меню, форма бронирования) должна происходить **не более чем за 2 секунды** при нормальной нагрузке.

**Масштабируемость:**

* Система должна поддерживать **одновременную работу не менее 200 пользователей** без значительного ухудшения производительности.

**Работа с базой данных**

* Запросы к БД по ключевым действиям (создание, чтение, обновление, удаление) должны выполняться **в пределах 200–500 мс.**
* Долгие операции (например, генерация отчётов) могут выполняться в фоне, с отображением индикатора загрузки и сохранением результата.

**Обновление данных в реальном времени**

* Изменения в бронированиях и статусах заказов должны отображаться на клиенте **с задержкой не более 1 секунды.**

**6.3 Безопасность**

**Контроль доступа:**

* Все пользователи проходят **аутентификацию** (логин/пароль, токены доступа).
* Чувствительные функции (например, редактирование меню, удаление брони) доступны только администраторам.

**Хранение и передача данных:**

* Все данные передаются по **защищённому каналу (HTTPS).**
* Пароли пользователей хранятся в виде **хэшей**.
* Персональные данные (email, телефон, имя) шифруются или маскируются при необходимости.

**Безопасность ПО и серверов:**

* Серверы защищены **межсетевыми экранами**, антивирусным ПО и средствами мониторинга активности.

**Защита от атак:**

* Реализованы механизмы защиты от:
  + SQL-инъекций
  + XSS (межсайтового скриптинга)
  + CSRF (подделки межсайтовых запросов)
  + Брутфорса (ограничение количества попыток входа)

**Конфиденциальность пользователей:**

* Пользователи имеют право:
  + просматривать свои данные;
  + редактировать их;
  + удалить свой аккаунт (или запросить удаление);
  + отключить уведомления.

**6.**4 **Техника безопасности**

**Защита от потери данных:**

* Регулярное **автоматическое резервное копирование** базы данных (не реже 1 раза в сутки).
* Хранение резервных копий на **отдельных серверах или в облаке** с ограниченным доступом.
* Внедрение функции **восстановления системы** из последнего стабильного состояния при сбое.

**Безопасность пользователей:**

* Исключение отображения критических ошибок пользователю — вместо этого выводятся безопасные уведомления с инструкциями.
* Операции, связанные с платежами или удалением данных, сопровождаются **двойным подтверждением**.
* В случае подозрительной активности (например, множественные неудачные попытки входа) инициируется **автоматическая блокировка аккаунта** или уведомление администратора.

**Ответственность за конфиденциальные данные:**

* Все пользователи предупреждаются о правилах обработки и защиты своих данных при регистрации.
* В случае утечки данных предусмотрены **процедуры уведомления пострадавших** и устранения причин.

**6.5 Другие атрибуты качества**

**6.5.1 Доступность**

 Приложение должно быть доступно **24/7**, с допустимым временем простоя не более **0.5% в месяц.**

 Веб-интерфейс должен корректно отображаться при низкой скорости соединения и на экранах от 360px шириной.

**6.5.2 Надежность**

 Система должна корректно работать при кратковременных сбоях сети или потере связи с внешними сервисами.

 В случае сбоя должны быть предусмотрены механизмы **повторной попытки и восстановления состояния.**

 Ошибки логируются с уровнями критичности (INFO/WARN/ERROR).

**6.5.3 Устойчивость**

 Приложение должно справляться с нештатными ситуациями: перегрузка сервера, неправильный ввод, внезапная остановка процесса.

 Используется автоматическая обработка ошибок и отображение понятных сообщений для пользователя без раскрытия технических деталей.

**6.5.4 Документированность**

Приложение сопровождается:

* документацией по установке;
* инструкцией для администратора;
* пользовательским руководством (клиентская часть).

**7. Требования по интернационализации и локализации**

**Языковая поддержка:**

* Интерфейс приложения должен быть локализуемым.
* Поддерживаются как минимум два языка:
  + **Русский** (по умолчанию)
  + **Английский** (в будущем — возможность расширения)

**Форматирование данных:**

* Форматы отображения должны меняться в зависимости от региона пользователя:
  + **Дата:**
    - ДД.ММ.ГГГГ — для RU
    - MM/DD/YYYY — для US
  + **Время:**
    - 24-часовой формат — для большинства стран
    - 12-часовой формат (AM/PM) — для США, Канады и др.
  + **Числа и цены:**
    - Разделители тысяч и десятичные символы зависят от языка (например, 1 000,00 ₽ vs 1,000.00 RUB)
  + **Телефонные номера:**
    - Отображение и валидация с учётом кода страны (например, +375 (\_\_\_) \_\_\_-\_\_-\_\_ для РБ)

**Валюта:**

* Поддержка отображения и расчётов в различных валютах:
  + ₽ (рубль), $ (доллар), € (евро) и др.
* Автоматическое отображение валютного символа в зависимости от локали.
* Возможность настройки валюты по умолчанию в админ-панели.

**Правила ввода имени:**

* Поля имени и фамилии должны допускать различные алфавиты (латиница, кириллица и др.).
* Не должно быть жёсткой привязки к шаблону имени (некоторые культуры не используют фамилии).
* Поддержка двойных имён, дефисов, апострофов.

**8. Остальные требования**

**8.1Юридические и нормативные требования**

Приложение обязано соответствовать:

* Закону Республики Беларусь от 07.05.2021 № 99-З "О защите персональных данных"
* Пользователь должен давать согласие на обработку персональных данных (отдельно и явно).
* Оператор персональных данных (владелец системы) обязан:
  + обеспечивать безопасное хранение данных;
  + использовать персональные данные только с конкретной целью, указанной при сборе;
  + по запросу пользователя предоставить доступ к его данным и возможность удаления.
* В системе должно быть реализовано:
  + хранение логов согласий;
  + форма отзыва согласия;
  + политика конфиденциальности (на белорусском и/или русском языке).

**8.2 Интеллектуальная собственность и авторские права**

* Используемые изображения, иконки, шрифты и текстовый контент:
  + должны быть лицензированы или собственными;
  + запрещено использование контента без разрешения правообладателя;

**Приложение А.**

ПО (Программное обеспечение)- Программа или система, реализующая функциональность приложения «Ресторан».

БД (База данных)- Хранилище, в котором систематизировано хранятся все данные пользователей, заказов, меню, отзывов и бронирований.

API (Application Programming Interface )— интерфейс прикладного программирования; способ взаимодействия между клиентской частью приложения и сервером.

HTTP/HTTPS - Протокол передачи гипертекста (Hypertext Transfer Protocol / Secure) — используется для обмена данными между клиентом и сервером.

JSON(JavaScript Object Notation) — лёгкий формат обмена данными между клиентом и сервером в виде текста.

SQL(Structured Query Language) — язык структурированных запросов, используемый для взаимодействия с базой данных.

Платёжный шлюз - Сервис-посредник, обеспечивающий безопасную передачу платёжных данных между пользователем и банком.

Пользователь-Клиент ресторана, использующий приложение для бронирования столика, оформления заказа или написания отзыва.

Администратор - Пользователь с расширенными правами доступа, управляющий содержимым приложения и данными пользователей.

Резервное копирование - Процесс создания копии данных для восстановления в случае сбоя или потери информации.

Бронирование - Процесс предварительного резервирования столика пользователем на конкретную дату и время.

Заказ -Выбор и оформление блюд из меню пользователем, с указанием количества и суммы.

Меню - Список блюд, напитков и других позиций, доступных для заказа.

Отзыв - Текстовая оценка, оставляемая пользователем по итогам посещения или заказа.

Авторизация - Процесс проверки прав доступа пользователя (обычный или админ).

Аутентификация - Проверка личности пользователя через логин и пароль или токен.

Локализация - Адаптация интерфейса, дат, валют, времени и прочего под региональные стандарты.

Масштабируемость - Способность системы справляться с увеличением нагрузки (например, в праздничные дни).

**Приложение Б**

Диаграмма «Сущность–Связь»

• Использована для построения логической структуры базы данных.

• Представлены ключевые сущности: Пользователь, Бронирование, Заказ, Меню, Отзыв, Администратор и их связи.

Диаграмма включает:

o связи один-ко-многим (например, пользователь → бронирования);

o связь многие-ко-многим (например, заказ ↔ блюда через промежуточную таблицу).

Диаграмма переходов состояния

• Применима к объектам со сменой состояний, например:

o Бронирование: создано → подтверждено → отменено;

o Заказ: оформлен → принят → готовится → завершён / отменён.

• Показывает, какие действия или события вызывают переходы между состояниями.

Диаграмма вариантов использования (Use Case)

• Отображает взаимодействие пользователей и администраторов с системой.

• Может включать:

o сценарии: «Забронировать столик», «Сделать заказ», «Оставить отзыв», «Изменить меню»;

o роли: Гость, Постоянный клиент, Администратор.