

Тема: Система информационной поддержки процесса обслуживания клиентов

Уважаемый Иван Иванович, для того чтобы мы точно поняли ваши потребности и создали удобную для вас систему, просим ответить на несколько вопросов, сгруппированных по темам.

Блок 1. Общие организационные сведения

Предоставьте основные реквизиты вашей организации: полное и краткое наименование, юридический адрес и контактный телефон.

ООО УК «ИНФОТЕЛЛ», ИНН: 7811476996, 191023, Санкт-Петербург г, Итальянская ул, дом № 12, корпус литер Е, оф.б

Блок 2. Данные для учета и контроля процесса обслуживания клиентов

2. Какие поля данных будут необходимы для описания заявки (например, тема заявки, описание проблемы, приоритет)?

Поля должны быть настраиваемыми. Обязательное поле – Наименование.

- Проясните, откуда будут заводиться эти заявки: пользователи вручную через интерфейс, почта, телефония, импорт из файлов или из существующей базы данных?

Пользователь вручную

- Если у вас уже есть информация/база — опишите её (есть ли существующая база данных; какие файлы: CSV, XLSX, XML, JSON; структура данных и примерные объёмы). Нужно понять, требуется ли миграция данных и в каком формате ожидать исходные данные.

Есть база, в которой содержится информация об исполнителях. Информация может быть выгружена в excel файл формата:

	A	B	C	D	E
1	№ пп	Фамилия	Имя	Отчество	Должность
2	1	Иванов	Иван	Иванович	Инженер
3					

Остальные данные вводятся вручную.

- Если база данных отсутствует — планируется ли старт с нуля?

см выше

- Уточните, какие дополнительные метаданные нужны для заявки (например, канал обращения, идентификатор клиента, договор, SLA, вложения и т.п.).

Поля настраиваются при формировании заявки

3. Какие этапы обработки заявки будут использоваться в системе (например, "Новая", "В работе", "Ожидает подтверждения", "Закрыта")?

Этапы работ настраиваются при формировании заявки

- Какой ожидаемый результат работы системы по каждой заявке (что считается успешным завершением: закрытие с отчётом, согласование заказчиком, передача в другой отдел и т.п.)? Опишите желаемые выходные состояния и бизнес-правила завершения.

Завершение всех обязательных работ по заявке.

Какие обязательные и необязательные этапы может проходить заявка?

Этапы работ должны настраиваться при формировании заявки

Какие данные об исполнителях необходимо хранить (ФИО, специализация, контактные данные и т.д.)?

ФИО, должность

6. Каким образом будет назначаться исполнитель на заявку (автоматически, вручную)?

Автоматически из числа свободных исполнителей. Необходимо обеспечить соответствие должности исполнителя требованиям к квалификации исполнителя, а также равномерную загрузку исполнителей.

7. Какие временные ограничения необходимо учитывать для каждого этапа обработки заявки?

Настраиваются при формировании заявки

8. В каком формате необходимо формировать отчеты по работе системы (например, количество заявок, среднее время обработки, рейтинг исполнителей)?

Число обработанных заявок за отчетный период, объем выполненных работ каждым исполнителем.

Блок 3. Интеграция

9. С какими другими системами должна быть интегрирована данная система (например,

Интеграция не требуется. Необходимо предусмотреть возможность выгрузки данных в файл excel. При этом поля к выгрузке должны настраиваться.

- Нужно уточнить: интеграция со смежными системами обязательна или желательна? Если интеграция не требуется — отметьте.

см выше

- Укажите, какие именно смежные системы уже есть (наименования, версии) и будут ли они доступны для интеграции.

см выше

- Для каждой смежной системы опишите, какие данные должны быть получены/переданы и в каком формате они сейчас хранятся (API/REST, SOAP, файлы CSV/XLSX/XML, база данных, очередь сообщений и т.п.).

см выше

- Уточните требования к частоте синхронизации (реального времени, пакетно/раз в день и т.д.) и требования к безопасному соединению (VPN, TLS, VPN+API-ключи и т.п.).

по требованию пользователя.

Блок 4. Численность персонала

11. Сколько сотрудников будут одновременно работать с системой?

около 80

- Уточните ориентировочно численность по ролям: сколько исполнителей (операторов), сколько руководителей/менеджеров (кто формирует состав этапов и назначает исполнителей), сколько администраторов и т.д.

Численность не фиксирована, часто изменяется

Укажите ожидаемую пиковую нагрузку (максимальное количество одновременных пользователей) и типичную ежедневную активность.

максимальная нагрузка до 110 подключений к системе. типовая 80

12. Планируется ли в будущем существенное увеличение их числа?

Существенное не планируется

Блок 6. Безопасность и соответствие требованиям

Укажите, будет ли система обрабатывать конфиденциальную или особо важную информацию (например, коммерческую тайну, персональные данные), для которой требуется реализация строгих механизмов контроля доступа и защиты.

Персональные данные

Существуют ли специфические требования, отраслевые стандарты или внутренние регламенты компании (например, в области защиты персональных данных, коммерческой тайны, хранения информации или отчетности), которым должна строго соответствовать разрабатываемая система?

Специфических нет.

Блок 7. Платформа для разработки

15. Для разработки системы необходимо выбрать программное обеспечение.

Наша рекомендация - Visual Studio Code (VS Code) — это мощный, легкий и очень популярный кроссплатформенный редактор кода, который благодаря своей архитектуре расширений фактически функционирует как полноценная IDE.

Для разработки системы обслуживания клиентов, которая, вероятно, будет включать в себя выбором благодаря следующим преимуществам:

- исключительная расширяемость: главное преимущество VS Code — его огромная экосистема расширений. Можно установить расширения для Python (обеспечивающие IntelliSense, отладку, форматирование кода), для работы с Git (встроенная поддержка контроля версий), для баз данных, Docker, а также для различных веб-фреймворков (таких как React, Angular, Vue), что критически важно для комплексной системы.

- интегрированные инструменты: поставляется с интегрированным терминалом, мощным отладчиком для множества языков (включая Python), и встроенной поддержкой Git, что значительно упрощает рабочий процесс разработки, отладки и управления исходным кодом.

- высокая производительность: относительно легок и быстр, что позволяет эффективно работать даже на средних по мощности машинах, при этом предлагая функционал, сравнимый с полноценными IDE.

- универсальность: подходит для работы с несколькими языками и технологиями одновременно, что идеально для проектов, где есть и Python-бэкенд, и JavaScript/HTML/CSS-фронтенд.

Вы можете использовать любое ПО, соответствующее требованиям законов РФ и Устава в части лицензирования и позволяющее разработать Систему, соответствующую ожиданиям Заказчика. При необходимости Исполнитель за свой счет обеспечивает обучение сотрудников Заказчика работе с системой до уровня уверенного пользователя.