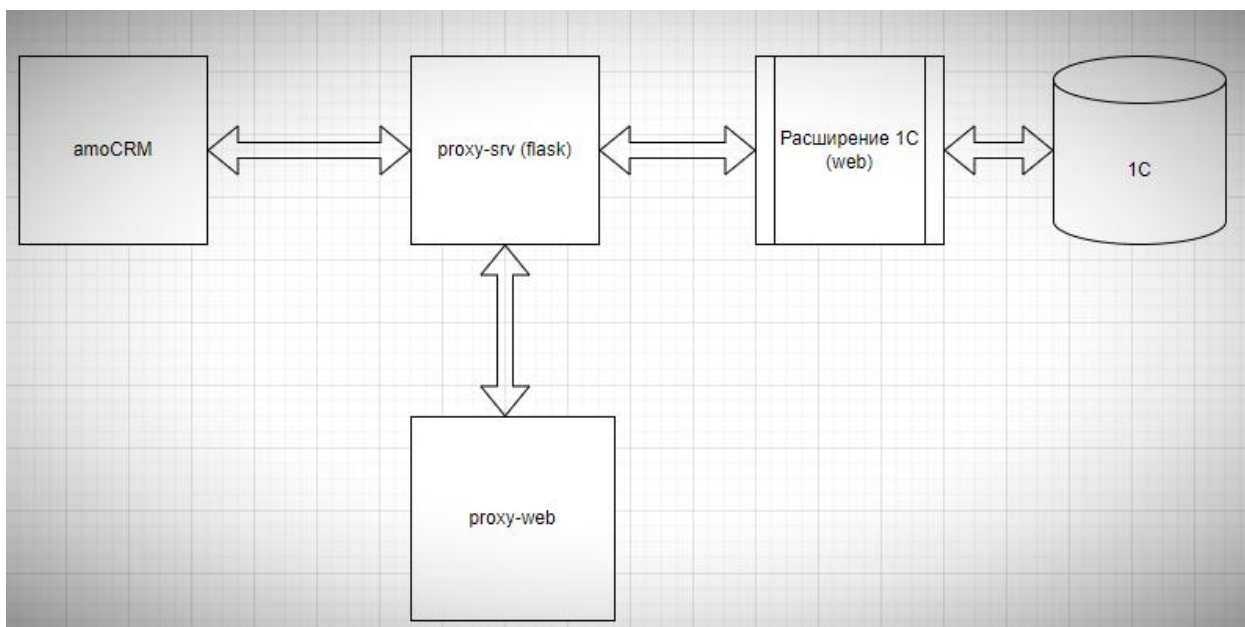


## Серверное взаимодействие 1С и amoCRM



На блок схеме показана структура интеграции amoCRM с базой 1С

- Widget
- Proxy-srv
- Proxy-web
- Расширение 1С

### Widget

Widget полностью написан на JavaScript, занимается логикой расстановки элементов в интерфейсе, валидации данных и взаимодействие с proxy-srv.

Как только пользователь открывает карточку лида, widget генерирует разметку на странице и вешает события на элементы «Дата записи», «Список врачей», «Время записи» и кнопку «Зарегистрировать запись к врачу».

Как только пользователь сгенерировал одно из перечисленных событий, отправляется запрос на proxy-srv, после чего происходит его обработка.

При выборе «Дата записи» собирается json следующего вида

```
let dataTo1C = {  
  "date": date,  
  "token": "00000000-0000-0000-0000-000000000000",  
}
```

- Date – то, что выбрал пользователь
- Token – это token авторизации на proxy-srv

После получения ответа от proxy-srv, обработчик js заполняет данные врачей и свободного времени записи.

После записи лида ему присваивается id, с этим идентификатором будет создан документ в 1С.

Пример json для создания документа.

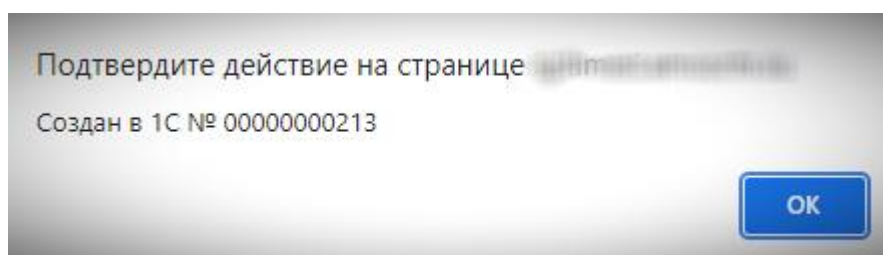
```
let dataTo1C = {  
  "token": "Jm...E47r",  
  "numberAmo": idLead,  
  "date": self.choiceDate.value,  
  "doctor": self.inputDoctor.value,  
  "time": self.inputTime.value,  
  "customer" : customer,  
  "tel": tel,  
  "gender": genderKod,  
  "DateofBirth": dateofBirth  
}
```

- Token – секретный ключ
- Date – дата записи к врачу
- Doctor – Врач
- Time – Время записи к врачу
- Customer – Клиент
- Gender – Пол клиента
- DateofBirth – дата рождения клиента

Обязательные поля: token, date, doctor, time.

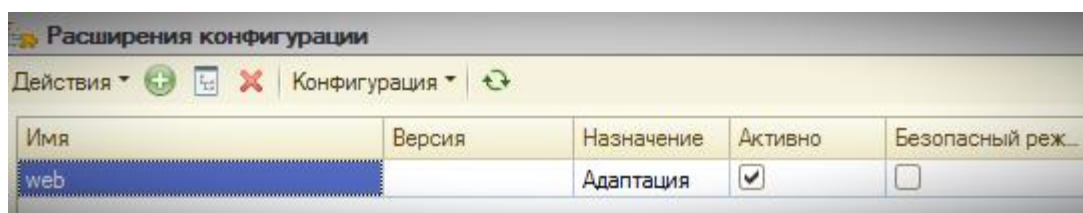
Если получен успешный ответ, в alert присвоится номер документа, если произошла какая-то ошибка на сервере 1С или на проху, то в результате будет показано содержимое ошибки.

Пример успешного создания документа.

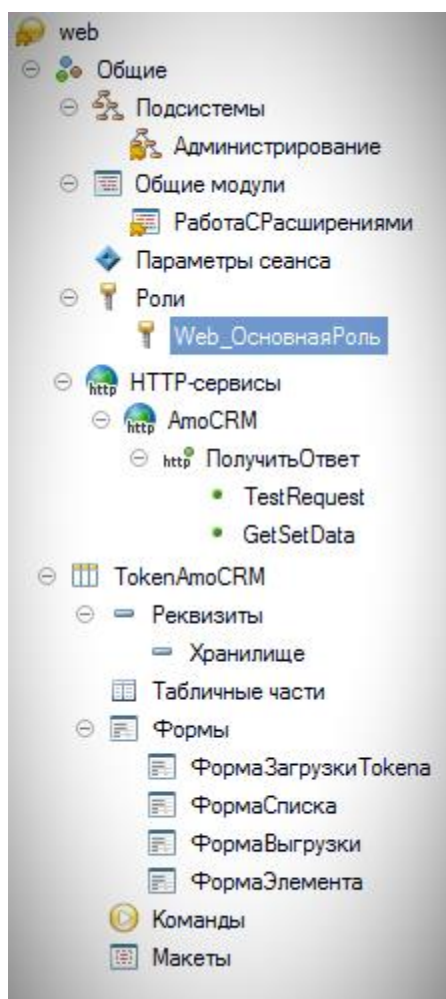


## Расширение 1С web

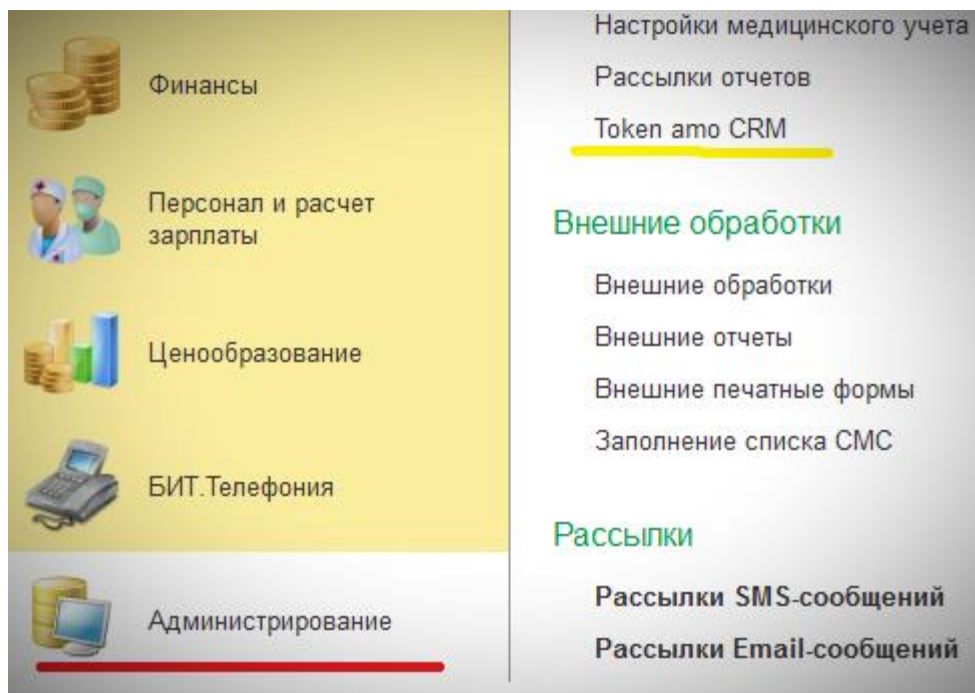
В базе 1С было создано расширение, которое позволяет, взаимодействует с web-сервисом без доработки основной конфигурации.



## Структура расширения



Экспортирована подсистема «Администрирование» для работы со справочником «TokenAmoCRM»



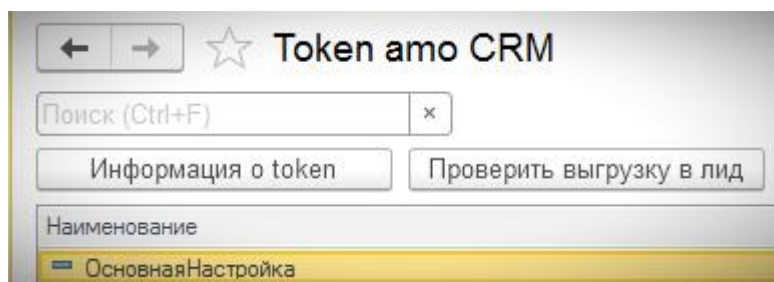
Добавленная роль «Web\_ОсновнаяРоль» имеет полные права на все объекты расширения.

Сервис AmoCRM занимается обслуживанием входящих GET, POST запросов

- TestRequest – принимает GET-запрос и возвращает «ок» с кодом 200
- GetSetData – занимается приемом POST-запросов. В текущей версии занимается обработкой данных «врач», «свободное время» и созданием/обновлением документа «Заявка».

## Справочник TokenAmoCRM

Справочник необходим для работы с API amoCRM без посредников. Когда происходит проведения документа «Оказание услуг», вызывается процедура отправки суммы оплаты от клиента в лид. Процедура, которая расположена в модуле менеджера справочника. Справочник предназначен для хранения одной записи, взаимодействие с элементом напрямую не происходит.



Для того чтоб открыть «основную настройку» нужно нажать кнопку «Информация о token».

В шапке имеются 3 кнопки:

- Загрузить token – в случаях, когда переопубликуывается или обновлен widget, в этом случае меняются ключи: client id, client secret, code
- Обновить token – так как token действует 24 часа, а дальше он становится не актуальным, нужно воспользоваться кнопкой «обновить». В низу формы есть дата и время последнего обновления token. При каждом успешном обновлении, дата смещается на момент загрузки.
- Сохранить token – сохраняет внесенные изменения.

Так выглядят данные в widgete:

Описание

Ключи и доступы

Выданные доступы

Редактировать

Секретный ключ:

ID интеграции:

Код виджета:

szaryynlvwerfczaetac7lhjb8hjjnmhcy0lhhoi

Код авторизации (действителен 20 минут)

Каждые 24 часа запускается процедура обновления токена.

## Proxy server

Решаемые проблемы с помощью данного Proxy server приложения:

- трафик между amoCRM и Proxy server зашифрован с помощью самоподписанного сертификата
- ограничено прямое взаимодействие в 1С из внешнего интернета.

Proxy server делится на две части **api** и **web-интерфейс**.

**1) Web-интерфейс** необходим для настройки взаимодействия с 1С, просмотр логов обработанных соединений.

После авторизации на сайте доступны 3 раздела для просмотра и управления:

- logs
- users
- 1c settings

### Раздел logs

Данный раздел выводит логи POST-запросов, время когда он был сделан, на какой url он был отправлен, данные которые пришли и статус обработки.

На одной странице логов содержится 10 строк, по этой причине логи разделены на страницы.

6	46.191.235.245	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.110 Safari/537.36	/API/get_doctor_time	True	{'doctors': ['Ефимова Е.В. [8:55]', 'Ишаева М.И. [8:00,8:05,8:10,8:25,8:30,8:35,8:40,8:45,8:50,8:55,9:00,9:05,9:10,9:15,9:20,9:25,9:30,9:35,9:40,9:45,9:50,9:55]']}	2021-12-26 06:08:08
7	46.191.235.245	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.110 Safari/537.36	/API/get_doctor_time	True	{'doctors': ['Ефимова Е.В. [8:15,8:20,8:25,8:30,8:35,8:40,8:45,8:50,8:55,9:00,9:05,9:10,9:15,9:20,9:25,9:30,9:35,9:40,9:45,9:50,9:55,10:00,10:05,10:10,10:15,10:20,10:25,10:30,10:35]']}	2021-12-26 11:13:02

## Раздел users

Форма показывает пользователей заведенных в БД, которые могут ходить в веб-интерфейс. Сверху таблицы пользователей есть форма для создания нового пользователя. Заполняем логин, пароль и подтверждение пароля, если валидация прошла успешно, то пользователь заведен. Автоматически страничка перезагрузится и в таблице появится созданный пользователь.

См скрин.

The screenshot displays a web interface for user management. At the top, there is a red header bar. Below it, the section is titled "Создать пользователя" (Create user). This section contains three input fields: "Логин" (Login), "Пароль" (Password), and "Подтверждение пароля" (Confirm password). A green button labeled "СОЗДАТЬ" (CREATE) is positioned to the right of the password fields. Below the form, the section is titled "Список пользователей" (List of users). This section contains a table with the following data:

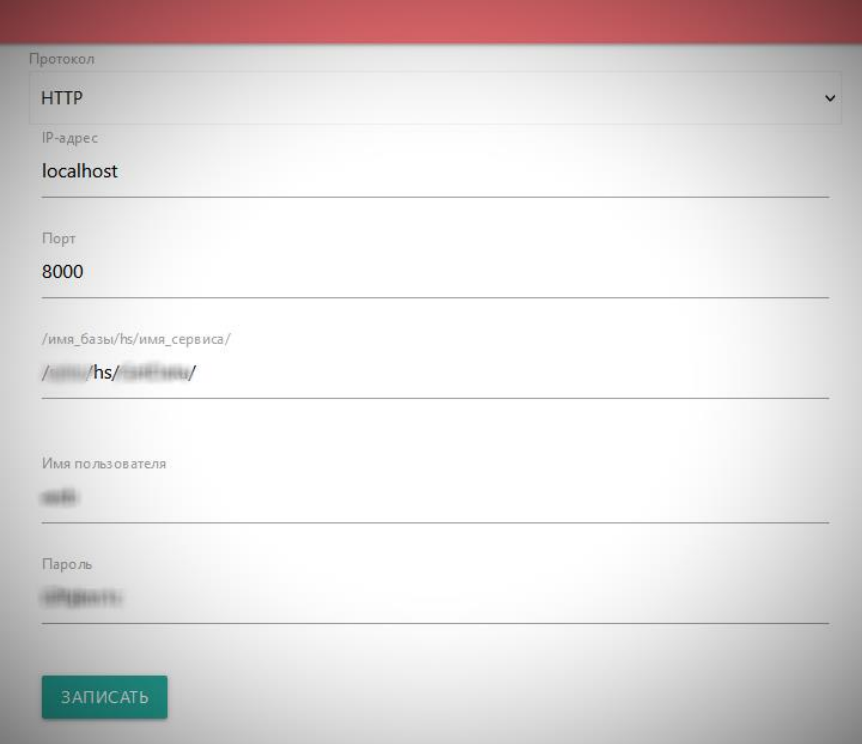
№	Имя	Дата создания
1	root	2021-12-26 05:55:08



### ***Раздел 1c settings.***

В этом разделе выводится настройка с именем «1С». По умолчанию основные параметры заполняются при создании БД, но в этой форме их можно переопределить.

Поля приведенные на скрине понятные, объяснения не требуют. Обратите внимание на поле URL, сверху указано как оно должно быть заполнено!



Протокол

HTTP

IP-адрес

localhost

Порт

8000

/имя\_базы/hs/имя\_сервиса/

/hs/

Имя пользователя

admin

Пароль

admin

ЗАПИСАТЬ

## 2) Реализация API

Реализованы два метода

- API/get\_doctor\_time
- API/create\_document

Каждый из этих обработчиков проверяет token. Если он не равен эталону, хранящемуся в БД, то такой запрос не будет обрабатываться.

Каждый запрос прошедший проверку записывается в таблицу logs, для отслеживания проксирования запросов.

get\_doctor\_time — обрабатывает и пересылает post-запрос в 1С для получения списка врачей и свободного времени для записи

create\_document - обрабатывает и пересылает post для создания документа «Заявка».

Если в теле post-запроса не будут заполнены необходимые параметры, вернется ошибка «error format Json», если токен не верный, то «Invalid token».

В приложение реализована следующая структура БД

- Logs — таблица содержащая запросы от amoCRM
- Users – список пользователей кому можно работать с web-интерфейсом приложения
- Token — токен для прохождения проверки post-запросов
- Settings — хранится настройка подключения в 1С
- TokenSession — при успешной авторизации в web-интерфейсе создается токен, который присваивается пользователю. Токен действует 60 дней.

Для корректной работы приложения нужны стартовые настройки, этим занимается процедура initDefaultSettings. Она проверяет, есть ли минимально-необходимые настройки для функционирования приложения:

- Users — root пользователь
- Token — токен, для взаимодействия с amoCRM
- Settings — для подключения к 1С

Если какой-либо из этих настроек нет в БД, то функция создаст их и заполнит значениями по умолчанию, которые присутствуют в коде.

Файл БД расположен рядом с приложением и имеет следующее наименование «DB.db».

В случае его отсутствия запустится процедура создания новой БД, после этого будет запуск initDefaultSettings.