

Установка адаптера

Установка адаптера для обмена данными с интеграционной шиной требуется если планируется использовать ЛК в качестве ЛК обучающегося и/или преподавателя (ЛКО/ЛКП). Для использования ЛК в качестве ЛК абитуриента (ЛКА) установка адаптера не требуется.

Содержание

1. [Требования к серверу для адаптера](#)
2. [Установка приложения при помощи скрипта](#)
3. [Установка приложения вручную](#)
4. [Настройка адаптера](#)

Требования к серверу

Адаптер для обмена данными с НСИ работает в качестве отдельного микросервиса, подключенного к базе данных Личного кабинета.

Для установки приложения должны выполняться следующие требования:

- Установленное приложение Личного кабинета
- PHP = 7.4. На более ранних версиях PHP приложение работать не будет.
 - Установленное расширение JSON
 - Установленное расширение SOAP
 - Установленное расширение Mbstring
 - Установленное расширение OpenSSL
 - Установленное расширение PDO
 - php-fpm
- MySQL 8 или MariaDB >= 10.4.
- Nginx (1.18 или старше)

- Composer >=2.0
- xml-twig-tools (xml_split)
- Git >= 2.25

Установка приложения при помощи скрипта

1. Предварительно должна быть выполнена установка личного кабинета через [скрипт](#).
2. Запросить доступ к репозиторию lk/nsi-adapter по ключу. Ключ сообщит скрипт. Доступ нужно запросить через руководителя соответствующего проекта компании ТАНДЕМ (РП).
3. Выполнить пункты, предлагаемые скриптом (*пункт меню 3 подпункт 3 и пункт меню 4 подпункт 3*).
4. Настроить адаптер через .env-файл (раздел Настройка / Настройка окружения).
5. Проверить [корректность установки](#).

Установка приложения вручную

Запросить доступ по ключу к следующему репозиторию:

```
gitlab.tandemservice.ru:lk/nsi-adapter
```

Проверить доступ к репозиторию:

```
ssh -T git@gitlab.tandemservice.ru
```

Получить последнюю версию приложения:

```
git clone git@gitlab.tandemservice.ru:lk/nsi-adapter.git
```

Перейти в папку приложения и выполнить команду:

```
composer install --no-dev
```

Настройка

Настройка окружения

Перейдите в директорию приложения и откройте файл `.env`. Если его нет, то создайте копию файла `.env.example` и переименуйте его в `.env`.

1. Создайте криптографический ключ:

```
php artisan key:generate
```

2. Откройте файл `.env`.

3. Настройте:

3.1 Продуктовый режим работы приложения:

```
APP_ENV=production  
APP_DEBUG=FALSE
```

3.2 Часовой пояс:

```
APP_TIMEZONE="Asia/Yekaterinburg"
```

4. Настройте параметры соединения с БД:

Необходимо указать реквизиты базы данных, которую использует приложение личных кабинетов ([Настройка окружения](#)).

```
```\nDB_CONNECTION=mysql\nDB_HOST=127.0.0.1\nDB_PORT=3306\nDB_DATABASE=dbname\nDB_USERNAME=dbuser\nDB_PASSWORD=password\n```\n
```

5. Настройте параметры аутентификации, указав логин и пароль от настраиваемой подсистемы в шине данных:

```
NSI_SUBSYSTEM_LOGIN=login
NSI_SUBSYSTEM_PASSWORD=password
```

Итоговый файл может иметь следующий вид:

```
APP_NAME="Адаптер HCI"
APP_ENV=production
APP_KEY=base64:Z234567890900NEb1Tm0zWC5673nM5zLHb6h30o4n90c=
APP_DEBUG=FALSE
APP_TIMEZONE="Asia/Yekaterinburg"
```

```
LOG_CHANNEL=stack
```

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=dbname
DB_USERNAME=dbuser
DB_PASSWORD=password
```

```
NSI_SUBSYSTEM_LOGIN=login
NSI_SUBSYSTEM_PASSWORD=password
```

```
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
```

## Настройка директории `public`

Для работы приложения необходимо настроить корневую папку, из которой будет запускаться приложение. В директории приложения находится папка `public`, а внутри папки находится файл `index.php`. Необходимо настроить виртуальный хост веб-сервера таким образом, чтобы он указывал на папку `public`.

Пример конфигурации при развёртывании `nginx`, `php-fpm` из стандартных пакетов Debian 11:

```
server {
 listen 8089;
 server_name _;
 set $BACK_APP_ROOT /путь до папки/public;
 root $BACK_APP_ROOT;
```

```

gzip on;
gzip_disable "msie6";
gzip_vary on;
gzip_proxied any;
gzip_comp_level 5;
gzip_http_version 1.1;
gzip_min_length 1100;
gzip_types text/plain text/html application/json text/xml
application/xml

location ~ /\.php$ {
 fastcgi_split_path_info ^(.+\.(php|php5))(/.+)$;
 fastcgi_pass unix:/run/php/php-fpm.sock;
 fastcgi_param SCRIPT_FILENAME
$realpath_root$fastcgi_script_name;
 fastcgi_index index.php;
 include fastcgi_params;
}

location / {
 root $BACK_APP_ROOT;
 try_files $uri $uri/ /index.php?$query_string;
 index index.php;
}
}

```

## Права доступа к директориям

Все вложенные папки и файлы в директориях storage и bootstrap/cache должны быть доступны для чтения и записи веб-сервером.

В ОС Linux:

Предполагается, что /var/www/html/laravel/root/directory – путь до директории приложения, имя вашего пользователя tandem, а имя пользователя для веб-сервера www-data. Замените данные параметры на свои и выполните команды в терминале:

```

sudo chown -R tandem:www-data
/var/www/html/laravel/root/directory

```

```
sudo usermod -a -G www-data tandem
```

И затем настройте права доступа:

```
sudo find /var/www/html/laravel/root/directory -type f -exec
chmod 664 {} \;
sudo find /var/www/html/laravel/root/directory -type d -exec
chmod 775 {} \;
```

## Проверка корректности установки

После установки нужно проверить корректность работы адаптера.

По адресу `http://ip.адрес.сервера:8089/services/NSIService?wsdl` сервер должен вернуть xml с описанием веб-сервиса:

