**ЛАБОРАТОРНАЯ РАБОТА №9**

**АВТОРИЗАЦИЯ ПО ПРОТОКОЛУ OAUTH**

**(SPRING BOOT)**

Цель работы: получение навыков разработки системы авторизации по протоколу OAuth.

Программное обеспечение: IntelliJ IDEA, JDK 1.8+, Spring Boot, Hibernate, MySQL.

Необходимая теоретическая подготовка: Maven, Java Spring, Spring Boot, OAuth.

ПРАКТИЧЕСКАЯ ЧАСТЬ

Яндекс

На странице входа в систему нашего сайта пользователи могут выбрать один из двух вариантов входа. Первый — указать имя пользователя и пароль, если у них уже есть учетная запись на нашем сайте. А другой заключается в том, чтобы войти на сайт с помощью существующей учетной записи Яндекс.

Когда они нажимают кнопку «Войти через Яндекс», запускается поток авторизации через Яндекс и он переводит пользователей на сайт Яндекс. Оказавшись там, они входят в систему со своими учетными данными Яндекс, после чего они будут перенаправлены на страницу подтверждения.

На странице подтверждения у пользователей будет запрошено разрешение на передачу информации о своей учетной записи Яндекс на сторонний сайт. В этом случае сторонний сайт — это сайт, на котором они хотят использовать свою учетную запись Яндекс для входа в систему. Им будут представлены два варианта: они могут разрешить или запретить это.

После того, как они разрешат поделиться своей информацией со сторонним сайтом, пользователи будут перенаправлены на сторонний сайт, откуда они инициировали поток входа в Яндекс.

На этом этапе пользователь вошел в систему с помощью Яндекс, а сторонний сайт имеет доступ к информации профиля пользователя, которую можно использовать для создания учетной записи и входа в систему.

Это основной процесс интеграции авторизации Яндекс на нашем сайте.

**Настройка проекта для авторизации через Яндекс**

Переходим на странницу <https://oauth.yandex.ru/>

Нажимаем кнопку "Регистрация клиентских приложений".

Вводим название (обязателен) и описание приложения, ссылку на иконку и ссылку на сайт приложения (Рис.1)

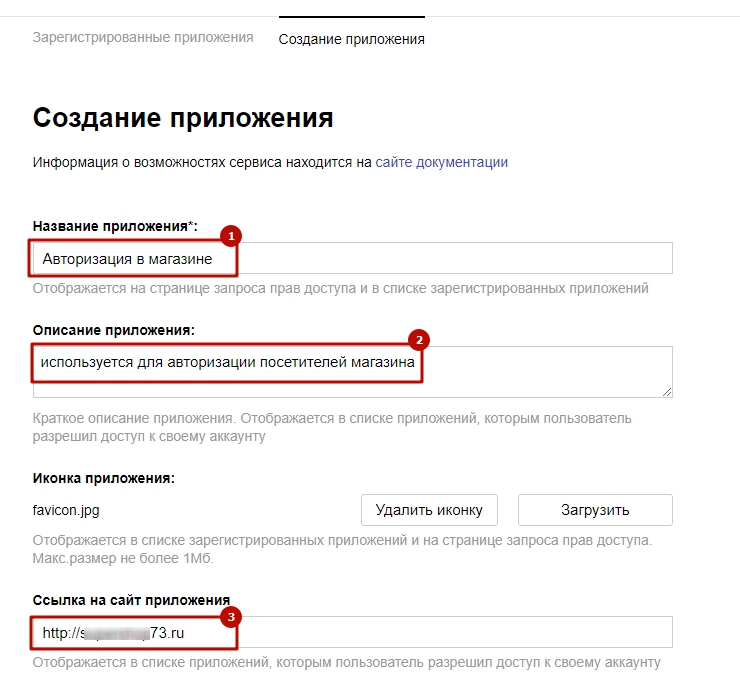
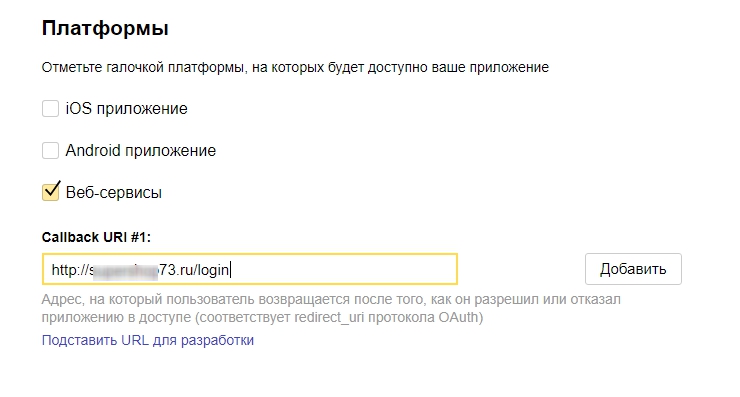


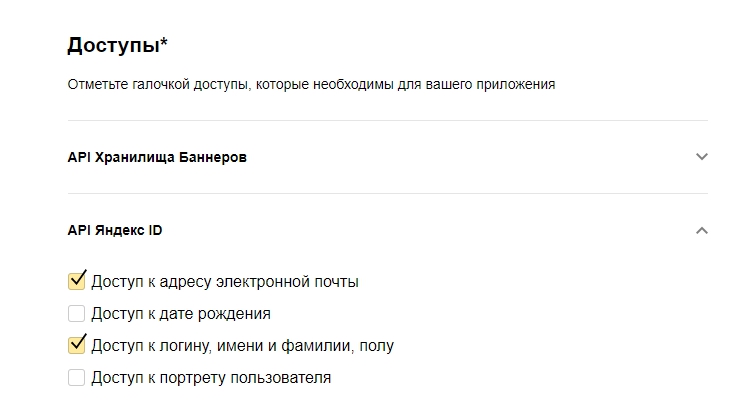
Рисунок 1.

Далее в разделе "Платформы" ставим галочку около "Веб-сервисы", в открывшемся поле вводим ссылку на страницу авторизации магазина [адрес сайта]/Login (Рис.2)

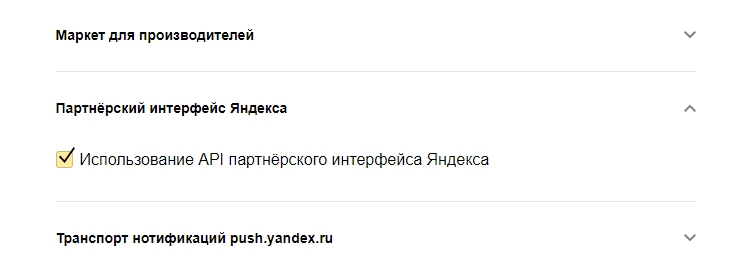
  
Рисунок 2.

Устанавливаем права приложения:

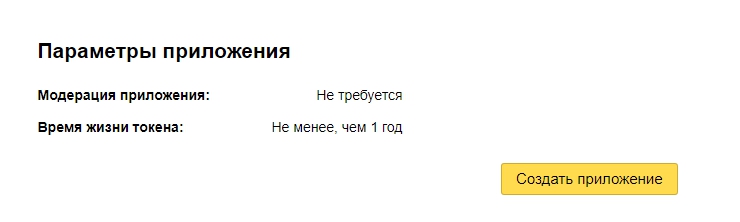
На вкладке API **Яндекс.ID** устанавливаем галочку на значениях "Доступ к логину, имени и фамилии, полу" и "Доступ к адресу электронной почты" (Рис. 3)

  
Рисунок 3.

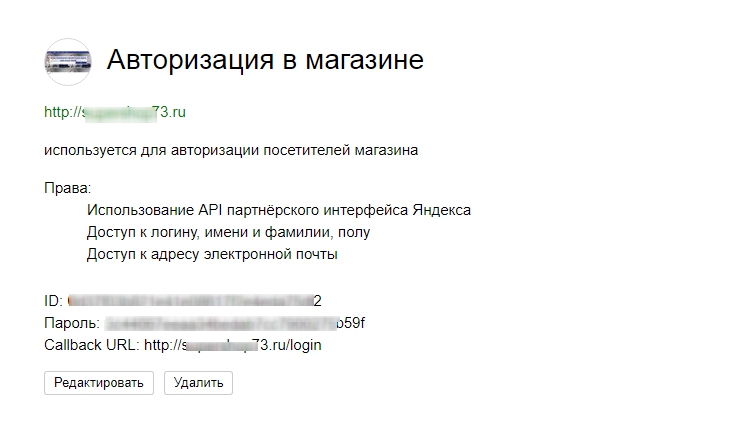
Устанавливаем галочку "**Использование API партнёрского интерфейса Яндекса**"(Рис.4)

  
Рисунок 4.

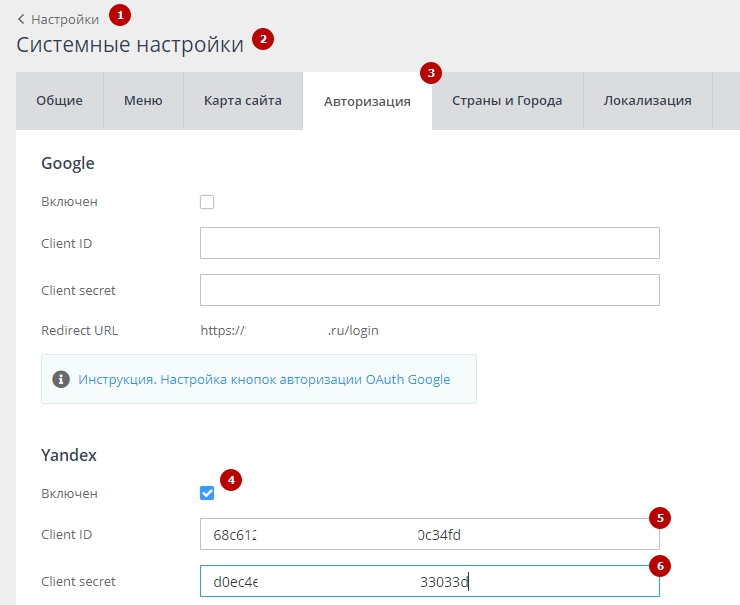
Спускаемся вниз страницы и нажимаем "Создать приложение" (Рис.5)

  
Рисунок 5.

После сохранение настроек открывается следующее окно (Рис.6):

  
Рисунок 6.

Копируем значения параметров ID в **Client ID** и Пароль в **Client secret**, сохраняем в настройках вашего интернет магазина ("**Настройки**"– "**Системные настройки**", вкладка "**Авторизация**") (Рис.7)

  
Рисунок 7.

На данный момент мы создали клиентское приложение Яндекс OAuth2, и теперь нам нужно получить возможность использовать это приложение для интеграции на нашем сайте входа в систему через Яндекс. Пожалуйста, запишите идентификатор клиента и секретный ключ клиента, они потребуются при настройке приложения с нашей стороны. Вы всегда можете найти идентификатор клиента и секретный ключ клиента при редактировании приложения.

Установите клиентскую библиотеку Яндекс PHP SDK

Есть два варианта установки. Использовать Composer.

Скачать релиз

Если вы не хотите использовать Composer, вы также можете загрузить последнюю стабильную версию с [официальной страницы API](https://github.com/googleapis/google-api-php-client/releases).

К настоящему момент у вас должно быть настроено приложение Яндекс и установлена клиентская библиотека API Яндекс PHP.

Интеграция клиентской библиотеки

Библиотека устанавливается с помощью пакетного менеджера [Composer](https://getcomposer.org/).

Добавьте библиотеку в файл composer.json вашего проекта:

{

"require":

{

"yandex-market/yandex-market-php-partner": "\*"

}

}

Включите автозагрузчик Composer в код проекта:

require \_\_DIR\_\_ . '/vendor/autoload.php';

Давайте пройдемся по ключевым частям кода. Первое, что нам нужно сделать, это включить файл autoload.php. Это часть Composer, она гарантирует, что классы, которые мы используем в скрипте, загружаются автоматически.

Далее у нас есть раздел конфигурации, который инициализирует конфигурацию приложения путем установки необходимых настроек. Конечно, вам необходимо заменить заполнители на соответствующие значения.

|  |  |
| --- | --- |
| 1  2  3  4 | // init configuration  $clientID = '<YOUR\_CLIENT\_ID>';  $clientSecret = '<YOUR\_CLIENT\_SECRET>';  $redirectUri = '<REDIRECT\_URI>'; |

Затем мы добавили электронную почту и данные профиля, таким образом после входа в систему у нас есть доступ к основной информации профиля.

|  |  |
| --- | --- |
| 1  2 | $client->addScope("email");  $client->addScope("profile"); |

После нажатия на ссылку для авторизации через Яндекс пользователи будут перенаправлены на сайт Яндекс. После входа в систему Яндекс перенаправляет пользователей на наш сайт, передавая переменную строки запроса code. И вот тогда будет запущен код PHP в блоке if. Мы будем использовать код для обмена токеном доступа.

Получив токен доступа, мы можем использовать сервис Яндекс\_Service\_OАuth2 для получения информации о профиле вошедшего в систему пользователя.

Таким образом, мы получим доступ к информации профиля, как только пользователь войдет в учетную запись Яндекс. Мы можем использовать эту информацию для создания учетных записей на своем сайте или сохранить ее в сеансе.