

Web Push API

Данный модуль позволяет отправлять Push уведомления клиентам и администраторам через Web Push платформу.

Клиент реализован на ServiceWorker, поэтому будет работать только на домене с валидным SSL (HTTPS) сертификатом. Для отправки в Chrome нужен ключ Google API Platform Key. (Брать в [Google Cloud Console](#)). Ключ нужно записать в `libexec/config.pl $conf{GOOGLE_API_KEY}`.

Пуши в Firefox работают без ключа.

После записи ключа создать файл манифеста `cgi-bin/manifest.json`.

```
cd /usr/abills/cgi-bin/  
cp ../misc/gcm/manifest.json.default manifest.json
```

Заменить `%GCM_APPLICATION_NUMBER%` на номер своего проекта в [Google Cloud Console](#).

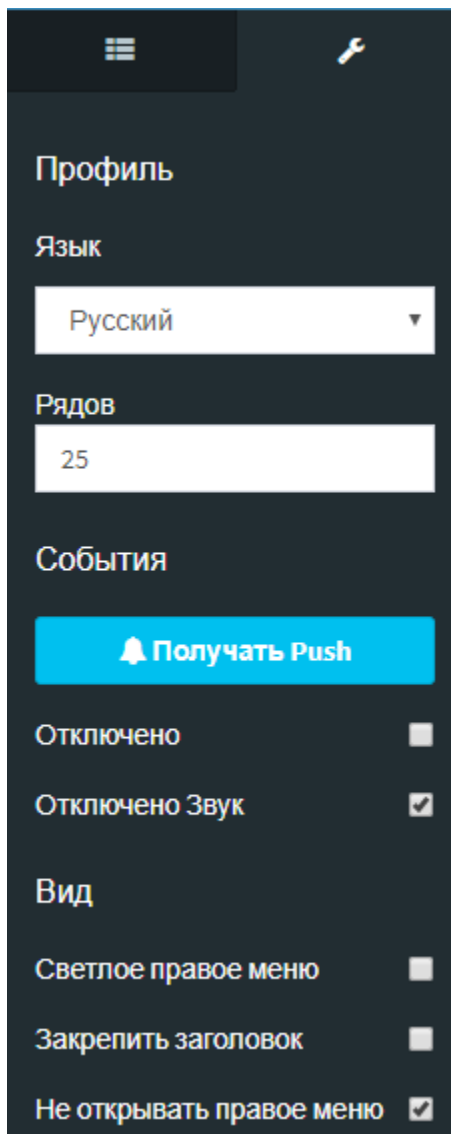
```
vim manifest.json
```

Создать таблицы, в которых хранятся клиенты и сообщения.

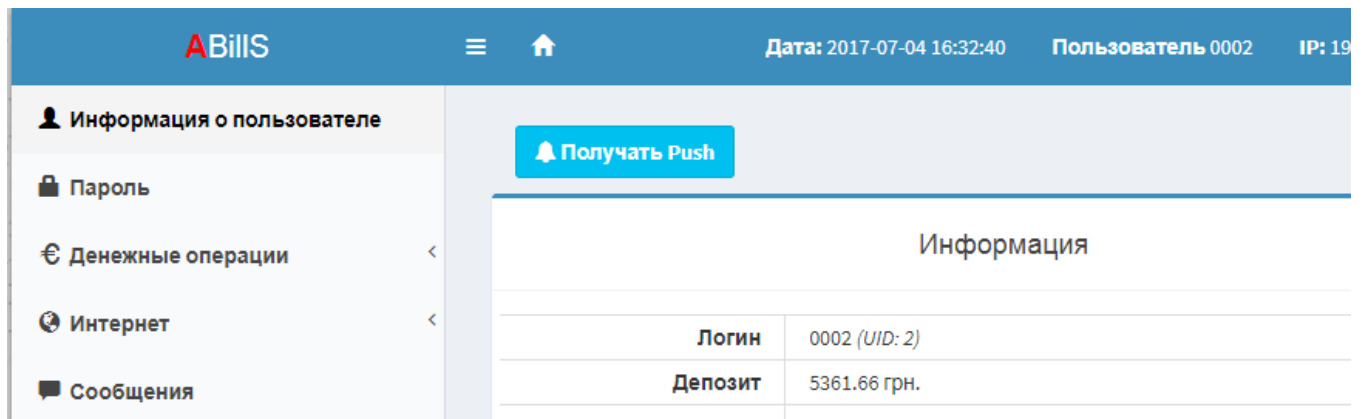
```
mysql -D abills < db/Push.sql
```

После этого можно включить скрипты Push переменной конфигурации `$conf{PUSH_ENABLED}`.

Администраторы подписываются на Push уведомления на странице «Профиль».



Пользователям предложит подписаться сразу при входе в кабинет.



Для разработчиков

Пример использования:

```

#
use Abills::Sender::Core;
my $Sender = Abills::Sender::Core->new($db, $admin, \%conf, {
    SENDER_TYPE => 'Push'
});

#
my $request_sent = $Sender->send_message({
    UID      => 2,
    TITLE    => 'ABillS Push',
    MESSAGE  => 'Hello, world sent via Push message'
});

# $request_sent == 1,      Push . ( )

```

Детали реализации

Сама схема работы WebPush хорошо расписана в [документации для разработчиков Google](#).

В данном случае есть ServiceWorker в браузере, который отвечает за подписку и уведомление сервера о своей подписке.

Сервер, получив данные о подписке записывает контакты в таблицу **push_contacts** (модуль `Contacts`) и отдаёт клиенту **contact_id**, который сохраняется внутри **indexedDB** в браузере.

Когда нужно отправить уведомление, модуль **Push** получает контакт из таблицы и отправляет запрос на **Push-уведомление** сервису **Push**. Если запрос завершился удачно, сообщение сохраняется в таблицу **push_messages** с некоторым TTL (24 часа).

Когда клиент получает **Push-уведомление** от Push-сервиса, он используя сохранённый **contact_id**, получает с сервера ABillS сообщение для этого **contact_id**. После отправки сообщения клиенту, оно удаляется из таблицы.

Дневной периодик функцией **msgs_clear_push_messages**, удаляет сообщения, которые так и не были доставлены. Для модуля `Notepad` есть [billd плагин](#), для рассылки напоминаний через Push.