### **Описание задания**

Компания «Ваш компьютерный мастер» использует конфигурацию «Управление IT-фирмой» от стороннего поставщика. Вам поставлена задача добавить в конфигурацию модуль для работы с заявками на обслуживание клиентов и модуль для расчёта управленческой заработной платы.

Для добавленных объектов, реквизитов, методов используйте префикс ВКМ\_. Если меняется программный код существующих методов, изменения должны быть выделены комментариями.

Все добавленные объекты должны быть размещены в новой служебной подсистеме ВКМ\_ДобавленныеОбъекты. В добавленных объектах не должно быть ошибок, выявляемых встроенным механизмом проверки 1C:EDT.

**Обслуживание клиентов**

Необходимо планировать обслуживание сотрудниками оборудования и программ клиентов.

Процесс работы с заявками построен следующим образом:

1. Клиент звонит менеджеру и оставляет заявку на работу специалиста с указанием проблемы, для решения которой нужен специалист.
2. Менеджер ищет свободное время и планирует заявку на это время.
3. При планировании телеграм-бот оповещает специалистов о появлении новой заявки.
4. Специалист в назначенное время в офисе клиента или удалённо проводит необходимые работы.
5. Специалист получает подписанный лист учёта рабочего времени или скан/фото листа учёта от клиента. В листе учёта перечисляются виды работ, выполненные специалистом, и фиксируется количество часов к оплате. В дальнейшем документ будет являться подтверждением проведения работ.
6. Специалист вносит в информационную систему информацию о выполненных работах, количество фактически потраченных часов на каждый вид работы, количество часов к оплате клиенту за каждый вид работы и прикрепляет к документу скан/фото листа учёта рабочего времени.

Все объекты, связанные с обслуживанием клиентов, должны располагаться в новой подсистеме Обслуживание клиентов.

**Алгоритм решения задачи:**

1. Добавьте документ ОбслуживаниеКлиента
2. В документ добавьте реквизиты:
   * Клиент
   * Договор
   * Специалист
   * Дата проведения работ
   * Время начала работ (план)
   * Время окончания работ (план)
   * Описание проблемы
   * Комментарий
3. В документ добавьте табличную часть ВыполненныеРаботы с колонками:
   * Описание работ
   * Фактически потрачено часов
   * Часы к оплате клиенту
4. Реализуйте форму документа и форму списка с разумным размещением полей, подключите к формам подсистему Подключаемые команды
5. Добавьте возможность хранить присоединенные к документу файлы
6. Создайте роль для работы с документом
7. Добавьте регистр накопления для хранения информации о суммах задолженности клиента ВыполненныеКлиентуРаботы
   * Измерения - Клиент, Договор
   * Ресурсы - Количество часов, Сумма к оплате
8. Реализуйте проверку при проведении документа, что выбран договор с типом абонентская плата и что дата документа лежит между датой начала и датой окончания действия договора
9. Реализуйте обработку проведения по регистру накопления. Сумма к оплате должна рассчитываться исходя из ставки часа, указанной в договоре
10. Добавьте документ в подсистему Обслуживание клиентов
11. Для оповещения через Телеграм создайте константы:
    * токен управления телеграм-ботом
    * идентификатор группы для оповещения
12. Реализуйте отдельный общий модуль для интеграции с телеграм-ботом.
13. Создайте справочник Уведомления телеграм-боту для отправки с нулевой длиной кода и наименования и реквизитом Текст сообщения (строка бесконечной длины)
14. При записи документа Обслуживание клиента, если документ записывается первый раз или если дата, время или специалист изменились, добавьте в справочник сообщение с текстом, описывающим изменения.
15. Добавьте регламентное задание, которое будет обходить элементы справочника Уведомления, отправлять и после успешной отправки удалять элементы справочника.

# Интеграция с Телеграм

## Подготовка

### Создание бота

1. Напишете в Телеграм [https://t.me/BotFather](https://github.com/netology-code/fonecmid-diplom/blob/main/tasks/BotFather) команду "/start"
2. Напишите команду "/newbot"
3. Выберите имя вашего бота, которое будут видеть пользователи
4. Выберите идентификатор вашего бота (он должен заканчиваться на "bot")
5. Скопируйте токен, который пришлет BotFather и сохраните его в константу

### Создание группы

1. Создайте группу в Телеграм
2. Добавьте в группу только что созданного бота
3. Назначьте боту права администратора

### Получите идентификатор группы

1. Напишите любое сообщение в группу
2. С помощью браузера или Postman выполните запрос https://api.telegram.org/bot[ВашТокен]/getUpdates. В полученном JSON найдите идентификатор группы, в которой получено сообщение, и сохраните его в константу.

## Отправка сообщений

1. Создайте общий модуль "Телеграм"
2. В модуле реализуйте метод, который получает на вход текст сообщения
3. В рамках метода:
   * создайте защищённое соединение с сервером "api.telegram.org"
   * создайте HTTP-запрос к ресурсу /bot[ВашТокен]/sendMessage
   * добавьте заголовок "Content-Type" со значением "application/json"
   * подготовьте структуру с полями:
     + "chat\_id" - идентификатор группы из константы,
     + "text" - текст сообщения
   * установите JSON строку, полученную из структуры, в качестве тела HTTP-запроса
   * отправьте запрос с помощью метода "POST"
   * проверьте, что получет ответ с кодом состояния 200, если код состояния отличный от 200, получите тело ответа как строку и запишите в журнал регистрации информацию об ошибке.