**Список нефункциональных требований к системе управления гостиницей**

**1. Атрибуты качества (Группа Runtime)**

**1.1. Производительность**

* Время отклика системы на любое действие пользователя (клиента или администратора) не должно превышать 2 секунд при стандартной нагрузке.
* Система должна выдерживать пиковую нагрузку до 50 одновременныхпользователей (из которых 45 — клиенты, просматривающие сайт и бронирующие номера, и 5 — администраторы, работающие в панели управления).
* Поиск и отображение списка свободных номеров по заданным параметрам (даты, тип) должен выполняться не более чем за 3 секунды.
* Операция подтверждения бронирования должна завершаться не более чем за 5 секунд.

**1.2. Надежность**

* Система должна автоматически сохранять данные в процессе бронирования для предотвращения их потери в случае сбоя сети или браузера.
* В случае программного сбоя система должна корректно восстановиться без потери данных транзакций, выполненных за последние 5 минут.
* Все финансовые операции (подтверждение брони, расчет стоимости) должны быть защищены от повторного выполнения.

**1.3. Безопасность**

* Доступ в панель администратора должен быть защищен аутентификацией по логину и надежному паролю (требования: минимум 8 символов, заглавные и строчные буквы, цифры).
* Все пароли должны храниться в базе данных в хешированном виде.
* Передача персональных данных клиентов (ФИО, паспортные данные) между браузером пользователя и сервером должна быть защищена с использованием протокола HTTPS**.**
* Должна быть реализована система разграничения прав: клиенты имеют доступ только к своим данным, а администраторы — ко всем функциям системы.

**1.4. Удобство использования**

* Интерфейс сайта для клиента должен быть интуитивно понятен.
* Веб-интерфейс должен быть адаптивным и корректно отображаться на экранах мобильных устройств, планшетов и компьютеров.
* Все элементы управления должны иметь четкие и понятные подписи. Пользователь должен получать информативные сообщения о статусе своих действий (например, "Бронь успешно подтверждена", "Необходимо заполнить все поля").
* Процесс первого бронирования для нового клиента не должен требовать предварительного изучения инструкций.

**1.5. Масштабируемость**

* Архитектура системы должна позволять увеличивать вычислительную мощность (вертикальное масштабирование) для обработки роста количества пользователей на 50% без изменения кода.
* Система должна быть спроектирована с возможностью в будущем разделения на микросервисы (например, отдельный сервис для работы с бронированиями, отдельный — для клиентов) для горизонтального масштабирования.

**2. Атрибуты качества (Группа Design Time)**

**2.1. Расширяемость**

* Система должна быть спроектирована таким образом, чтобы добавление нового типа номера или новой категории скидки требовало минимальных изменений в коде (не более 2-3 человеко-дней).
* Архитектура должна позволять в будущем легко интегрировать систему с внешними сервисами (например, с онлайн-букинг-системами, сервисами email-рассылок или мессенджерами для уведомлений).

**2.2. Сопровождаемость**

* Исходный код должен быть хорошо документирован и соответствовать принятым стандартам.
* Система должна быть построена по модульному принципу, где логически связанные функции (управление номерами, бронирование, работа с клиентами) выделены в отдельные модули.
* Среднее время на исправление критической ошибки не должно превышать 8 часов с момента ее обнаружения.

**3. Ограничения**

* **База данных:** В качестве системы управления базами данных должна использоваться **SQLite**.
* **Браузерная совместимость:** Клиентская часть должна корректно работать в последних версиях браузеров Google Chrome, Mozilla Firefox, Safari и Microsoft Edge.
* **Мобильность:** Административная панель должна быть доступна только с персональных компьютеров. Функционал для клиентов должен быть доступен с любых устройств.

**4. Бизнес-правила**

* **Правило скидок:** Постоянным клиентам (сделавшим более 5 бронирований) автоматически предоставляется персональная скидка в 10%. Данная скидка может суммироваться с акционными предложениями.
* **Правило бронирования:** Бронирование номера считается подтвержденным только после получения клиентом системного email-уведомления с уникальным кодом брони.
* **Правило данных:** Система должна хранить полную историю всех бронирований и заселений в течение 5 лет с момента выезда гостя.

**5. Внешние интерфейсы**

* Система должна предоставлять REST API для основных операций: проверка свободных номеров, создание брони, получение данных о клиенте.
* Система должна интегрироваться с SMTP-сервером для автоматической отправки email-уведомлений клиентам (подтверждение брони, напоминание о заезде).

**6. Условия разработки ПС**

* **Система контроля версий:** Весь исходный код должен храниться в репозитории Git на платформе GitHub.
* **Документация:** По окончании разработки должна быть предоставлена документация, включающая руководство пользователя для клиента и администратора, а также техническое описание API.
* **Сроки:** Разработка системы должна быть выполнена в течение +-4месяцев.

**7. Условия тестирования ПС**

* **Тесты:** Не менее **80%** критически важного кода (логика бронирования, расчеты скидок) должно быть покрыто модульными и интеграционными тестами.
* **Нагрузочное тестирование:** Перед сдачей в эксплуатацию система должна пройти успешное нагрузочное тестирование, подтверждающее выполнение требований к производительности при 50 одновременных пользователях.
* **Приемочное тестирование:** Финальное приемочное тестирование (UAT) должно быть проведено с участием реальных будущих администраторов системы.