

КОНКУРСНОЕ ЗАДАНИЕ ДЛЯ КОМАНДНОГО МОДУЛЯ

Командообразование между компетенциями

«Цифровые возможности для бизнеса»

и «Администрирование отеля»

Финала Чемпионата по профессиональному мастерству «Профессионалы» 2025 г

*г. Санкт-Петербург*

2025 г.

**Командный модуль. Система оперативной оценки гостевого опыта (Guest Experience Pulse)**

*Время на выполнение модуля – 4 часа.*

Конкурсное задание выполняется в командах в составе: два конкурсанта компетенции «Администрирование отеля» и два конкурсанта компетенции «Цифровые возможности для бизнеса». Для выполнения командного модуля по результатам жеребьевки формируются 5 команд по 4 конкурсанта.

**Задания:**

Ключевая цель:

Разработать и внедрить технологическое решение для моментального сбора обратной связи от гостей о качестве обслуживания в различных зонах отеля, с автоматизированным анализом результатов для принятия оперативных управленческих решений.

Ключевые требования к продукту:

• Для гостя: процесс опроса должен быть максимально простым, быстрым (не более 2 минут) и ненавязчивым.

• Для администрации: инструмент должен предоставлять актуальные данные в наглядном виде (дашборд), позволяя выявлять проблемы и тенденции в режиме, близком к реальному времени.

Подробное техническое задание:

а) Разработка опросника (Зона ответственности: Администраторы + Программисты)

• Структура: Опросник должен быть сегментирован по гостевым зонам (например: Reception, Housekeeping, Restaurant, SPA, Concierge).

• Типы вопросов:

Оценка зоны: 1-2 обязательных вопроса с рейтингом по шкале от 1 до 10 или по 5-звездочной системе (NPS-подобный). Пример: "Оцените, пожалуйста, чистоту и комфорт вашего номера".

Выявление проблемы: Необязательный вопрос с множественным выбором для плохих оценок (1-3). Пример (если гость поставил низкий балл за номер): "Что именно вас не устроило?" с вариантами: "Чистота", "Запах", "Работа техники", "Постельное белье".

Открытый фидбек: Необязательное текстовое поле "Ваши комментарии или предложения".

• Логика: если гость ставит высокий балл (8-10/5 звезд), он переходит к следующей зоне или завершает опрос. Если балл низкий, ему сразу предлагаются уточняющие вопросы по этой зоне.

б) Определение места и времени для опроса (Зона ответственности: Администраторы)

• Канал распространения: Telegram-бот как наиболее универсальный и не требующий скачивания дополнительного приложения (гость может быть не из РФ и не иметь VK/WhatsApp).

• Механика получения ссылки:

1. Вариант А (Основной): QR-код на ключ-карте или в номере ("Оцените свой отдых – отсканируйте код"). QR ведет на запуск бота.

2. Вариант Б (Дополнительный): Ссылка в финальном e-mail после выезда.

• Время: Гость может пройти опрос в любое удобное время during his stay, начиная со второго дня проживания (дать время познакомиться с сервисом). Это повышает актуальность отзывов.

в) Технологическое решение и анализ (Зона ответственности: Программисты)

• Бэкенд:

Реализация Telegram-бота на Python (библиотека python-telegram-bot) или др.

Проектирование простой базы данных (например, SQLite или PostgreSQL) для хранения ответов. Структура: id, guest\_id (анонимный), timestamp, zone, rating, issue\_tag, comment.

Настройка веб-сервера.

• Фронтенд (Дашборд):

Разработка простого веб-интерфейса для администрации (логин/пароль).

Дашборд должен включать:

* общий средний балл по отелю и по зонам за выбранный период (день, неделя, месяц).
* визуализация: графики и диаграммы (столбчатые – по средним оценкам, круговые – по распределению проблем).
* таблица с последними негативными отзывами (фильтр по оценке < 6) с комментариями.
* фильтры по дате и по зонам обслуживания.

Технологии: можно использовать простой фреймворк и библиотеки для графиков.

Распределение ролей и план на 4 часа

Команда:

• Администратор 1: "Главный по качеству" (Product Owner)

• Администратор 2: "Гость-адвокат" (UX/UI Researcher)

• Программист 1: "Бэкенд-мастер" (Backend Developer)

• Программист 2: "Дашборд-волшебник" (Frontend Developer)

План работы (с примерным хронометражем):

Час 1: Планирование и дизайн (Вся команда)

• (15 мин) Общее обсуждение: все вместе финализируют понимание задачи на основе этого ТЗ.

• (30 мин) Проектирование опросника: администраторы продумывают точный список зон и вопросов для каждой. Программисты консультируют по технической реализации логики (ветвление вопросов).

• (15 мин) Проектирование БД: программисты рисуют на доске/листке простую схему таблицы для хранения ответов. Согласовывают с админами.

Час 2: Разработка прототипа (Параллельная работа)

• Программист 1 (Бэкенд): Создает Telegram-бота с помощью BotFather, настраивает базовый вебхук. Пишет основной скелет бота, который принимает сообщения. Начинает работу с БД (создает таблицу).

• Программист 2 (Фронтенд): Поднимает простейший веб-сервер. Создает страницу входа для дашборда. Набрасывает HTML-каркас будущих графиков.

• Администраторы: Готовят финальный текст для всех вопросов и ответов бота на русском и английском (или целевом языке). Тестируют BotFather, продумывают сценарий общения гостя с ботом (user flow).

Час 3: Интеграция и наполнение (Параллельная работа + синхронизация)

• Программист 1: Реализует всю логику опроса в боте: переход между зонами, запись оценок, обработку уточняющих вопросов, сохранение всего в БД.

• Программист 2: Подключает библиотеку графиков (например, Chart.js). Пишет код на сервере, который выгружает данные из БД (например, средние оценки по зонам) и передает их на фронтенд для отрисовки.

• Администраторы: Помогают с тестированием: проходят опрос через бота, проверяют, корректно ли все сохраняется. Формулируют требования к дашборду: "Нам важно видеть вот это и вот это".

Час 4: Тестирование, подготовка презентации, репетиция (Вся команда)

• (30 мин) Финальное тестирование: Вся команда по очереди проходит опрос, проверяет, что данные появляются в дашборде. Ищут баги.

• (20 мин) Подготовка презентации: Создают несколько слайдов:

Проблема (почему это важно?).

Решение (наш продукт: бот + дашборд).

Демо: живые примеры работы бота и дашборда (главный экран!).

Технологический стек (например, Python, Telegram Bot API, SQLite, Flask, Chart.js).

Выгоды для отеля (оперативность, наглядность, повышение лояльности гостей).

• (10 мин) Распределение ролей для выступления: Кто представляет проблему, кто демонстрирует бота, кто показывает дашборд, кто рассказывает о технической реализации.

Что должно быть готово к презентации через 4 часа:

1. Работающий Telegram-бот: Ссылка на бота. Любой человек может перейти по ней и пройти весь опрос. Данные сохраняются.

2. Работающий веб-дашборд: Ссылка на сайт. Логин/пароль для входа (можно простой, например, admin/admin). На дашборде отображаются тестовые данные, введенные командой: графики и список отзывов.

3. Короткая и ясная презентация (3-4 слайда), объясняющая ценность продукта.

4. Слаженное выступление команды (по 2-3 минуты на человека), где каждый освещает свою часть работы.