

# Helix-v21-20230808\_093929

## Системные настройки

Для запуска необходимо включить переадресацию:

```
sudo a2enmod rewrite
sudo service apache2 restart
```

Заказчик должен настроить доменное имя для https соединения с пробросом на 443 порт сервера [MGERM](#) и предоставить сертификаты для настройки SSL Engine в Apache на сервере [MGERM](#).

На сервере [MGERM](#) в настройках виртуальных хостов Apache должен быть файл helix.conf

```
<VirtualHost *:81>
    ServerAdmin support@mgerm.ru
    DocumentRoot /var/www/mgerm25/_external_services
    <Directory />
        Options +FollowSymLinks -Indexes
        AllowOverride None
    </Directory>
    <Directory /var/www/mgerm25/_external_services>
        Options +FollowSymLinks -Indexes
        AllowOverride All
        Order allow,deny
        allow from all
    </Directory>
    ErrorLog ${APACHE_LOG_DIR}/external_services_error.log
    LogLevel error
    CustomLog ${APACHE_LOG_DIR}/external_services_access.log combined
</VirtualHost>

<VirtualHost *:443>
    SSLEngine on
    SSLCertificateFile /etc/ssl/1med39.ru.crt
    SSLCertificateKeyFile /etc/ssl/private.key
    SSLCertificateChainFile /etc/ssl/GlobalSign_ca.crt
    ServerAdmin support@mgerm.ru
    DocumentRoot /var/www/mgerm25/_external_services
    <Directory />
        Options +FollowSymLinks -Indexes
        AllowOverride None
    </Directory>
```

```
<Directory /var/www/mgerm25/_external_services>
    Options -Indexes
    AllowOverride All
    Order allow,deny
    allow from all
</Directory>
ErrorLog ${APACHE_LOG_DIR}/external_services_error.log
LogLevel error
CustomLog ${APACHE_LOG_DIR}/external_services.log combined
</VirtualHost>
```

В параметрах SSLCertificateFile указываем сертификат, выпущенный заказчиком, SSLCertificateKeyFile - соответствующий ключ, SSLCertificateChainFile - нужный сертификат.

В папке conf проекта **MGERM** должен быть конфигурационный файл helix.ini.

```
nano /var/www/mgerm25/conf/helix.ini
```

вставляем:

```
clientId =
clientSecret =
redirectUri = url
urlAuthorize = https://auth-mdo.helix.ru/connect/authorize
urlAccessToken = https://auth-mdo.helix.ru/connect/token
urlResourceOwnerDetails = https://auth-mdo.helix.ru/
scopes = mis catalog_service offline_access
urlApi = https://mdo.helix.ru/
contractID =
authorizeUri = url
```

В свойства redirectUri и authorizeUri ставим адрес с доменным именем + /\_income/callback/oAuthRedirect.php, полученный от заказчика, например:

[https://mgerm.clinica-boli.ru/\\_income/callback/oAuthRedirect.php](https://mgerm.clinica-boli.ru/_income/callback/oAuthRedirect.php)

Указать clientId clientSecret и contractID, внося данные для подключения от Хеликс.

**Внимание!** В логине и пароле не должно быть скобок! Если они есть, нужно просить выдать их заново без скобок. Скобки вызывают ошибку чтения конфигурационного файла. При этом выводится сообщение: "Can't read /conf/helix.ini file properly!"

Так же в случае непонятной ошибки с подключением к API Хеликса необходимо убедиться, что файл \_services\lib\Helpers\LeagueReplace\AbstractProvider.php скопирован в ib\vendor\league\oauth2-client\src\Provider\ с замещением исходного файла.

## Настройки cron

Необходимы для обновления контракта загрузки результатов:

```
sudo crontab -e
```

Вставляем:

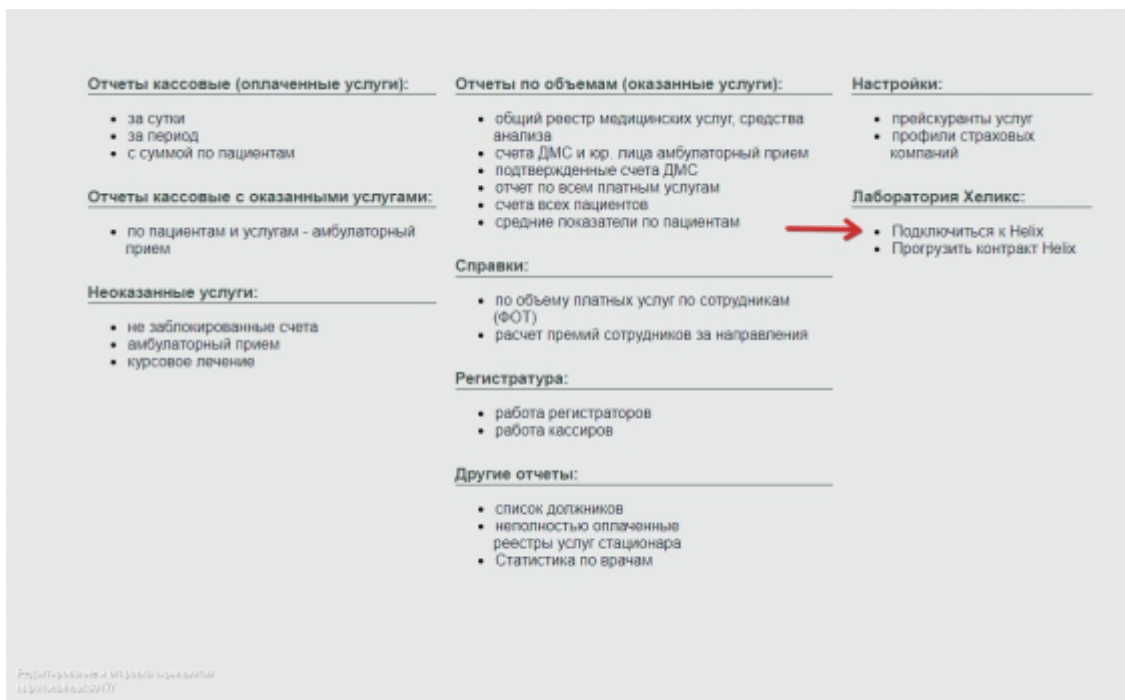
```
0 */12 * * * curl  
"http://localhost/_services/Laboratory/Helix/Invoice.loadContract?no_connection=1"  
0 * * * * curl  
"http://localhost/_services/Laboratory/Helix/Invoice.getAndStoreResult?no_connection=1"
```

## Настройки MGERM

Добавляем системного пользователя и запись результата анализа. В системной переменной "Внешняя лаборатория" ставим helix.



И жмем ссылку правой кнопкой мыши и выбираем “открыть в новой вкладке”:



Авторизуемся, используя данные от Хеликс: пользователь на Identity Server (Username, Password). После успешной авторизации на открывшемся экране ничего не меняем и жмем кнопку “Yes, allow”.

Загрузку контракта рекомендуется провести через консоль для проверки работы автоматической загрузки:

```
curl http://localhost/_services/Laboratory/Helix/Invoice.getAndStoreResult?no_connection=1
```

Корректный ответ: {«status»:«success»}

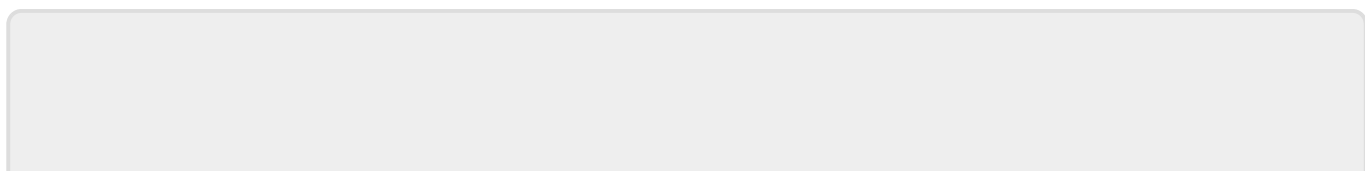
Далее необходимо провести тестовый заказ, используя диапазон тестовых кодов, переданных Хеликсом и передать идентификатор заказа службе поддержки. Идентификатор можно узнать в базе данных в epm\_records.jsonData, используя recordID заказа, например:

```
{«taskID»: «5ef19a56-a1c4-44cd-ac05-0362914c9a4e», «LaboratoryTests»: {«02-002»: 108183, «03-003»: 106007, «06-003»: 106019}}
```

Копируем и отсылаем «taskID»: «5ef19a56-a1c4-44cd-ac05-0362914c9a4e».

Если cron работает и настроен согласно инструкции, в карте можно будет увидеть результат тестового анализа. Так же можно форсировать импорт в консоли:

```
curl «http://localhost/_services/Laboratory/Helix/Invoice.getAndStoreResult?no_connection=1»
```



From:

<http://mgdemo.ru:5555/> - **МИС Mgerm**

Permanent link:

<http://mgdemo.ru:5555/doku.php?id=install:helix&rev=1730545451>

Last update: **02-11-2024 11:04**

