ООО «КВАНТУМ АРТ»

115184, Москва, Озерковский переулок, д. 12

тел. (495) 783-65-74

Программный продукт «QP8.CMS с поддержкой PostgreSQL»

Руководство администратора

Москва
2025

Назначение документа

Настоящий документ содержит руководство администратора по программному продукту «QP8.CMS с поддержкой PostgreSQL». Цель документа – предоставить администратору сведения о продукте, достаточные для установки и эксплуатации.

Целевая аудитория

Документ предназначен для администраторов, обладающих следующими компетенциями:

- администрирование операционных систем семейств Microsoft Windows и Linux,
- администрирование СУБД Microsoft SQL Server или PostgreSQL,
- администрирование систем для управления данными,
- знание веб-технологий (HTTP, DNS).

История изменений

Версия	Дата	Автор	Описание
0.9.8	21.04.2025	Молькова М.Е.	Добавлен раздел «Интеграция с KeyCloak»
0.9.7	20.03.2025	Молькова М.Е.	Добавлен раздел «Интеграция с Camunda»
0.9.6	06.06.2024	Селю П.Н.	Добавлен раздел «Интеграции с внешними системами Интеграция с MinIO»
0.9.5	21.03.2024	Молькова М.Е.	Добавлен раздел «Серверная валидация HTML-тегов»
0.9.4	20.10.2023	Молькова М.Е.	Добавлены разделы: • Конфигурационный файл веб-сайта • Интеграция с AD
0.9.3	06.06.2023	Селю П.Н.	• Дополнена установка под Linux: – создание /qplibrary, – необходимые права доступа после распаковки, – создание симлинков, – SQL-скрипт настройки QP.Storage (дополнение), – замечания по установке под Astra и RedOS, – измененный порядок выполнения шагов при разворачивании базы (прогон current.sql), – исправленный таргет для NLog.config.
0.9.2	14.04.2023	Селю П.Н.	• Установка под Linux (без использования Docker) • Установка под Linux (с использованием Kubernetes) • Обновление JWT и использование nginx • Добавлены ссылки на дистрибутив • Обновлена версия SDK и Runtime
0.9.1	07.04.2023	Селю П.Н.	• Версия документа только для QP8.CMS с поддержкой PostgreSQL • Установка под Linux (с использованием Docker)

0.9.0	02.12.2020	Селю П.Н.	Добавлено: • Разделение на продукты QP8.CMS и QP8.CMS с поддержкой PostgreSQL • Установка БД
0.8.4	02.07.2020	Мальцева Ю.А.	Добавлено: • Требование изменить пароль при следующем входе и требования к паролю (см. Параметры входа (Login Parameters)) Обновлено: • Требования к операционной системе (см. Операционная система). Обновленное требование - серверная ОС Microsoft Windows Server 2012 или выше • Требования к СУБД (см. СУБД). Использование Microsoft SQL Server 2012 или выше • Требования к веб-серверу (см. Веб-сервер) Использование Microsoft IIS 8.5 или выше • Требования к используемому ПО для разработчика (см. ПО для Разработчика). Использование Microsoft Visual Studio 2017 или выше
0.8.3	20.02.2018	Советкали Б.С.	Обновлено: • Требования к используемому ПО для разработчика (см. ПО для Разработчика). Обновленное требование - использование .NET Framework не ниже 4.7.1 • Требование к используемому веб-браузеру Internet Explorer не ниже 11.0 (см. Программное обеспечение для работы с ГПИ продукта) Добавлено: • Ссылка на установку программного продукта QP (см. Начало установки)

Оглавление

1 ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 6

1.1 Общие термины и определения 6
1.2 Термины и определения для QP 7
1.3 Определения для ролей пользователей в Системе 7
1.4 Аббревиатуры 7

2 ОБОЗНАЧЕНИЯ 9

3 ОБЩИЕ СВЕДЕНИЯ О ПРОДУКТЕ 10

3.1 Получение дистрибутива продукта 10
3.2 Системные требования 10
3.2.1 Аппаратное обеспечение 10
3.2.2 Программное обеспечение 10
3.3 Документация по продукту 11

4 УСТАНОВКА ПРОДУКТА 13

4.1 Установка продукта на Windows 13
4.2 Установка продукта на Linux 15
4.2.1 Установка продукта на Linux (с использованием Docker) 15
4.2.2 Установка продукта на Linux (без использования Docker) 15
4.2.3 Установка продукта на Linux (с использованием Kubernetes) 20
4.2.4 Изменение JWT-токена 20
4.2.5 Настройка nginx 20

5 УСТАНОВКА БД 22

5.1 Установка БД на Linux 22
5.2 Установка БД на Windows 24
5.2.1 Минимальная конфигурация 24
5.2.2 Полная конфигурация 25

6 ОБНОВЛЕНИЕ ПРОДУКТА 26

6.1 Обновление продукта на Windows 26
6.2 Обновление продукта на Linux 26
6.2.1 Обновление продукта на Linux (с использованием Docker) 26
6.2.2 Обновление продукта на Linux (без использования Docker) 26
6.2.3 Обновление продукта на Linux (с использованием Kubernetes) 26

7 НАСТРОЙКА ПРОДУКТА 27

7.1 Конфигурационный файл QP 27
7.1.1 Конфигурационный файл QP на Windows 27
7.1.2 Конфигурационный файл QP на Linux 27
7.1.3 Настройки конфигурации продукта 27
7.1.4 Customer code 28
7.2 Конфигурационный файл веб-сайта 29
7.2.1 Серверная валидация HTML-тегов 35
7.3 Конфигурационный файл QP.Storage 36
7.4 Настройки аутентификации 38
7.4.1 Настройки для Windows 38
7.4.2 Интеграция с внешними системами 42
7.5 Страница «Настройки» корневого элемента дерева 43
7.6 Запись и воспроизведение действий 47
7.6.1 Воспроизведение через ГПИ бэкенда 47
7.6.2 Программа QP8.ConsoleDbUpdate 48

7.6.3 Использование QP8 Backend API *51*
7.6.4 Лог операций воспроизведения действий *52*
7.6.5 Лог операций запуска воспроизведения *53*
7.7 Аудит 53
7.7.1 Представление «Журнал действий» *53*
7.7.2 Представление «Пользовательские сессии» *54*
7.7.3 Представление «Неудачные попытки входа» *54*
7.8 Управление учётными записями пользователей 54
7.8.1 Пользователь *54*
7.8.2 Группа пользователей *56*
7.8.3 Связывание пользователя и группы пользователей *57*
7.8.4 Рекомендуемая последовательность при работе с пользователями *57*
7.9 Импорт пользователей из AD 58
7.9.1 Настройка синхронизации *58*
7.9.2 Правила синхронизации *58*
7.10 Настройка прав доступа 59
7.10.1 Уровни доступа *59*
7.10.2 Настройка доступа к сущностям *59*
7.10.3 Настройка доступа к действиям *64*
7.10.4 Права доступа на связи *68*
7.10.5 Права доступа по типу продукта *69*

8 ЗАДАЧИ ПО РАСПИСАНИЮ (SCHEDULED TASKS) 71

9 ИНТЕГРАЦИИ С ВНЕШНИМИ СИСТЕМАМИ 74

9.1 Интеграция с MinIO 74
9.1.1 Миграция *74*
9.1.2 Настройка QP8 на использование MinIO *75*
9.2 Интеграция с Camunda 76
9.2.1 Настройки в ГПИ QP *77*

10 УДАЛЕНИЕ ПРОДУКТА 85

10.1 Удаление продукта на Windows 85
10.2 Удаление продукта на Linux 85
10.2.1 Удаление продукта на Linux (с использованием Docker) *85*
10.2.2 Удаление продукта на Linux (без использования Docker) *85*
10.2.3 Удаление продукта на Linux (с использованием Kubernetes) *85*

11 ПРОЧЕЕ 86

11.1 Плагин «Яндекс.Спеллер» для визуального редактора 86

12 ПАРТНЁРСТВО И СЕРТИФИКАЦИЯ 87

1 Термины и определения

1.1 Общие термины и определения

Термин или определение	Описание
Active Directory (далее **«AD»**)	Служба каталогов для операционных систем Microsoft Windows Server. Базируется на протоколе LDAP
Camunda	Открытая платформа для автоматизации бизнес-процессов и управления задачами
Development-окружение	Среда, в которой осуществляется разработка и отладка Систем
Entity Framework (далее **«EF»**)	Технология для доступа к данным с использованием объектно-реляционного сопоставления (ORM, от англ. «Object-Relational Mapping»)
KeyCloak	Система идентификации и управления доступом с открытым исходным кодом
MinIO	Открытое программное обеспечение, позволяющее создавать собственное S3-совместимое хранилище на своем сервере или в облаке.
NuGet	Средство для управления пакетами, используемое при разработке программных продуктов на платформе Microsoft.
Production-окружение	Среда, используемая для размещения Систем, готовых к эксплуатации неограниченным кругом пользователей
QP8.CMS или **QP8.CMS с поддержкой PostgreSQL** (далее **«QP»**)	Программный продукт, обладающий широким спектром возможностей для разработки программной части Систем различной сложности
QP8.WidgetPlatform (также **«Виджетная платформа»**)	Продукт, расширяющий возможности QP. Позволяет через бэкенд наполнять веб-страницы Системы самостоятельно разработанными Модульными приложениями. Виджетная платформа и виджеты основаны на шаблоне архитектуры MVC (от англ. «Model-View-Controller», «Модель-Представление-Контроллер»).
Stage-окружение	Среда, максимально приближенная к production-окружению, в которой персоналом организации-разработчика осуществляется тестирование Систем
Графический пользовательский интерфейс (далее **«ГПИ»**)	Метод взаимодействия пользователя с Системой, при котором все ключевые способы управления Системой выполнены с использованием различных графических элементов
Инструмент	Часть Системы, обладающая определёнными функциональными возможностями
Информационная Система (далее **«Система»**)	Автоматизированный программно-аппаратный комплекс, предназначенный для хранения, обработки и выдачи данных
Модульное приложение (также **«Приложение»**, **«Виджет»**)	Обладающий ГПИ инструмент, содержащий набор функциональных возможностей для взаимодействия пользователей с какой-либо Системой (текущей или сторонней)
Обработчик	Программное средство, используемое на серверной части Системы для обработки запросов пользователей к веб-сайту Системы
Пагинация	Нумерация страниц

1.2 Термины и определения для QP

Термин или определение	Описание
DNS	Доменное имя, используемое в Системе для работы с веб-сайтом
Бэкенд	Копия QP. Бэкенд обладает ГПИ для работы с содержимым БД Системы
Виртуальный путь	URI до объекта
Код клиента (Customer code)	Уникальный параметр, определяющий БД Системы, с которой взаимодействует бэкенд QP
Контент	Раздел сайта
Поле	Атрибут контента. С использованием полей формируется структура данных для контента
Пользовательское действие	Дополнительная функциональная возможность для бэкенда, добавленная Разработчиком в Систему
Реплейс	Уникальное кодовое имя для статьи, с использованием которого можно вызвать содержимое этой статьи в других статьях текущего сайта
Сайт	Набор данных в бэкенде. Допускается создание нескольких сайтов. Содержимое каждого сайта определяется созданными в нём контентами
Статья	Элемент контента. Статья содержит данные, заданные в поля контента
Физический путь	Путь до объекта в файловой системе

1.3 Определения для ролей пользователей в Системе

Роль	Определение
Администратор	Пользователь с правами на внесение любых изменений в Систему, которые можно выполнить с использованием бэкенда QP либо отдельной административной панели управления
Контент-менеджер	Пользователь с ограниченными правами на изменение содержимого Системы с использованием бэкенда QP либо отдельной административной панели управления
Пользователь	Персона, осуществляющая взаимодействие с Системой посредством интерфейсов, предоставляемых Системой
Разработчик	Пользователь с правами на внесение любых изменений в Систему (в том числе в содержимое скриптов, структуру БД)

1.4 Аббревиатуры

Аббревиатура	Значение
AD (Active Directory)	Служба каталогов для операционных систем Microsoft Windows Server. Базируется на протоколе LDAP

API (Application programming interface, интерфейс программирования приложений)	Набор правил по использованию функциональных возможностей Системы, предоставляемый разработчикам для организации взаимодействия сторонних программных продуктов с Системой
CLI (Command Line Interface)	Интерфейс командной строки, с помощью которого пользователь взаимодействует с ПО
DPC (Digital Product Catalog)	Программный продукт, разработанный на базе QP; ориентирован на удобство работы со структурой данных для различных продуктов Заказчика.
HTML (HyperText Markup Language)	Язык разметки документов
ID (Identifier)	Идентификатор объекта
JSON (JavaScript Object Notation)	Текстовый формат обмена данными
LINQ (Language-Integrated Query)	Компонент .NET Framework для работы с данными из БД как с объектами. Запросы к СУБД формируются с использованием языков программирования .NET
LINQ to SQL	Решение для LINQ, позволяющее в качестве источника данных использовать СУБД Microsoft SQL Server
PNG (Portable Network Graphics)	Растровый формат хранения для графических данных
POCO (Plain old CLR object)	Подход к разработке программного обеспечения, предполагающий использование максимально простых классов для работы с объектами
SSO (Single Sign-On)	Метод аутентификации, который позволяет пользователям безопасно аутентифицироваться в несколько связанных сервисов, используя один набор учетных данных
UI (User Interface)	Пользовательский интерфейс
XML (eXtensible Markup Language)	Расширяемый язык разметки документов
БД	База данных
СУБД	Система управления базами данных

2 Обозначения

Обозначение	Описание	Пример использования
`Технические данные`	Используется для выделения различных технических данных в тексте: URL, названия свойств и методов, имена файлов и т.п.	ГПИ Системы доступен по URL `https://www.domainname.zone/`.
`Код`	Пример кода.	`public DataTable Data { get; set; }`
Переменная	Используется для указания переменного значения.	Формат URL: *Базовый URI/Псевдоним объекта*
Требует дополнения	TBD (to be determined). Указывает, что необходима доработка текста – проверка корректности утверждения, детализация, правка после внесения изменений в документ и т.п.	Система работает с одной БД.
Примечание:	Дополнительные данные справочного характера.	**Примечание:** используется при генерации классов LINQ to SQL.
Внимание:	Важные данные, которые требуется обязательно учитывать.	**Внимание:** опция поддерживается только ASP-сборкой в целях совместимости.

3 Общие сведения о продукте

3.1 Получение дистрибутива продукта

Продукт «QP8.CMS с поддержкой PostgreSQL» доступен по адресу: https://downloads.quantumart.ru/qp8_pg.

3.2 Системные требования

3.2.1 Аппаратное обеспечение

3.2.1.1	**Минимальная конфигурация**	**Рекомендуемая конфигурация**
Процессор	Intel Pentium IV 1.8 ГГц	Intel Xeon 2.4 ГГц x2
Память	2 ГБ	8 ГБ
Дисковое пространство	2 ГБ	100 ГБ и больше (в зависимости от применения)

3.2.2 Программное обеспечение

3.2.2.1 Операционная система

Окружение	**Описание**
Production	Требуется серверная ОС: • Microsoft Windows Server 2012 R2 или выше • Linux (в.т.ч. Ubuntu, Debian, CentOS, RedOS, AstraLinux)
Stage	Достаточно ОС для настольных ПК: • Microsoft Windows 10 ver. 1809 или выше Рекомендуется ОС из списка для production-окружения.
Development	

Поддерживаются 64-битные версии ОС.

Для рабочего места Разработчика достаточно любой из перечисленных ОС для настольных ПК.

3.2.2.2 СУБД

Поддерживается любая из списка:

- Microsoft SQL Server 2012 или выше
- PostgreSQL (или Postgres Pro) 12 или выше с установленным contrib

3.2.2.3 ПО для установки на Linux (с использованием Docker)

- docker
- docker-compose

3.2.2.4 ПО для установки на Linux (без использования Docker)

- Для работы приложения:
  - aspnetcore-runtime-6.0 (см инструкцию)
  - nginx
- Для сборки из исходников:
  - .NET Core SDK 6.0.408 и выше
  - NodeJs 16

3.2.2.5 ПО для установки на Linux (Kubernetes)

- Кластер Kubernetes с версией компонентов 1.20 или выше
- kubectl 1.20 или выше

3.2.2.6 ПО для установки на Windows

- IIS 8.5, при этом обязательна установка следующих компонентов:
  - Web Management Tools, все компоненты, кроме IIS6-специфичных
  - Security – все компоненты
  - Common HTTP Features – все, кроме WebDAV Publishing
  - Application Development Features – все, компоненты кроме ASP и связанных с .NET 3.5 и .NET 4.8
- Powershell 5.1 или выше с модулем WebAdministration (для IIS)
- Для установки на SQL Server:
  - PS-модуль SqlServer или SqlPs
- Для установки на PostgreSQL:
  - Postgres CLI (psql, pg_restore)
- ASP.NET Core Runtime 6.0.16 (Hosting Bundle)

Примечание: Для работы Postgres CLI Может потребоваться дополнительная установка компонента (https://support.microsoft.com/ru-ru/help/2977003/the-latest-supported-visual-c-downloads), если он не был установлен в системе ранее.

3.2.2.7 Программное обеспечение для работы с ГПИ продукта

Работа с ГПИ ведётся с использованием веб-браузера. Поддерживаемые веб-браузеры:

- Google Chrome (или веб-браузер на основе Chromium),
- Microsoft Internet Explorer (не ниже 11.0),
- Microsoft Edge,
- Mozilla Firefox.

Примечание: рекомендуется использовать актуальную версию веб-браузера.

3.2.2.8 ПО для разработчика

- Необходимые SDK и фреймворки
  - .NET Core SDK 6.0.408 и выше
  - NodeJs 16
- Поддерживаемые IDE
  - Microsoft Visual Studio 2022 (v.17.4 или выше)
  - Rider 2022.1 или выше

3.3 Документация по продукту

Пакет документов по продукту содержит:

Название	Описание
Руководство администратора	Установка, обновление, удаление продукта и эксплуатация Системы на основе продукта
Руководство разработчика	Разработка Системы на основе продукта

Руководство редактора	Работа с данными в Системе, созданной на основе продукта, в роли Контент-менеджера

4 Установка продукта

4.1 Установка продукта на Windows

Внимание: установка продукта возможна только от имени пользователя, обладающего правами администратора.

Внимание: продукт в данном варианте устанавливается без БД, в предположении, что в дальнейшем вероятно будут устанавливаться продукты QP8.ProductCatalog или QP8.WidgetPlatform со своими БД. Если это не планируется, можно отдельно установить БД.

Продукт доступен по ссылке: http://downloads.quantumart.ru/QP8_PG/QP8.zip

1. Скачать и распаковать дистрибутив (перед распаковкой необходимо разблокировать архив, чтобы в дальнейшем извлекаемые из него файлы были также разблокированы):

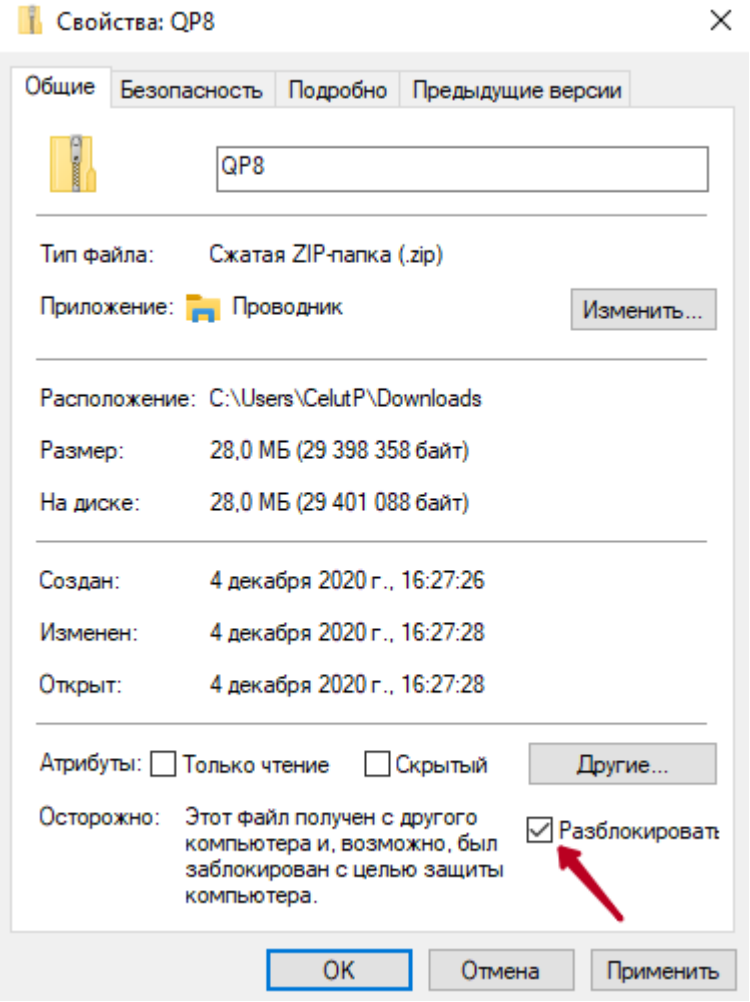


Рисунок 4.1. Разблокировка архива QP8.

2. Запустить InstallQP8.cmd (минимальный режим – все параметры по умолчанию) или InstallQP8Interactive.cmd (интерактивный режим – можно задать все параметры).
3. Если выбран интерактивный режим, то можно задать следующие параметры:
   a. Имя сайта в IIS (по умолчанию – QP8)
   b. Порт сайта (по умолчанию – 89)
   c. Путь к существующему конфигурационному файлу (если мы хотим использовать конфигурацию от предыдущих инсталляций)

   d. URL внешнего конфигурационного сервиса (если мы хотим использовать уже развернутый на момент инсталляции сервис; запрашивается, если не задать параметр из пункта c)
   e. Папка для нового конфигурационного файла (по умолчанию – C:\QA; запрашивается, если не задать параметр из пункта d)
   f. Токен внешнего конфигурационного сервиса (запрашивается, если задать параметр из пункта d)
   g. Папка для временных файлов (по умолчанию – C:\Temp)
   h. Папка для логов (по умолчанию – C:\Logs)
   i. Регистрировать ли конфигурационный файл QP как глобальный на уровне компьютера (запрашивается, если задать параметр из пункта e)
   j. Использовать ли Windows-аутентификацию?
   k. Включать ли сервис расписания статей?
   l. Включать ли общий сервис расписания?
   m. Очищать ли старую инсталляцию?

4. Если выбран минимальный режим, то параметры получат следующие значения:
   a. Имя сайта в IIS – QP8
   b. Порт сайта – 89
   c. Путь к существующему конфигурационному файлу – пустая строка
   d. URL внешнего конфигурационного сервиса – пустая строка
   e. Папка для нового конфигурационного файла – C:\QA;
   f. Токен внешнего конфигурационного сервиса – пустая строка
   g. Папка для временных файлов – C:\Temp
   h. Папка для логов – C:\Logs
   i. Регистрировать ли конфигурационный файл QP как глобальный на уровне компьютера – да
   j. Использовать ли Windows-аутентификацию? – нет
   k. Включать ли сервис расписания статей? – нет
   l. Включать ли общий сервис расписания? – нет
   m. Очищать ли старую инсталляцию? – да

5. Как в случае интерактивного, так и в случае обычного режима, выполняются следующие шаги инсталляции:
   a. Выполняется проверка на существование сайта с заданным именем; если сайт существует и выбрана опция очистки старой инсталляции, то он удаляется, в противном случае установка прерывается с ошибкой.
   b. Выполняется проверка доступности порта, если порт занят, то установка прерывается.
   c. Выполняется проверка на версию ASP.NET Core Runtime, установленную в системе, если нужной версии нет, то установка прерывается.
   d. Выполняется проверка на существование пула, одноименного с сайтом; если пула нет, то он создается.
   e. На основании имени сайта и пути для Default Web Site в IIS, создается папка сайта, туда выполняется копирование файлов.
   f. Разрешается использование windows-аутентификации на уровне IIS.
   g. В случае, если задана папка для нового конфигурационного файла:
      i. Создается папка
      ii. Там создается конфигурационный файл по умолчанию – config.xml

iii. Если задана соответствующая настройка, этот файл регистрируется как глобальный
iv. Производится поиск утилиты sqlmetal на машине, если она найдена, то копируется в папку конфигурационного файла
v. Пользователю, под которым запущен пул QP8, даются права на папку с конфигурационным файлом.

h. Создается папка для логов, на нее даются права пользователю, под которым запущен пул QP8.
i. Создается папка для временных файлов, на нее даются права пользователю, под которым запущен пул QP8.
j. Выполняется настройка файла NLog.config.
k. Выполняется настройка файла appsettings.json.
l. Создается сайт QP8.
m. Пользователю, под которым запущен пул QP8, даются права на wwwroot (папка Default Web Site).

4.2 Установка продукта на Linux

Внимание: при установке на RedOS при наличии selinux его нужно либо отключить, либо перевести в менее строгий режим. Например, можно заменить текст `SELINUX=enforcing` на `SELINUX=permissive`. https://redos.red-soft.ru/base/arm/arm-other/disable-selinux/. Если этого не сделать, то при попытке подключиться к веб-интерфейсу как по IP адресу, так и по DNS-имени выдаётся ошибка 502.

4.2.1 Установка продукта на Linux (с использованием Docker)

Внимание: продукт в данном варианте устанавливается без БД, в предположении, что в дальнейшем вероятно будут устанавливаться продукты QP8.ProductCatalog или QP8.WidgetPlatform со своими БД. Если это не планируется, можно отдельно установить БД (см. Установка БД

Установка БД на Linux)

1. Скачать архив qpconfig.tar и распаковать его содержимое в `/etc/qpconfig`.
2. Перейти в папку `/etc/qpconfig/compose` и выполнить команду: `sudo docker-compose up -d`
3. В результате запустятся три контейнера и приложение будет доступно по адресу http://localhost:5400

4.2.2 Установка продукта на Linux (без использования Docker)

Внимание: продукт в данном варианте устанавливается без БД, в предположении, что в дальнейшем вероятно будут устанавливаться продукты QP8.ProductCatalog или QP8.WidgetPlatform со своими БД. Если это не планируется, можно отдельно установить БД (см. Установка БД

Установка БД на Linux).

Внимание: при отсутствии `dotnet` в каталоге `/bin` можно сделать симлинк на его фактическое местоположение, например `/usr/share/dotnet/dotnet`.

- Создать пользователя группу `qp` и пользователя `qp` в этой группе.

- Скачать архив qpconfig.tar и распаковать его содержимое в `/home/qp/qpconfig`. У пользователя `qp` должны быть права на чтение `qp/qpconfig`.
- В файле `/home/qp/qpconfig/config.xml` в переменной *TempDirectory* указываем путь для временных файлов QP (Пример: `/tmp/qp`). Папка должна существовать и у пользователя `qp` должны быть права на модификацию её содержимого.
- Создать папку `/qplibrary`. Дать пользователю `qp` права на модификацию её содержимого.
- Скачать архив qp.tar.gz (только при использовании скомпилированного кода).

4.2.2.1 Установка сервиса конфигурации

- При сборке из исходников
  - Вытягиваем исходники из репозитория на github.
  - В папке проекта `QP.ConfigurationService\QP.ConfigurationService` (там должен быть файл `QP.ConfigurationService.csproj`) выполняем команду на публикацию:

```
dotnet publish -c Release -o bin/release/publish/ -r linux-x64 --self-contained=false
```

  - В папке `/home/qp` создаём подпапку `QP.ConfigurationService` и копируем туда всё содержимое папки `bin/release/publish`, в которую осуществлялась публикация. Нужно проверить, что у пользователя `qp` есть права на чтение и исполнение содержимого папки `QP.ConfigurationService`.
- При использовании скомпилированного кода:
  - В папке `/home/qp` создаём подпапку `QP.ConfigurationService`
  - Распаковываем содержимое папки `configservice` из скачанного архива `qp.tar.gz` в созданную папку `QP.ConfigurationService`. Нужно проверить, что у пользователя `qp` есть права на чтение и исполнение содержимого папки `QP.ConfigurationService`.
- Переходим в созданную папку `QP.ConfigurationService`
- В файле appsettings.json заменяем `__ConfigurationPath__` на путь к конфигурационному файлу `/home/qp/qpconfig/config.xml`.
- В файле appsettings.json заменяем `__ConfigurationToken__` на значение секрета JWT, например `QP.TestSecret.gwfeP7mbqqLiDmquIrlr`.

Внимание: данное значение секрета приводится только для тестовых целей, для реальных окружений необходимо использовать свои значения. Подробнее о возможности смены значения будет написано далее.

- В файле `NLog.config` находим `internalLogFile` и `<variable name="defaultLogDir" value=` и прописываем там путь `/var/log/qp-config-service/`.
- В файле `NLog.config` в секции `rules` в логгере где пишутся все логи (задан через `*`) в параметре `writeTo` добавляем `web`, чтоб получилось:

```
<logger name="*" writeTo="console, web" minlevel="Info" />.
```

- Создаём папку`/var/log/qp-config-service/` и выдаём пользователю `qp` права на владение этой папкой.
- Создаём файл `qp-config.service` в папке `/usr/lib/systemd/system/` и заполняем его следующим содержимым:

```
 [Unit]
Description=QP Configuration service
```

```
After=network.target
StartLimitIntervalSec=0

[Service]
Type=simple
Restart=on-failure
RestartSec=5
User=qp
WorkingDirectory=/home/qp/QP.ConfigurationService/
ExecStart=/bin/dotnet QP.ConfigurationService.dll --urls http://*:5200

[Install]
WantedBy=multi-user.target
```

Внимание: если в конкретном дистрибутиве Linux отсутствует папка `/usr/lib/systemd/system/`, то все изменения вносим в папку `/lib/systemd/system/`.

1) Прописываем сервис в автозапуск, выполнив команду:

```
systemctl enable qp-config.service
```

2) Запускаем сервис, выполнив команду:

```
systemctl start qp-config.service
```

4.2.2.2 Установка storage-сервиса

- При сборке из исходников:
  - Вытягиваем исходники из репозитория на github.
  - В папке проекта `QP.Storage\QP.Storage.WebApp` (там должен быть файл `QP.Storage.WebApp.csproj`) выполняем команду на публикацию:

```
dotnet publish "QP.Storage.WebApp.csproj" -c Release -o bin/release/publish/ -r linux-x64 --self-contained=false
```

  - В папке `/home/qp` создаём подпапку `QP.Storage` и копируем туда всё содержимое папки `bin/release/publish`, в которую осуществлялась публикация. Нужно проверить, что у пользователя `qp` есть права на чтение и исполнение содержимого папки `QP.Storage`
- При использовании скомпилированного кода:
  - В папке `/home/qp` создаём подпапку `QP.Storage`
  - Распаковываем содержимое папки `storage` из скачанного архива `qp.tar.gz` в созданную папку `QP.Storage`. Нужно проверить, что у пользователя `qp` есть права на чтение и исполнение содержимого папки `QP.Storage`
- В папке `QP.Storage` создать символическую ссылку с названием `wwwroot`, которая ссылается на `/qplibrary`. Проверить, что через символическую ссылку `wwwroot` есть доступ к содержимому `/qplibrary` и у пользователя `qp` есть права на чтение этого содержимого.
- Создаём файл `qp-storage.service` в папке `/usr/lib/systemd/system/` и заполняем его следующим содержимым:

```
[Unit]
```

```
Description=QP Storage
After=qp-config.service
StartLimitIntervalSec=0

[Service]
Type=simple
Restart=on-failure
RestartSec=5
User=qp
WorkingDirectory=/home/qp/QP.Storage/
ExecStart=/bin/dotnet QP.Storage.WebApp.dll --urls http://*:5000

[Install]
WantedBy=multi-user.target
```

Внимание: если в конкретном дистрибутиве Linux отсутствует папка `/usr/lib/systemd/system/`, то все изменения вносим в папку `/lib/systemd/system/`.

3) Прописываем сервис в автозапуск, выполнив команду:

```
systemctl enable qp-storage.service
```

4) Запускаем сервис, выполнив команду:

```
systemctl start qp-storage.service
```

4.2.2.3 Установка QP

- При сборке из исходников:
  - Вытягиваем исходники из репозитория на github.
  - В папке QP выполняем установку npm-пакетов командой `npm ci`.
  - Собираем фронт командой `npm run build`.
  - В папке проекта `QP\siteMvc` (там должен быть файл `WebMvc.csproj`) выполняем команду на публикацию:

```
dotnet publish "WebMvc.csproj" -c Release -o bin/release/publish/ -r linux-x64 --self-contained=false
```

  - В папке `/home/qp` создаём подпапку `QP` и копируем туда всё содержимое каталога `bin/release/publish/`, в который осуществлялась публикация. Нужно проверить, что у пользователя `qp` есть права на чтение и исполнение содержимого папки `QP`.
- При использовании скомпилированного кода:
  - В папке `/home/qp` создаём подпапку `QP`.
  - Распаковываем содержимое папки `backend` из скачанного архива `qp.tar.gz` в созданную папку `QP`. Нужно проверить, что у пользователя `qp` есть права на чтение и исполнение содержимого папки `QP`.
- Переходим в созданную подпапку `QP`.
- В файле `appsettings.json` в секции *Properties* добавляем параметр *QpConfigUrl* и в качестве значения указываем адрес сервиса конфигурации QP который мы развернули ранее. В нашем примере это был `http://localhost:5200`.

- В файле `appsettings.json` в секции *Properties* добавляем параметр *QpConfigToken* и в качестве значения для нашего примера указываем `eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJRUC5Db25maWd1cmF0aW9uU2VydmljZS5Jc3N1ZXIiLCJleHAiOjIwNDQ1NjQ1NjQsImF1ZCI6IlFQLkNvbmZpZ3VyYXRpb25TZXJ2aWNlLkF1ZGllbmNlIn0.Y-gT7zIMGTVU6orvEdQ_yvPA5j2S9lxmb-dccnPqxMU`.

Внимание: данное значение токена приводится только для тестовых целей, для реальных окружений необходимо использовать свои значения. Подробнее о возможности смены значения будет написано далее.

- В файле `NLog.config` находим `internalLogFile` и `<variable name="defaultLogDir" value=` и прописываем там путь `/var/log/qp-service/`.
- В файле `NLog.config` в секции `rules` во всех логгерах, в которых в параметре `writeTo` задано значение `console`, заменяем его на `fileStructured`.
- Создаём папку`/var/log/qp-service/` и выдаём пользователю `qp` права на владение этой папкой.
- Создаём файл `qp.service` в папке `/usr/lib/systemd/system/` и заполняем его следующим содержимым:

```
[Unit]
Description=QP service
After= qp-storage.service
StartLimitIntervalSec=0

[Service]
Type=simple
Restart=on-failure
RestartSec=5
User=qp
WorkingDirectory=/home/qp/QP/
ExecStart=/bin/dotnet Quantumart.QP8.WebMvc.dll --urls http://*:5400

[Install]
WantedBy=multi-user.target
```

Внимание: если в конкретном дистрибутиве Linux отсутствует папка `/usr/lib/systemd/system/`, то все изменения вносим в папку `/lib/systemd/system/`.

5) Прописываем сервис в автозапуск, выполнив команду:

```
systemctl enable qp.service
```

6) Запускаем сервис, выполнив команду:

```
systemctl start qp.service
```

4.2.3 Установка продукта на Linux (с использованием Kubernetes)

Внимание: продукт в данном варианте устанавливается без БД, в предположении, что в дальнейшем вероятно будут устанавливаться продукты QP8.ProductCatalog или QP8.WidgetPlatform со своими БД. Если это не планируется, можно отдельно установить БД (см. Установка БД

Установка БД на Linux)

7) Скачать манифесты, распаковать и перейти в папку `kubernetes.`

8) В файле pv.yaml настроить *persistent volumes* `qp-config` (для конфигурации QP) и `qp-library`. (для библиотеки QP) В примере используются NFS-volumes, но может быть настроен любой тип, поддерживаемый на вашем кластере. В случае использования NFS-volumes, необходимо поправить свойства `path` и `server` в разделе `nfs`, указав адрес сервера и путь на сервере.

Внимание: Если не поддерживается тип режим доступа *ReadWriteMany*, он может быть заменен на *ReadWriteOnce*, но в этом случае будет необходимо настраивать дополнительные правила *affinity,* чтобы все поды, использующие эти *persistent volumes*, запускались на одной ноде кластера.

- Создать *persistent volumes* командой `kubectl apply -f pv.yaml`
- В файле *qp.yaml* можно настроить:
  - Конфигурационный файл QP в *ConfigMap* с именем `qp-app-config`, если например заранее развёрнуты или известны БД, с которыми будет работать QP.
  - Токен и секрет для JWT в *ConfigMap* с именами `qp-config` и `conf-service-config` (по умолчанию используется тестовые).
  - Таймзону для приложения (по умолчанию - Europe/Moscow) в *ConfigMap* с именем `qp-config.`
- Разворачиваем сервисы командой `kubectl apply -f qp.yaml`
- При необходимости можно настроить ингрессы к `qp` и `qp-storage` в файле *ing.yaml*, задав свои DNS вместо `storage.qp.dev.qsupport.ru` и `qp.dev.qsupport.ru` и выполнив команду `kubectl apply -f ing.yaml.`

4.2.4 Изменение JWT-токена

Сгенерировать JWT-токен можно, например, с помощью сайта jwt.io:

1) Вставляем тестовый JWT-токен в поле `Encoded.`
2) При необходимости меняем параметр дату истечения действий токена с помощью параметра *exp* в поле `Payload`.

Замечание: не стоит задавать дату истечения токена в далёком будущем, так как она не пройдёт валидацию в приложении.

1) Задаём желаемый секрет в поле `Verify Signature`. При этом изменится значение токена.
2) Измененный секрет сохраняем в параметрах сервиса конфигурации, а токены – в параметрах QP, как было описано выше.

4.2.5 Настройка nginx

4.2.5.1 При установке с использованием docker

1) Перейдите в папку `/etc/qpconfig/nginx`.

2) В конфигурационном файле `nginx.conf` пропишите актуальные внешние DNS для QP и QP.Storage вместо `qp.test` и `storage.test.`
3) Выполните команду `sudo docker-compose up –d`.

4.2.5.2 При установке без использования docker

1) Найдите конфигурационный файл установленного nginx.
2) Добавьте в него настройки из файла `/home/qp/qpconfig/nginx/nginx.conf`, заменив строчки `qp.test` и `storage.test` на актуальные внешние DNS для QP и QP.Storage.
3) Перезапустите nginx.

5 Установка БД

5.1 Установка БД на Linux

1. Скачать БД для PostgreSQL и распаковать
2. Создать новую пустую БД в PostgreSQL, например, с именем *demosite*
3. Восстановить БД с помощью утилиты `pg_restore` из файла дампа, распакованного на шаге 1, в пустую БД *demosite*, созданную на шаге 2. Пример команды (необходимо использовать реальные логин, пароль и имя сервера вместо *dbuser*, *dbpass* и *dbserver*, вместо DUMP_PATH указывается путь к распакованному файлу дампа):

```
pg_restore -Fc -d 'postgresql://dbuser:dbpass@dbserver/demosite' -j 4 'DUMP_PATH'
```

Замечание: рекомендуется использовать pg_restore версии 12.*, можно использовать версию старше.

4. Выполнить файл `current.sql`, распакованный на шаге 1, на БД *demosite,* предварительно переключив текущий сеанс на схему *qp* с помощью команды:

```
SET search_path TO qp;
```

5. Создать пользователя, например, с именем *demo* и дать ему права на базу данных. Список прав:

```
GRANT CONNECT,TEMP ON DATABASE demosite TO demo;

GRANT USAGE,CREATE ON SCHEMA qp TO demo;

GRANT SELECT,INSERT,UPDATE,DELETE,REFERENCES,TRIGGER ON ALL TABLES IN SCHEMA qp TO demo;

GRANT USAGE,SELECT,UPDATE ON ALL SEQUENCES IN SCHEMA qp TO demo;

GRANT EXECUTE ON ALL ROUTINES IN SCHEMA qp TO demo;

ALTER ROLE demo SET search_path TO qp,public;

ALTER DEFAULT PRIVILEGES IN SCHEMA qp GRANT SELECT,INSERT,UPDATE,DELETE,REFERENCES,TRIGGER ON
TABLES TO demo;

ALTER DEFAULT PRIVILEGES IN SCHEMA qp GRANT USAGE,SELECT,UPDATE ON SEQUENCES TO demo;

ALTER DEFAULT PRIVILEGES IN SCHEMA qp GRANT EXECUTE ON ROUTINES TO demo;

CALL qp_change_contents_ownership('demo');
```

Замечание: если в дальнейшем под созданным пользователем не будет выполняться обновление БД при переходе на новую версию, а вместо этого будет использована административная учётная запись, то права REFERENCES и TRIGGER для таблиц и UPDATE для последовательностей можно не задавать.

6. Если необходимо использовать разных пользователей БД для QP и для сайта, то вот скорректированный набор прав для пользователя сайта (пользователь *demosite*):

```
GRANT CONNECT,TEMP ON DATABASE demosite TO demosite;

GRANT USAGE ON SCHEMA qp TO demosite;

GRANT SELECT,INSERT,UPDATE,DELETE ON ALL TABLES IN SCHEMA qp TO demosite;

GRANT USAGE,SELECT ON ALL SEQUENCES IN SCHEMA qp TO demosite;

GRANT EXECUTE ON ALL ROUTINES IN SCHEMA qp TO demosite;

ALTER ROLE demosite SET search_path TO qp,public;
```

```
ALTER DEFAULT PRIVILEGES IN SCHEMA qp GRANT SELECT,INSERT,UPDATE,DELETE ON TABLES TO demosite;
ALTER DEFAULT PRIVILEGES IN SCHEMA qp GRANT USAGE,SELECT ON SEQUENCES TO demosite;
ALTER DEFAULT PRIVILEGES IN SCHEMA qp GRANT EXECUTE ON ROUTINES TO demosite;
```

7. Добавить в конфигурационный файл QP `config.xml` новую запись `customer`, либо модифицировать дефолтную запись `customer`, указав актуальную строку подключения к БД *demosite*. При этом если необходимо запускать приложение под стандартным пользователем *postgres,* нужно добавить в строку подключения дополнительный параметр для поиска схемы `;Search Path=qp,public;`
8. Настроить установленную базу на ранее развёрнутый QP.Storage, выполнив скрипт (*demosite* можно заменить на желаемое имя папки, остальные настройки соответствуют тому, что было развёрнуто по умолчанию):

```
update site set upload_dir = '/qplibrary/demosite', upload_url = '/demosite',
upload_url_prefix = 'http://localhost:5000'
```

Внимание: при использовании для QP.Storage внешнего DNS необходимо использовать этот внешний DNS вместо `localhost:5000`. Вариант с localhost подходит только если все сервисы разворачиваются на одном компьютере и пользователь будет открывать приложение через браузер также на этом компьютере.

9. Скачать файл с медиа-ресурсами, распаковать его и переложить содержимое папки `demo_site_rus` в папку `/qplibrary/demosite` (если ранее было выбрано другое имя вместо `demosite`, то используйте его). Проверить, что у пользователя QP есть права на модификацию содержимого папки.

Внимание! Если настройки в БД не будут соответствовать физическому расположению файлов, то медиа-ресурсы не будут отображаться на сайте и будет невозможно управлять ими через интерфейс QP.

5.2 Установка БД на Windows

Базы данных для продукта «QP8.CMS с поддержкой PostgreSQL» доступны по ссылкам:

- Для PostgreSQL
- Для SQL Server

Нужно скачать и распаковать дистрибутив (перед распаковкой необходимо разблокировать архив, чтобы в дальнейшем извлекаемые из него файлы были также разблокированы).

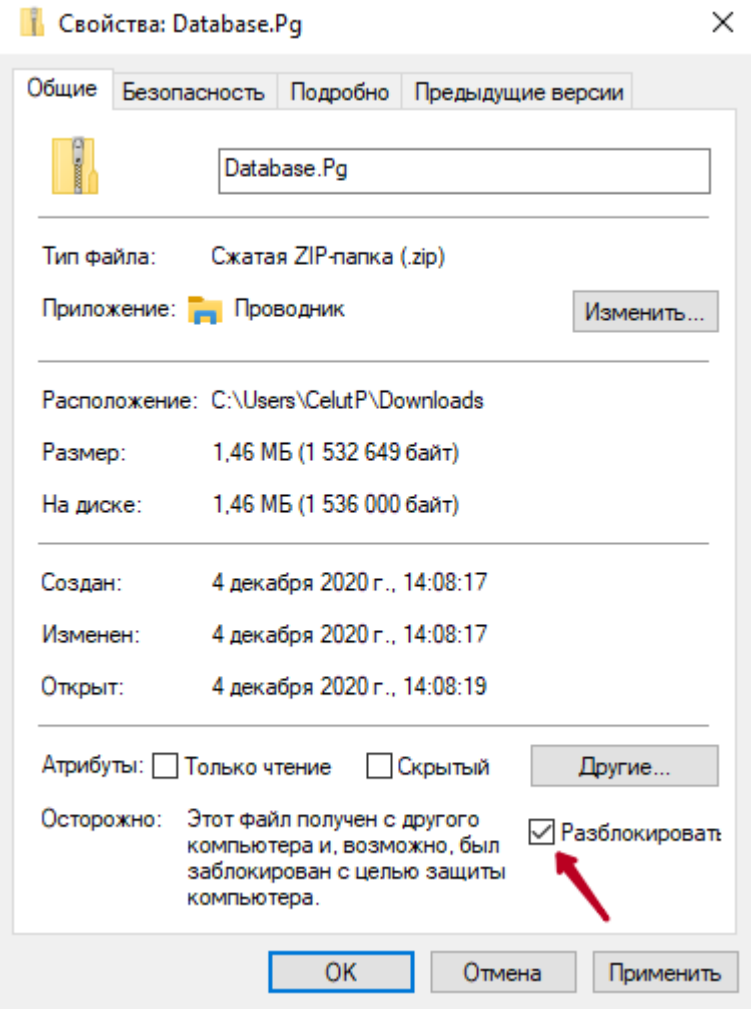


Рисунок 5.1. Разблокировка архива БД.

5.2.1 Минимальная конфигурация

Для установки БД в минимальном режиме (минимум параметров) можно использовать файл `customer_code_pg.bat` (для PostgreSQL) или `customer_code.bat` (для SQL Server). При этом будут запрошены:

1) создаваемый Customer Code,
2) имя сервера БД,
3) логин с административными правами (для восстановления базы),
4) пароль.

Логин и пароль запрашиваются только в варианте для PostgreSQL, для SQL Server будет использована Windows-аутентификация.

5.2.2 Полная конфигурация

Для установки БД в полном режиме (все параметры) можно использовать скрипт `InstallCustomerCodeOnly.ps1`. Кроме параметров минимального режима также можно задать:

1) логин и пароль пользователя, под которым будет работать QP8.CMS,
2) путь к файлу бэкапа на сервере установки,
3) целевой путь файла бэкапа на сервере БД (только для SQL Server),
4) путь к файлу `current.sql`, приводящем базу в актуальное состояние.

6 Обновление продукта

6.1 Обновление продукта на Windows

Процедура обновления продукта совпадает с процедурой установки продукта с включенными опциями очистки старой версии и использования существующего конфигурационного файла.

6.2 Обновление продукта на Linux

6.2.1 Обновление продукта на Linux (с использованием Docker)

1) Перейти в папку `/etc/qpconfig/compose`
2) Поправить значение переменной QP_TAG в файле на актуальный тег образа
3) Выполнить команду: `sudo docker-compose up -d`
4) При необходимости выполнить обновление БД, скачав актуальный архив с БД и выполнив скрипт `current.sql` из него.

6.2.2 Обновление продукта на Linux (без использования Docker)

1) Остановить сервисы командами:

```
systemctl stop qp.service
systemctl stop qp-storage.service
systemctl stop qp-config.service
```

2) Выполнить действия по получению скомпилированной версии продукта аналогичные действиям из процедуры установки.
3) Обновить содержимое папок сервисов, сохранив изменения в конфигурационных файлах `appsettings.json` и `NLog.config`
4) Запустить сервисы командами:

```
systemctl start qp-config.service
systemctl start qp-storage.service
systemctl start qp.service
```

5) При необходимости выполнить обновление БД, скачав актуальный архив с БД и выполнив скрипт `current.sql` из него.

6.2.3 Обновление продукта на Linux (с использованием Kubernetes)

1) Перейти в папку `/etc/qpconfig/kubernetes`
2) Поправить в файле qp.yaml теги образов
3) Выполнить команду: `kubectl apply -f qp.yaml`
4) При необходимости выполнить обновление БД, скачав актуальный архив с БД и выполнив скрипт `current.sql` из него.

7 Настройка продукта

7.1 Конфигурационный файл QP

7.1.1 Конфигурационный файл QP на Windows

Конфигурационный файл QP расположен на веб-сервере в директории продукта.

Примечание: по умолчанию используется имя файла `config.xml`. Местоположение файла содержится в ключе реестра `Configuration File` по пути `HKEY_LOCAL_MACHINE\SOFTWARE\Quantum Art\Q-Publishing`.

Файл содержит:

Внимание: для применения настроек конфигурации требуется перезагрузка веб-приложения бэкенда. Изменения в Customer code применяются без перезагрузки.

- настройки конфигурации продукта,
- параметры доступа к БД (Customer code).

7.1.2 Конфигурационный файл QP на Linux

7.1.2.1 Конфигурационный файл QP на Linux (с использованием Docker)

Конфигурационный файл QP по умолчанию называется `config.xml` и располагается на хостовой машине в папке `/etc/qpconfig` и подключается к контейнеру `qp-configuration` через *volumes.* Изменения в конфигурационном файле опрашиваются и применяются с интервалом в 30 секунд.

7.1.2.2 Конфигурационный файл QP на Linux (без использования Docker)

Конфигурационный файл QP по умолчанию называется `config.xml` и располагается в папке `/home/qp/qpconfig`. Изменения в конфигурационном файле опрашиваются и применяются с интервалом в 30 секунд.

7.1.2.3 Конфигурационный файл QP на Linux (с использованием Kubernetes)

Конфигурационный файл QP по умолчанию располагается в *ConfigMap* с именем `qp-app-config` Изменения в конфигурационном файле применяются при рестарте пода сервиса конфигурации:

```
kubectl rollout restart deployment conf-service -n qp
```

7.1.3 Настройки конфигурации продукта

Содержимое конфигурационного файла является общим для разных вариантов установки.

Синтаксис:

```
<configuration>
  <app_vars>
    ...
    <app_var app_var_name="TempDirectory">C:\Temp\</app_var>
    ...
  </app_vars>
</configuration>
```

Параметры:

Название	Описание

TempDirectory	**Внимание:** необходимо, чтобы пользователь ОС, от имени которого выполняется бэкенд, имел полные права на работу с временной директорией. Временная директория. Может переопределяться в appsettings.json При установке на Windows обычно `C:\Temp` или `C:\Windows\Temp`.
MailHost	Имя SMTP-сервера, через который должны отправляться электронные письма внутренних уведомлений.
MailLogin	Имя пользователя для SMTP-сервера. **Примечание**: если SMTP-сервер поддерживает режим *SMTP Relay* для веб-сервера, то значения для `MailLogin` и `MailPassword` задавать не требуется.
MailPassword	Пароль пользователя для SMTP-сервера. **Примечание**: если SMTP-сервер поддерживает режим *SMTP Relay* для веб-сервера, то значения для `MailLogin` и `MailPassword` задавать не требуется.
MailFrom	Адрес электронной почты для поля From
MailFromName	Имя пользователя для поля From
SqlMetalPath	Путь к SqlMetal. Используется для генерации классов LINQ to SQL. **Примечание:** Настройка применяется только под Windows.
ADsPath	Настройки интеграции с AD (см. Импорт пользователей из AD) **Примечание:** данные параметры больше не используются
ADsConnectionUsername	
ADsConnectionPassword	
ApplicationTitle	Значение Title для страницы с ГПИ бэкенд а. Пример: `<app_var app_var_name="ApplicationTitle">MSCDEV01 QP8 (release {release})</app_var>` Допускается использование параметра `{release}` для вывода номера релиза продукта. **Примечание:** чтобы различать установленные на одном веб-сервере различные экземпляры QP с совпадающим номером релиза, можно задать в `appsettings.json` экземпляра QP значение конфигурации для ключа `InstanceName` в разделе конфигурации `Properties`. Оно будет добавлено к заголовку бэкенда.
AllowSelectCustomerCode	Указатель, что на экране аутентификации требуется предоставлять выбор `Customer Code` из списка доступных значений вместо ручного ввода значения. Допустимые значения параметра: • `yes`, • `no`.

7.1.4 Customer code

Синтаксис:

```
<customer customer_name="demosite" db_type="Postgres">
    <db>Server=pgserver;Port=5432;Database=demosite;User Id=pguser;Password=pgpass</db>
</customer>
```

Возможные атрибуты:

Атрибут	Описание
`exclude_from_schedulers`	Применяется, если требуется, чтобы служба `ArticleSchedulerService` не выполняла проверок для какого-либо Customer Code
`db_type`	Тип базы данных. Принимает значения: • “Postgres” – для БД Postgres • “SqlServer” – для БД MS SQL

7.2 Конфигурационный файл веб-сайта

Ниже представлен локальный конфигурационный файл QP формата JSON и его описание.

```
{
  "Properties": {
    "Version": "#{GitVersion.Tag}#",
    "BuildVersion": "#{GitVersion.Tag}#.#{GitVersion.CommitsSinceTag}#-#{GitVersion.Sha}#",
    "BackendUrl": "",
    "DefaultTheme": "QP8",
    "UploadMaxSize": 1024,
    "Set500ForHandledExceptions": false,
    "EnableArticleScheduler": false,
    "EnableCommonScheduler": true,
    "ExternalAuthentication": {
       "Enabled": true,
       "Type": "ActiveDirectory",
       "DisableInternalAccounts": false
       },
    "QpConfigPollingInterval": "00:00:30",
    "Globalization": {
      "DefaultLanguageId": 1,
      "DefaultCulture": "en-us"
    }
  },
  "Form": {
    "ValueCountLimit": 2048
  },
  "ArticleScheduler": {
    "RecurrentTimeout": "00:00:30"
  },
  "CommonScheduler": {
    "Name": "CommonScheduler",
```

```
    "DefaultUserId": 1,
    "DefaultLanguageId": 1,
    "InterfaceNotificationJob": {
      "Schedule": "0/30 * * * * ?",
      "Description": "Sending notifications"
    },
    "InterfaceCleanupJob": {
      "Schedule": "0 0/10 * * * ?",
      "Description": "Clean up notifications"
    },
    "DisableUsersJob": {
      "Schedule": null,
      "Description": "Task for disabling users who was inactive during InactivePeriodInDays
days. Specified in appsettings.json",
      "SpecifiedConditions": {
        "InactivePeriodInDays": "90"
      }
    },
    "EnableUsersJob": {
      "Schedule": null,
      "Description": "Enabling inactive users. Only for dev",
      "SpecifiedConditions": {
        "ExcludedUsers": "admin"
      }
    },
    "CleanSystemFoldersJob": {
      "Schedule":  null,
      "Description": "Cleaning system folders on file storage which can be limited to
MaxNumberOfFilesPerRun parameter",
      "SpecifiedConditions": {
        "MaxNumberOfFilesPerRun": "5000"
      }
    },
    "UsersSynchronizationJob": {
      "Schedule":  null,
      "Description" : "Synchronizing users with Active Directory"
    }
   }
  },
  "Logging": {
    "IncludeScopes": false,
```

```
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning"
    }
  },
  "Ldap": {
    "Server": "127.0.0.1",
    "UseSsl": true,
    "Port": 389,
    "Domain": "MyDomain",
    "BaseSearchDistinguishedName": "OU=local,DC=MyDomain,DC=test",
    "AdminLogin": "Administrator",
    "AdminPassword": "StrongPass1234",
    "ConnectionTimeout": "00:00:30"
  },
  "KeyCloak": {
    "ApiUrl": "KeycloakAddress",
    "Realm": "realm_name",
    "ClientId": "sync_client",
    "ClientSecret": "sync_secret",
    "AuthClientId": "auth_id",
    "RedirectUrl": "https://QP",
    "EnableSettingName": "KEYCLOAK_ENABLED"
  },
  "Validation": {
    "TextFieldTags": {
      "Enabled": true,
      "LogValidationError": true,
      "AllowedTags": [
        {
          "Tag": "br"
        },
        {
          "Tag": "a",
          "AllowedDomains": ["ya.ru", "google.com"]
        }
      ]
    }
  },
  "S3": {
    "Endpoint": "minio:9000",
```

```
    "AccessKey": "minioLogin",
    "SecretKey": "minioPassword",
    "Bucket": "qp8"
  },
  "Camunda": {
    "IsEnabled": true,
    "Address": "https://camunda7.localhost.ru/engine-rest",
    "Login": "demo",
    "Password": "demo",
    "DefaultFetchCount": 10,
    "WorkerId": "QP_CMS",
    "DefaultLockDuration": "00:01:00",
    "ExternalTaskFetchWaitTime": "00:00:10",
    "ProcessCheckInterval": "00:00:10"
  }
}
```

Название	Тип	Описание
Properties	**Объект**	**Свойства**
Version	Строка	Версия продукта
BuildVersion	Строка	Версия сборки
BackendUrl	Строка	Внутренний URL
DefaultTheme	Строка	Тема по умолчанию
UploadMaxSize	Число	Максимальный размер загрузки в Кб
Set500ForHandledExceptions	Булевый	Установить код ошибки «500» для обработанных исключений
EnableArticleScheduler	Булевый	Использовать планировщик для расписания статей
EnableCommonScheduler	Булевый	Использовать планировщик для Scheduled Tasks
ExternalAuthentication	**Объект**	Настройки внешней аутентификации и синхронизации пользователей
Enabled	Булевый	Включить внешнюю аутентификацию (по умолчанию – false)
Type	Строка	Тип внешней системы. Поддерживаемые значения: "ActiveDirectory" и "KeyCloak". **Примечание:** если данный параметр не указан, то будет использоваться встроенная система аутентификации.
DisableInternalAccounts	Булевый	Отключить аутентификацию через встроенные учётные записи QP (по умолчанию – true). **Примечание:** если данная опция активирована (значение “true”) и в параметре Type не указано ни одно из предлагаемых значений (или соотв. выбранной системе секция конфигурации не

		заполнена), то прохождение аутентификации в QP будет невозможно.
QpConfigPollingInterval	Строка (интервал времени)	Частота опроса конфигурационного файла (под Linux)
Globalization	Объект	Параметры глобализации
DefaultLanguageId	Число	ID языка по умолчанию
DefaultCulture	Строка	Культура по умолчанию
Form	**Объект**	**Форма**
ValueCountLimit	Число	Максимально возможное число полей формы
ArticleScheduler	**Объект**	**Планировщик расписания статей**
RecurrentTimeout	Строка (интервал времени)	Интервал запуска
CommonScheduler	**Объект**	**Планировщик задач по расписанию (Scheduled Tasks)**
Name	Строка	Внутреннее название
DefaultUserId	Число	ID пользователя по умолчанию
DefaultLanguageId	Число	ID языка, используемого по умолчанию
InterfaceNotificationJob	Объект	Параметры задачи отправки уведомлений
Schedule	Строка	Расписание (в cron-формате)
Description	Строка	Описание
InterfaceCleanupJob	Объект	Параметры задачи очистки отправленных уведомлений
DisableUsersJob	Объект	Параметры задачи блокировки пользователей
InactivePeriodInDays	Строка	Период неактивности, по достижении которого пользователь блокируется (в днях)
EnableUsersJob	Объект	Параметры задачи разблокировки пользователей
SpecifiedConditions	Объект	Уточненные условия
ExcludedUsers	Строка	Пользователи, для которых задача не выполняется
CleanSystemFoldersJob	Объект	Параметры очистки системных папок
MaxNumberOfFilesPerRun	Строка	Максимальное число файлов, очищаемых за один раз
UsersSynchronizationJob	Объект	Параметры синхронизации пользователей с AD
Logging	**Объект**	**Раздел настроек логирования**
IncludeScopes	Булевый	Включать области видимости
LogLevel	Объект	Уровни логирования для различных модулей
Default	Строка	Минимальный уровень логирования по умолчанию
Microsoft	Строка	Минимальный уровень логирования для модулей Microsoft
Ldap	**Объект**	**Параметры LDAP-протокола**
Server	Строка	Адрес/доменное имя используемого сервера AD

UseSsl	Булевый	Использовать SSL/TLS канала для взаимодействия с сервером AD
Port	Число	Номер порта, по которому необходимо подключаться к серверу AD
Domain	Строка	Название домена, в котором требуется производить поиск
BaseSearchDistinguishedName	Строка	Название группы, пользователи которой имеют право авторизоваться в QP
AdminLogin	Строка	Логин технической учетной записи, с которой осуществляется поиск учетных записей и групп пользователя
AdminPassword	Строка	Пароль технической учетной записи
ConnectionTimeout	Строка (интервал времени)	Время ожидания ответа от сервера AD при соединении
KeyCloak	**Объект**	**Параметры интеграции с KeyCloak**
ApiUrl	Строка	Базовый адрес KeyCloak
Realm	Строка	Название realm-а
ClientId	Строка	Id клиента, используемого для синхронизации пользователей
ClientSecret	Строка	Токен клиента, используемый для синхронизации пользователей
AuthClientId	Строка	Id клиента, используемый для авторизации пользователей
RedirectUrl	Строка	Адрес, по которому развёрнут QP и куда будет осуществлён callback-вызов после авторизации в KeyCloak
EnableSettingName	Строка	Название настройки, которую необходимо прописать в настройках тенанта для активации SSO для выбранного тенанта. **Примечание:** данный параметр не обязательный. Если он не заполнен, то значение по умолчанию – "KEYCLOAK_ENABLED".
Validation	**Объект**	**Валидация**
TextFieldTags	Объект	Параметры валидации тегов в текстовых полях (подробнее – см. ниже)
Enabled	Булевый	Включение/выключение функции (значение по умолчанию – false)
LogValidationError	Булевый	Включение логирования ошибок валидации тегов (значение по умолчанию – false) **Примечание:** если при попытке сохранения статьи сработает проверка валидации, запрещающая сохранение, то в лог-файл попадёт соответствующая запись с уровнем Warning.
AllowedTags	Массив объектов	«Белый список» (разрешённые теги)

		Примечание: если включена валидация (Enabled = true), но в массиве AllowedTags не указано ничего, то запрещены все теги.
Tag	Строка	Разрешенный тег (задается для каждого объекта массива)
AllowedDomains	Массив строк	Разрешенные домены (опционально задается для каждого объекта массива) **Примечание:** если параметр не задан, то соотв. тег допускается с любыми доменами. При проверке на домены анализируются следующие атрибуты тегов: formaction, codebase, cite, background, srcset, src, href, action, longdesc, profile, usemap, data, classid, icon, manifest, poster, archive.
S3	**Объект**	**Настройки доступа к MinIO**
Endpoint	Строка	адрес MinIO
AccessKey	Строка	ключ доступа к MinIO (логин)
SecretKey	Строка	секрет доступа к MinIO (пароль)
Bucket	Строка	имя бакета
Camunda	**Объект**	**Раздел настроек интеграции с Camunda**
IsEnabled	Булевый	Активация интеграции
Address	Строка	Адрес REST-API платформы Camunda
Login	Строка	Логин в платформе Camunda для взаимодействия с ней через REST-API
Password	Строка	Пароль в платформе Camunda для взаимодействия с ней через REST-API
DefaultFetchCount	Число	Количество задач, которые будут выбраны для обработки за один раз
WorkerId	Строка	Идентификатор обработчика. **Примечание:** если QP запускается в нескольких экземплярах, то у каждого экземпляра должен быть свой уникальный WorkerId.
DefaultLockDuration	Строка (интервал времени)	Время, на которое выбранная для выполнения задача блокируется
ExternalTaskFetchWaitTime	Строка (интервал времени)	Период частоты, с которой QP будет проверять наличие задач для обработки
ProcessCheckInterval	Строка (интервал времени)	Период частоты, с которой QP будет проверять завершенные задачи в Camunda и выставлять соответствующий статус для процесса в QP

Примечание: жирным в таблице выделены ключи первого порядка (в иерархии записи).

7.2.1 Серверная валидация HTML-тегов

Возможность ограничения набора HTML-тегов, допустимых для сохранения в текстовых полях QP, обеспечивает дополнительную защиту сайта. Набор разрешенных тегов (т. н. «белый список») задается в конфигурации QP (файл `appsettings.json`). Для тегов, у которых поддерживается указание источника (например, через атрибут `src`) возможно задать список разрешенных доменов. В корневой элемент добавляется секция `Validation` с вложенностью `TextFieldTags`.

В разделе 7.2 приведён пример файла конфигурации, разрешающего тег «`br`» и тег «`a`», у которого разрешены домены «`ya.ru`» и «`google.com`».

Ввод запрещенных (не входящих в заданный набор) тегов/доменов в поле типов «Файл», «Изображение», «Строка», «Текстовое поле», «Визуальный редактор» при попытке сохранения повлечет возникновение ошибки.

7.3 Конфигурационный файл QP.Storage

```
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft": "Warning",
      "Microsoft.Hosting.Lifetime": "Information"
    }
  },
  "AllowedHosts": "*",
  "ImageResizeSettings": {
    "IsResizeAllowed": false,
    "SearchParameter": "size",
    "WidthParameter": "w",
    "ExtensionsAllowedToResize": [
      ".png",
      ".jpg",
      ".jpeg",
      ".webp",
      ".gif"
    ],
    "ResizedImageTemplate": "{0}-{1}{2}",
    "ReduceSizes": [
      {
        "ReduceRatio": 2,
        "Postfix": "medium"
      },
      {
        "ReduceRatio": 3,
        "Postfix": "small"
      }
    ]
  },
  "CustomExtensions": {
    ".avif": "image/avif"
  }
}
```

Название	Тип	Описание
Logging	**Объект**	**Раздел настроек логирования**
LogLevel	Объект	Уровни логирования для различных модулей
Default	Строка	Минимальный уровень логирования по умолчанию
Microsoft	Строка	Минимальный уровень логирования для модулей Microsoft
Microsoft.Hosting.Lifetime	Строка	Минимальный уровень логирования для модулей Microsoft.Hosting.Lifetime
AllowedHosts	**Строка**	**Допустимые хост-серверы**
ImageResizeSettings	**Объект**	**Настройки ресайза изображения**
IsResizeAllowed	Булевый	Возможен ли ресайз изображения
SearchParameter	Строка	Название параметра, в котором передаётся постфикс
WidthParameter	Строка	Параметр, в котором передаётся ширина, по которой вычисляется постфикс
ExtensionsAllowedToResize	Массив	Допустимые для ресайза форматы графических файлов
ResizedImageTemplate	Строка	Шаблон, по которому генерируются имена уменьшенных изображений: • `{0}` – имя исходного файла, • `{1}` – постфикс, • `{2}` – расширение.
ReduceSizes	Объект	Параметры уменьшения размера
ReduceRatio	Число	Кратность уменьшения размера изображения
Postfix	Строка	Постфикс, обозначающий тип размера измененного изображения. Значения по умолчанию: • `medium` (уменьшение в 3 раза), • `small` (уменьшение в 2 раза).
CustomExtensions	**Объект**	Параметр полезен в случаях, когда в QP Library загружен файл опр. расширения, но при попытке получить файл через QP Storage возникает ошибка 404. **Примечание:** как правило, это свойственно редким кастомным расширениям или новым форматам, которые ещё не добавлены в поддержку .NET. Расширение файла можно указать в первом столбце, а во втором – тип контента, с которым этот файл будет возвращён.
.avif	Строка	Тип контента. Список существующих типов контента можно посмотреть по ссылке. **Примечание:** в данном случае у изображения `.avif` указан тип контента “image/avif”. При переходе по ссылке на файл он откроется в браузере как рисунок. Если же указать “application/octet-stream”, то файл будет просто скачан браузером и сохранён на диск.

Примечание: жирным в таблице выделены ключи первого порядка (в иерархии записи).

Ниже приведены примеры запросов API:

Пример запроса	Логика работы

https://storage_url/images/image_name.jpg?size=small	При передаче параметра `size` запускается обработка изображения. Значение параметра (в данном случае `small`) должно содержаться в конфигурационном файле (секция `ReduceSizes`, ключ `Postfix`). Если значение не найдено, выдается исходное изображение.
https://storage_url/images/image_name.jpg?width=200	При передаче параметра `width` (в данном случае 200 пикселей) вычисляется необходимый коэффициент уменьшения и выбирается наиболее подходящее среди значений `ReduceRatio`. Если значение не найдено, выдается исходное изображение.
https://storage_url/_reduce_settings	Запрос на получение параметров автоматического ресайза. Возвращает набор ключей и их значений секции `ImageResizeSettings`.

7.4 Настройки аутентификации

7.4.1 Настройки для Windows

7.4.1.1 Аутентификация и права доступа на директории

По умолчанию в QP используется анонимная аутентификация (см. Аутентификация IIS).

Мастер установки наделяет требуемыми правами права пользователя, под которым запущен мастер:

1) на директорию по умолчанию для IIS (обычно `C:\inetpub\wwwroot`),
2) на временную директорию (см. Временная директория и директория под лог-файлы).

Прочие права требуется назначать вручную. Права на изменение следует выдавать либо на директории сайтов, с которыми требуется вести работу, либо на директорию `wwwroot` полностью (если необходима возможность создания новых сайтов средствами QP).

В случае анонимной аутентификации нужно дать права:

1) пользователю, под которым запущен пул приложений бэкенда (по умолчанию – `IIS AppPool\`*имя пула*),
2) анонимному пользователю (по умолчанию –`IUSR`).

Пользователь ОС, от имени которого выполняется бэкенд, должен иметь право «Modify» на:

1) директорию сайта (или отдельно на директории для Live- и Stage-режимов сайта),
2) временную директорию.

7.4.1.2 Временная директория и директория под лог-файлы

Временная директория и директория под лог-файлы размещены в директории QP, имена директорий – `Temp` и `Logs` соответственно.

Для изменения временной директории следует:

1) создать временную директорию по нужному пути (например, `C:\temp`);
2) дать на неё полные права пользователю, под которым выполняется бэкенд;
3) задать новый путь в конфигурационном файле QP (см. Настройки конфигурации продукта).

Для изменения директории под лог-файлы требуется задать путь до новой директории в файлах `NLog.config` приложений `Backend` и `WinLogon` (если используется).

7.4.1.3 Настройка IIS

Внимание: настройки сайта QP должны соответствовать настройкам веб-сайта IIS так, чтобы физические пути соответствовали виртуальным.

Основные понятия

Процесс (англ. «Process») – объект, который создается ОС после запуска веб-приложения определённым пользователем. Процессу выделяется отдельное адресное пространство, причём это пространство недоступно для других процессов. Для каждого процесса ОС создаёт один главный поток (англ. «thread»), который является потоком выполняющихся команд. При необходимости главный поток может создавать другие потоки, пользуясь для этого программным интерфейсом ОС.

Олицетворение (англ. «Impersonation») – механизм, позволяющий назначить текущему потоку реквизиты некоторого пользователя, что позволяет выполнять код от его имени. Данный механизм часто используется в веб-приложениях, когда один и тот же поток обслуживает запросы от различных пользователей.

Пул приложений

Каждый пул приложений (англ. «Application Pool») представляет собой пул потоков. Принцип работы пула:

- при старте процесса создаётся заданное число потоков;
- при поступлении HTTP-запроса из пула берется поток, чтобы его выполнить;
- если все потоки заняты, то используется механизм очередей;
- после выполнения веб-сервером запроса поток возвращается в пул.

Просмотр и настройка пользователей, от имени которых выполняется процесс в IIS 7+: *Имя сервера* → Application Pools → *Имя пула* → Advanced Settings → Process Model → Identity.

В качестве пользователя по умолчанию (если иное не определено в `Application Pool Defaults`) выступает специальное понятие `ApplicationPoolIdentity`, которое означает, что для каждого пула с такой настройкой будет создан специальный пользователь в группе `IIS AppPool`, имя которого совпадает с названием пула.

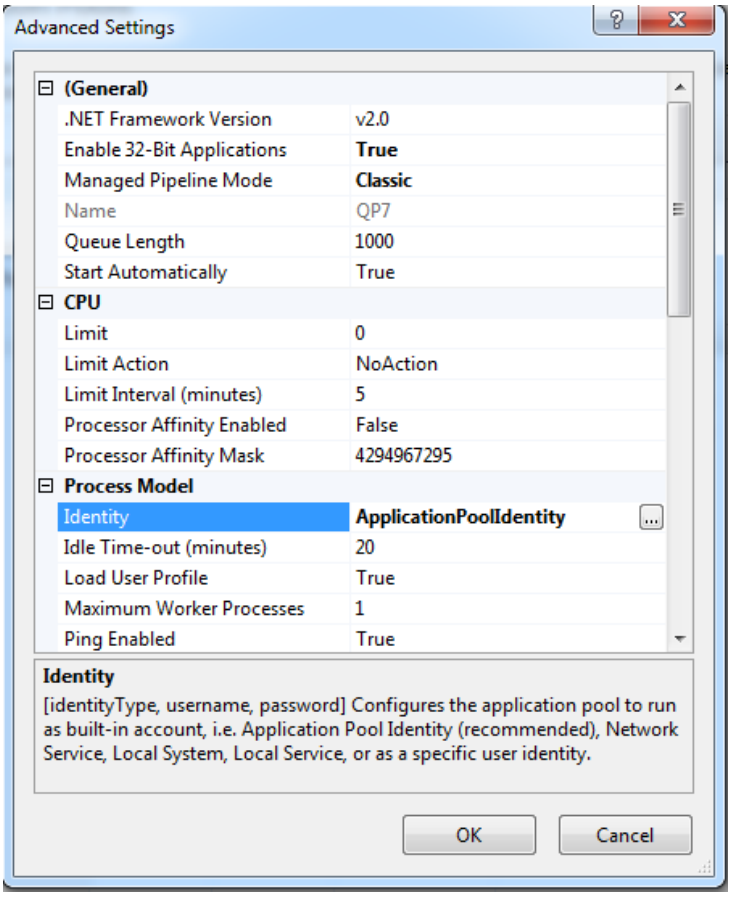


Рисунок 7.1. Настройки пула приложений.

Олицетворение

- По умолчанию любой поток из пула работает под тем же пользователем, что и сам пул, но возможно включение механизма олицетворения.
- Для ASP.NET-приложений олицетворение по умолчанию выключено и настраивается в конфигурационном файле веб-сайта (`web.config`).
- Если олицетворение включено, то потоку назначаются реквизиты пользователя, аутентифицированного IIS.

Аутентификация IIS

Аутентификация настраивается на уровне сайта (приложения) IIS: *Имя сайта (приложения)* → IIS → Authentication.

Примечание: если включены анонимная и Windows-аутентификации, то анонимная аутентификация имеет приоритет.

Анонимная аутентификация

Фактически аутентификации пользователя не происходит, а все запросы выполняются от имени специального служебного пользователя. В случае включённого механизма олицетворения веб-приложение будет запущено от его имени. Имя пользователя по умолчанию: `IUSR`.

Анонимный пользователь может быть изменён в свойствах анонимной аутентификации.

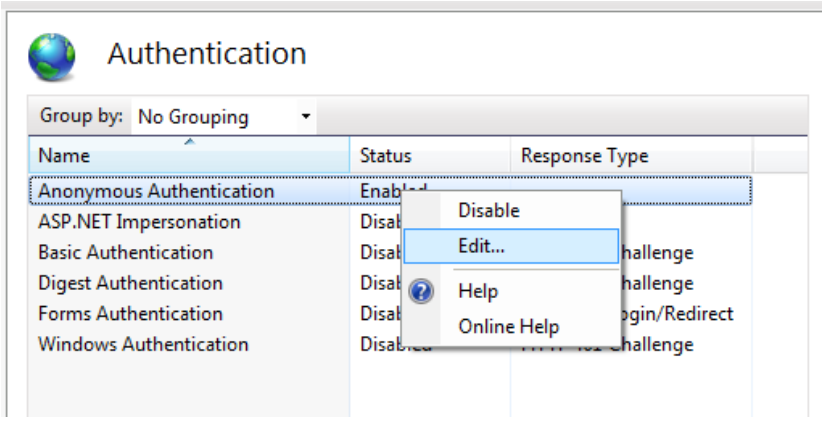


Рисунок 7.2. Изменение параметров анонимной аутентификации.

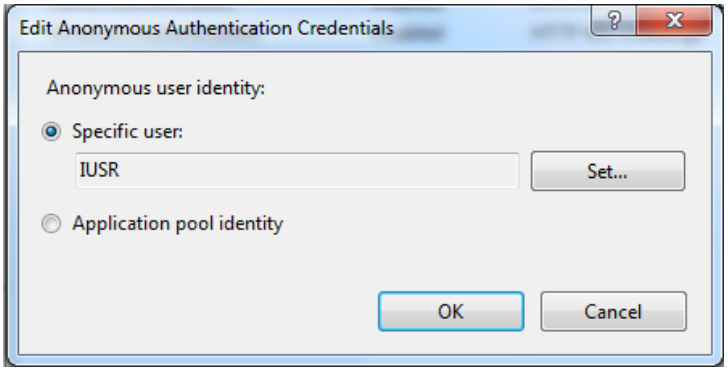


Рисунок 7.3. Параметры анонимной аутентификации.

Аутентификация Windows

Если пользователь находится внутри домена и использует браузер, поддерживающий протокол аутентификации Kerberos, то он будет автоматически аутентифицирован под своей учётной записью для работы с ОС. В противном случае Windows попросит ввести логин и пароль. При успешном прохождении Windows-аутентификации и включённом механизме олицетворения веб-приложение будет запущено от имени аутентифицированного пользователя.

Группа IIS_IUSRS

Внимание: пользователь `IUSR` не входит в группу `IIS_IUSRS`.

В IIS 7.0 существует группа `IIS_IUSRS`, в которую автоматически включаются все пользователи, под которыми запущены `Application Pools`.

О веб-приложениях

Веб-приложение, в отличие от Windows-приложения, является виртуальным. В рамках одного процесса может работать много веб-приложений. Когда мы говорим о том, под каким пользователем работает веб-приложение, фактически мы рассматриваем следующие вещи:

- Под каким пользователем работает Application Pool?
- Включен ли механизм олицетворения?
- Если включен, то какие настройки аутентификации приложения заданы в IIS?

Когда все настройки по умолчанию

ASP.NET-приложение выполняется под пользователем `IIS AppPool\`*Название пула*.

Порождение дочерних процессов

Если порождается дочерний процесс (например, с помощью `System.Diagnostics.Process.Start`), то он не учитывает олицетворение и по умолчанию запускается от имени пользователя, под которым запущен родительский процесс.

7.4.2 Интеграция с внешними системами

7.4.2.1 Интеграция с AD

Для активации возможности входа в интерфейс QP8.CMS через доменные учётные записи требуется произвести дополнительную настройку сервиса.

Для этого в конфигурацию («Appsettings» или «Параметры окружения») необходимо добавить секцию «Ldap» (см. Конфигурационный файл веб-сайта).

Примечание: параметр «Port» не является обязательным. Если его не указывать, то:
- если "UslSsl" = true, то используется порт 636;
- если "UslSsl" = false, то используется порт 389.

После заполнения настроек и перезапуска QP вход возможен с помощью доменных учётных записей AD. Для этого на странице авторизации в поле ввода логина необходимо указать логин в формате «domain\username», например «mydomain\administrator».

Внимание: для входа пользователь AD должен являться членом группы, указанной в параметре «BaseSearchDistinguishedName» конфигурации.

Если на странице авторизации указать имя учётной записи без указания домена, то используется стандартная авторизация по встроенным учётным записям в QP.

7.4.2.2 Интеграция с KeyCloak

Внимание! Системы AD и KeyCloak взаимоисключающие: запустить можно лишь одну интеграцию, но не обе одновременно.

Интеграция QP8.CMS с системой KeyCloak состоит из двух частей:

1) синхронизация пользователей;
2) аутентификация.

Для включения SSO и интеграции с KeyCloak требуется произвести дополнительную настройку сервиса через файл `appsettings.json` (см. Конфигурационный файл веб-сайта):

1. В секции `Properties`, подсекции `ExternalAuthentication`, следует:
   a) установить значение `true` для параметра `Enabled`;
   b) указать значение `KeyCloak` для параметра `Type`.

2. Необходимо добавить новую секцию `KeyCloak`.

Для успешной интеграции в KeyCloak должны быть созданы два клиента в указанном realm-е. При этом:

1. Клиент для синхронизации:
   - должен иметь доступ к чтению списка групп;

- должен иметь доступ к чтению списка пользователей групп;
- должен поддерживать flow `client_credentials`.

2. Клиент для авторизации пользователей:

- должен поддерживать flow `authorization_code`;
- в качестве адреса редиректа у него должен быть указан адрес `http[s]://qp/LogOn/SsoCallback`, где `qp` – адрес, по которому развёрнут QP (должен совпадать со значением параметра `RedirectUrl`);
- в качестве `Allowed Origins` у него должен быть указан адрес QP.

Настройки ГПИ

В ГПИ в настройках Customer Code следует:

1) активировать опцию "Включить синхронизацию пользователей с внешней системой";
2) в список настроек приложения добавить ключ с именем, указанным в файле конфигурации в параметре `EnableSettingName`, и значением `True` (Рисунок 7.4).

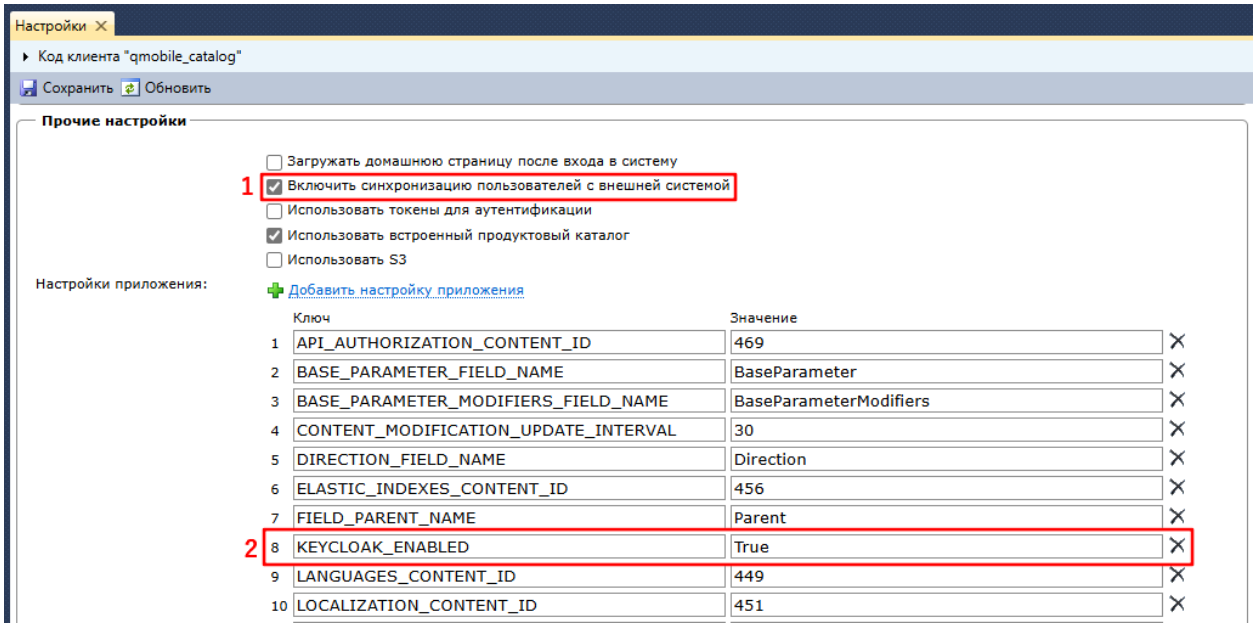


Рисунок 7.4. Настройки Customer Code для интеграции с KeyCloak

По указанному в настройках расписанию происходит синхронизация пользователей из KeyCloak в QP.

7.5 Страница «Настройки» корневого элемента дерева

На странице доступны общие настройки бэкенда. Доступ к странице осуществляется через контекстное меню корневого элемента дерева бэкенда (пункт «Настройки» (Settings)) (Рисунок 7.5).

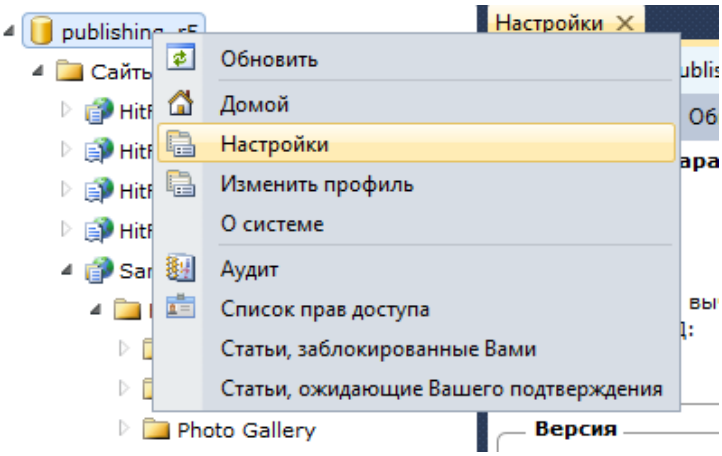


Рисунок 7.5. Настройки в контекстном меню

В таблице ниже описаны функциональные блоки настроек и параметров данной страницы.

Таблица 1. Параметры страницы «Настройки»

Название		Описание
Записывать действия (Record actions) (Рисунок 7.6)	Записывать действия в файл (Record actions into file)	см. Запись и воспроизведение действий
	Очистить файл, если он существует (Empty file if exists)	Применяется, если действия нужно записывать в пустой файл. В противном случае запись осуществляется в конец существующего файла. **Внимание:** данная настройка не сохраняется (разовая). Настройка появляется только при выборе опции «Записывать действия в файл).
	Скачать действия в файл (Download recorded action)	Соответствующие записи скачиваются в файл формата XML
Проиграть действия (Replay actions) (Рисунок 7.6)	XML с записанными действиями (XML with recorded actions)	Текстовое поле, куда помещается содержимое XML-файла с записью действий для воспроизведения
	Генерировать новые ID полей (Generate new field ids)	Указывает на то, что в ходе воспроизведения требуется создавать новые значения идентификаторов полей
	Генерировать новые ID контентов (Generate new content ids)	Указывает на то, что что в ходе воспроизведения требуется создавать новые значения идентификаторов контентов
	Использовать замену GUID (Use GUID substitution)	Указывает на то, что в ходе воспроизведения требуется использовать GUID статей. **Примечание:** при отключённой опции используются идентификаторы статей.
	Проиграть действия (Replay actions)	Кнопка нажимается после ввода данных для воспроизведения и задания требуемых параметров.

Прочие настройки (Other settings) (Рисунок 7.7)	Загружать домашнюю страницу после входа в систему (Load home page after logon)	Указывает, требуется ли автоматически открывать страницу «Домой» после авторизации пользователя.
	Включить синхронизацию пользователей с внешней системой (Turn on users synchronization with external system)	Указывает, требуется ли использовать синхронизацию пользователей бэкенда с пользователями из внешних систем (Active Directory, KeyCloak и пр.).
	Использовать токены для аутентификации (Use authentication tokens)	Указывает, требуется ли использовать токены для аутентификации в системе.
	Использовать встроенный продуктовый каталог (Use internal DPC)	Указывает, требуется ли использовать встроенный продуктовый каталог.
	Использовать CDC для отправки системных уведомлений (Use CDC for system notifications)	Указывает, требуется ли использовать механизм Change Data Capture в процессе отправки системных уведомлений.
	Использовать S3 (Use S3)	Активирует интеграцию с MinIO
	Настройки приложения (Application Settings)	Позволяет хранить произвольные данные «ключ-значение» (Key-Value). Данные можно получить методом GetAppSettings.
	Добавить настройку приложения (Add application setting)	Кнопка добавления новой настройки. В качестве значения чаще всего используется ID создаваемого элемента.
Версия (Version) (Рисунок 7.7)	ID	ID корневого элемента (= 1)
	Создано (Criated)	Дата и время создания корневого элемента дерева
	Изменено (Modified)	Дата и время последнего изменения корневого элемента дерева
	Изменил (Last Modified By)	Пользователь, осуществивший последнее изменение
Прямая ссылка (Direct Link) (Рисунок 7.7)	Url	URL-адрес

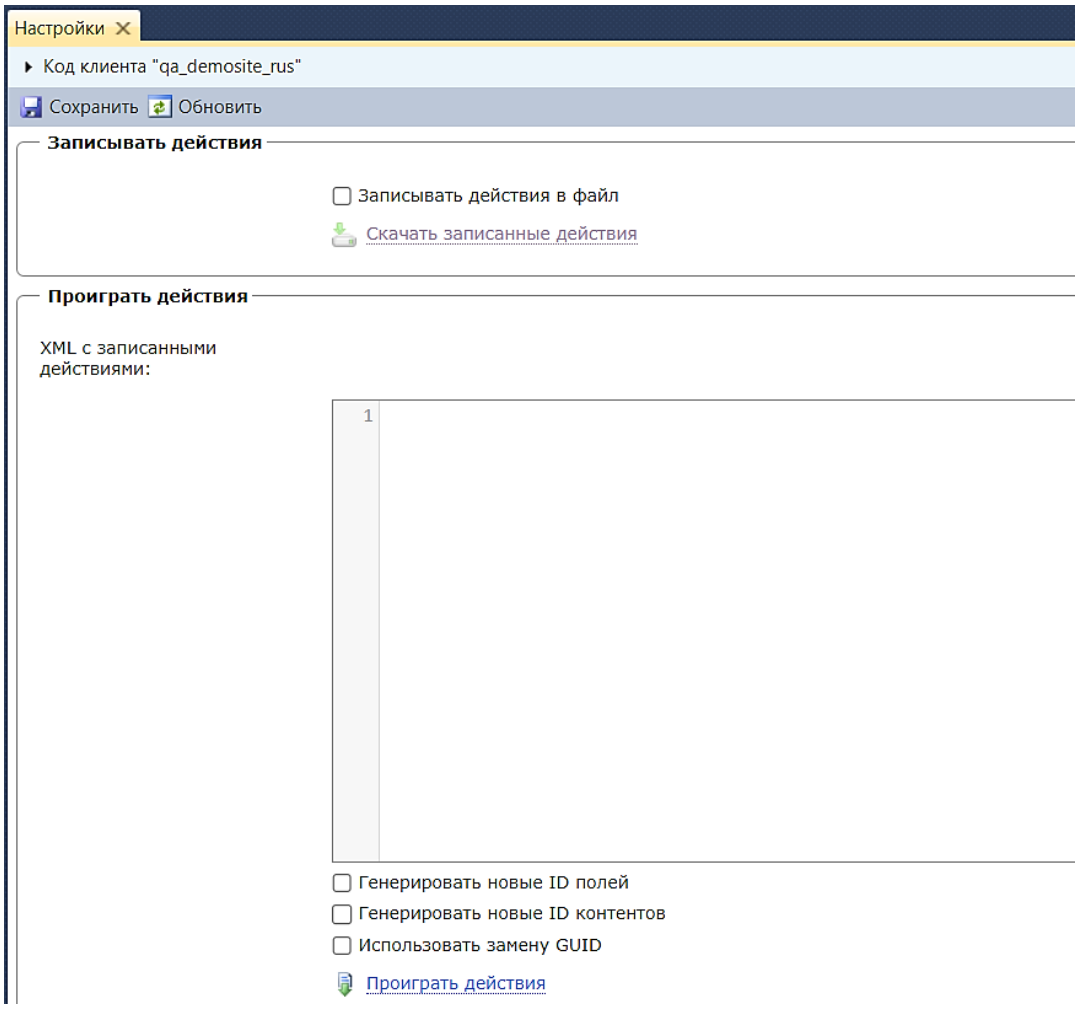


Рисунок 7.6. Страница «Настройки». Блоки «Записывать действия» и «Проиграть действия»

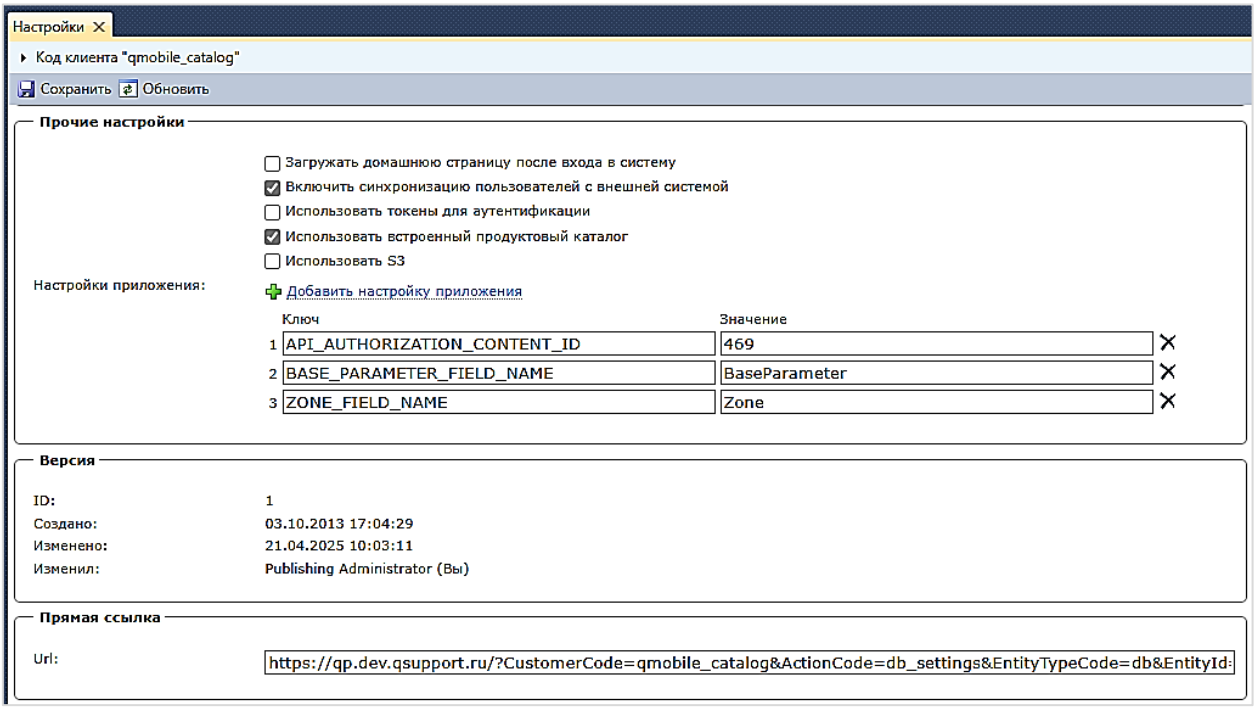


Рисунок 7.7. Страница «Настройки». Блоки «Прочие настройки», «Версия» и «Прямая ссылка»

Сохранять изменения и обновлять содержимое страницы позволяют кнопки «Сохранить» и «Обновить» (Рисунок 7.6).

7.6 Запись и воспроизведение действий

Существует функциональная возможность записать в файл последовательность всех действий пользователей в бэкенде (в пределах текущего Customer Code), приводящих к изменению данных в БД, а также в дальнейшем автоматически выполнить в бэкенде все действия из подобного файла.

Внимание: в текущей версии QP не поддерживается запись следующих действий:

- действия, связанные с изменением и удалением дочерних прав доступа;
- «Пересоздать изображения»;
- «Применить значение по умолчанию»;
- все действия с файлами (предполагается, что копирование файлов будет осуществлено отдельно).

Возможность полезна в ситуациях, когда Система существует в нескольких окружениях, при этом в части окружений возможность вносить изменения прочими штатными средствами QP отсутствует или занимает много времени, например, из-за корпоративных политик. Запись действий осуществляется в формате XML.

Примечание: данные по записи действий приведены в руководстве разработчика.

Существует несколько способов для воспроизведения действий:

1) через ГПИ бэкенда,
2) с помощью программы QP8.ConsoleDbUpdate:
    - через CLI,
    - через Docker CLI,
3) с использованием QP8 Backend API.

При воспроизведении не допускается повторное выполнение действия, которое уже было выполнено ранее (выполняется проверка значения хэша действия). Если запись действий содержит уже выполненное действие, то при воспроизведении оно пропускается.

Ведётся лог следующих операций:

1) воспроизведение действий (см. Лог операций воспроизведения действий),
2) запуск воспроизведения (см. Лог операций запуска воспроизведения).

7.6.1 Воспроизведение через ГПИ бэкенда

См. Интеграция с KeyCloak

Внимание! Системы AD и KeyCloak взаимоисключающие: запустить можно лишь одну интеграцию, но не обе одновременно.

Интеграция QP8.CMS с системой KeyCloak состоит из двух частей:

синхронизация пользователей;

аутентификация.

Для включения SSO и интеграции с KeyCloak требуется произвести дополнительную настройку сервиса через файл `appsettings.json` (см. Конфигурационный файл веб-сайта):

3) В секции Properties, подсекции `ExternalAuthentication`, следует:
4) установить значение `true` для параметра `Enabled`;

указать значение KeyCloak для параметра Type.

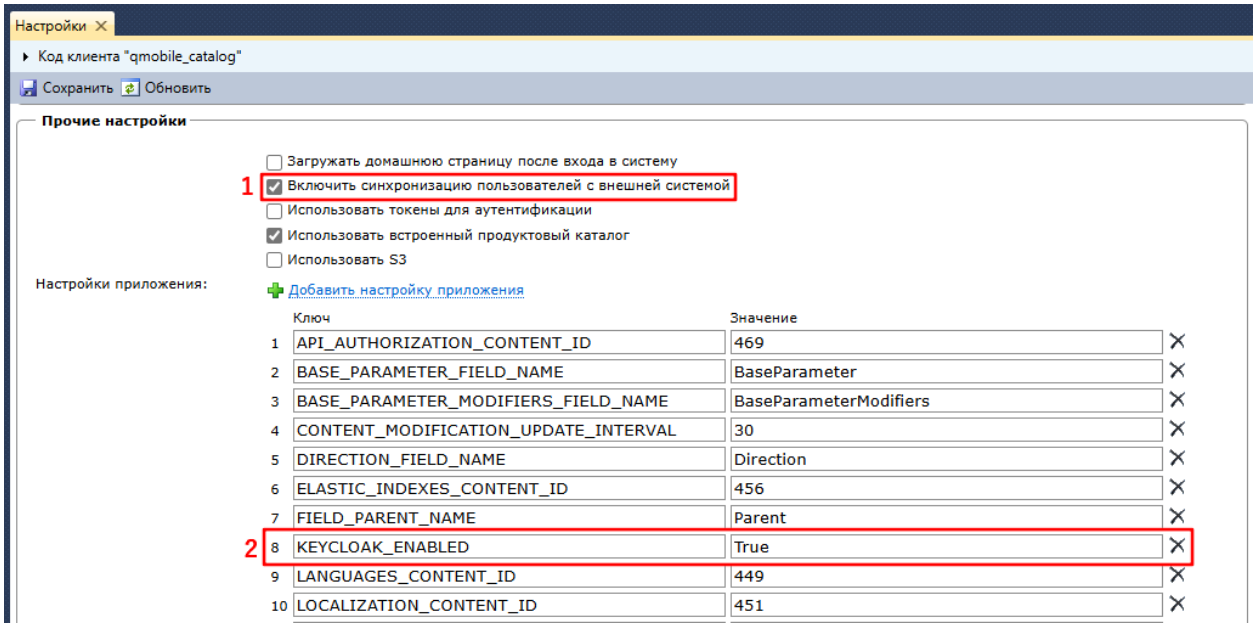
Необходимо добавить новую секцию KeyCloak.

Для успешной интеграции в KeyCloak должны быть созданы два клиента в указанном realm-е. При этом:

3. Клиент для синхронизации:
4. должен иметь доступ к чтению списка групп;
   c) должен иметь доступ к чтению списка пользователей групп;
   d) должен поддерживать flow client_credentials.
   e) Клиент для авторизации пользователей:
5. должен поддерживать flow authorization_code;

в качестве адреса редиректа у него должен быть указан адрес http[s]://qp/LogOn/SsoCallback, где qp – адрес, по которому развёрнут QP (должен совпадать со значением параметра RedirectUrl);

3. в качестве Allowed Origins у него должен быть указан адрес QP.
   - Настройки ГПИ
   - В ГПИ в настройках Customer Code следует:
   - активировать опцию "Включить синхронизацию пользователей с внешней системой";
   - в список настроек приложения добавить ключ с именем, указанным в файле конфигурации в параметре EnableSettingName, и значением True (Рисунок 7.4).



4.
5. Рисунок 7.4. Настройки Customer Code для интеграции с KeyCloak
   - По указанному в настройках расписанию происходит синхронизация пользователей из KeyCloak в QP.

Страница «Настройки» корневого элемента дерева.

7.6.2 Программа QP8.ConsoleDbUpdate

7.6.2.1 Общие правила использования и настройки

Все именованные аргументы передаются через пробел, например «mode xml». Для работы ПО требуется передать ряд обязательных аргументов:

Обязательный аргумент	Описание
`-m|--mode`	Тип файла, содержащего запись действий. Определяет какой из обработчиков файлов будет использован. Допустимые значения: • `xml` – XML-файл; • `csv` – CSV-файл. Пример: `--mode xml`
`customer_code|connection_string`	Customer code или строка подключения к БД QP. Неименованный параметр.

Файл с записью действий можно передать одним из следующих способов:

Способ передачи файла	Описание
`piped input`	Передача потока данных файла через pipe или перенаправление потока в консоли
через аргумент `-p|--path`	Путь к директории или файлу с записью. В данном случае будут обработаны все файлы в директории, субдиректории будут проигнорированы. Можно указать: • путь к директории, например `--path /qpactions/`, • путь к файлу, например `--path /qpactions/action.xml`, • пути к директориям, например `--path /qpactions1/ --path /qpactions2/`, • пути к файлам, например `--path /qpactions1/action1.xml --path /qpactions2/action2.xml`, • пути к директориям и файлам, например `--path /qpactions1/ --path /qpactions2/action.xml`. **Примечание:** в директориях происходит поиск файлов по расширению в зависимости от выбранного режима (см. `-m|--mode`). При передаче файла с неверным расширением он будет проигнорирован.

Корректно настроить приложение на работу с QP можно одним из следующих способов:

1. Передача строки подключения в виде неименованного аргумента. В этом случае приложение выполнит подключение к БД и применит изменения в этой БД. В случае, если строка подключения содержит пробелы, её необходимо обрамлять кавычками.
2. Передача `customer_code` в виде неименованного аргумента. В данном случае вместо строки подключения возможны следующие варианты:

    1) передача аргумента `-c|--config` (путь к файлу конфигурации QP): ПО проанализирует файл конфигурации и попытается найти строку подключения по переданному `customer_code`;
    2) передача аргументов `-r|--remoteConfig` (url к сервису конфигурации QP) и `-j|--jwt` (JWT-токен для доступа к нему): ПО запросит в сервисе конфигурации строку подключения по переданному `customer_code`;

3) без передачи дополнительных аргументов: в этом случае сервис попытается обратиться в реестр Windows и найти там строку подключения для указанного `customer_code`.
Примечание: данный способ работает только в Windows. Для его корректной работы требуется запускать ПО на том же устройстве, на котором запущена QP.

Ниже перечислены дополнительные настройки приложения:

Настройка	Описание
`-t|--type`	Тип используемой БД (`0` - SqlServer, `1` - Postgres). Применяется только при передаче строки подключения или `customer_code` без дополнительных аргументов. В остальных случаях тип БД берётся из используемого источника конфигурации.
`-v|--verboseLevel`	Уровень логирования. Поддерживаются уровни: `[v|vv|vvv]:[error|warning|info]`
`-s|--silent`	Отключение запроса подтверждения пользователя и вывода конфигурации на экран.
`--disablePipedInput`	Принудительное отключение `piped-input`. Полезно для использования приложения в Docker (не в интерактивном режиме).
`-h|--help`	Показывает справку о работе приложения

Ниже перечислены настройки, зависимые от режима работы (см. `-m|--mode`):

Режим работы	Настройка	Описание
CSV	`-e|--encoding`	Кодировка экспортированного файла. Должна совпадать с кодировкой, выбранной при экспорте из QP. По умолчанию — `windows-1251`.
	`-l|--culture`	Культура, в которой сохранены файлы. По умолчанию — `ru-RU`.
	`-u|--updateExisting`	Указатель необходимости обновлять существующие статьи. По умолчанию — `false`.
XML	`-g|--generateIds`	Показывает, необходимо ли генерировать новые ID. Допустимые значения: • `field` – генерировать новые идентификаторы для полей; • `content` – генерировать новые идентификаторы для контентов. По умолчанию отключено.
	`-d|--disable`	Аналогично настройке `-g|--generateIds`
	`--useGuid`	Указывает, что требуется использовать GUID статей. При отключённой опции используются ID статей.
	`--disableDataIntegrity`	Отключить проверку на повторное выполнение XML

7.6.2.2 Использование программы через Docker CLI

1. Для использования файлов требуется совместное подключение `volume`. Пример: `docker run -v /host_path/:/container_path/ --rm ImageName Args`. После этого `container_path`

передается в качестве отправной точки в аргументах. Однако по умолчанию приложение работает в интерактивном режиме и запросит у пользователя подтверждение. Чтобы этого избежать, нужно передать в качестве аргумента ключ `-s|--silent`.

2. По умолчанию контейнер запускается не в интерактивном режиме, следовательно, приложение будет работать в `pipe-mode` и ожидать передачи файла в себя через `pipe` или перенаправление в консоли. Для запуска приложения в интерактивном режиме следует использовать ключи `-it`, например: `docker run -v /host_path/:/container_path/ --rm -it ImageName Args`.
3. Если требуется запустить контейнер без интерактивного режима, но с использованием файлов действия или экспорта из QP в Docker, то в качестве аргумента приложения следует передать ключ `--disablePipedInput`. В этом случае приложение не будет ожидать `piped input` или перенаправления файла, а будет использовать ключи `-p|--path` или `-c|--config`.

Пример запуска программы:

```
docker run -v "local_path_to_xml_folder:/xml/" qpcms/qp-console:8.4.4.1 -r "http://config-service" -j "jwt-токен" -m xml -s --disablePipedInput -p "/xml/" "customer_code"
```

Примечание: список доступных образов доступен по адресу.

7.6.2.3 Использование программы через CLI

Программа предназначена для автоматизации операций по воспроизведению действий. Поддерживается импорт данных из файлов XML и CSV.

Примечание: программа поставляется отдельно от продукта QP. Дистрибутив (содержит исполняемый и конфигурационный файлы) доступен по адресу.

Допускается использование программы для одиночных операций, выполняемых пользователем. В качестве пользовательского интерфейса используется командная строка. Порядок использования:

```
consoledbupdate [OPTIONS]+ <customer_code>
```

Пример запуска программы:

```
consoledbupdate -r "http://config-service" -j "jwt-токен" -m xml -s -p "local_path_to_xml_folder" "customer_code"
```

7.6.3 Использование QP8 Backend API

Выполнение действий из XML-файла осуществляется с помощью вызовов QP8 Backend API.

Для получения возможности работы с QP8 Backend API нужно скомпилировать проект с дополнительными библиотеками.

Для выполнения необходимой операции следует создать экземпляр класса `ReplayService` из пространства имён `Quantumart.QP8.WebMvc.Extensions.Helpers.API`.

7.6.3.1 Конструкторы

```
ReplayService(string customerCode, int userId, XDocument fingerPrintSettings)

ReplayService(string customerCode, int userId, XDocument fingerPrintSettings, HashSet<string> identityInsertOptions)

ReplayService(string customerCode, int userId)
```

Параметры:

Название	Описание
customerCode	Значение Customer Code на сервере-получателе.
userId	Идентификатор пользователя, от имени которого должно быть выполнены действия. **Примечание:** можно использовать `userId = 1` (admin).
fingerPrintSettings	Правила вычисления отпечатка БД (содержимое в тэге `fingerprint`, включая сам тэг; содержится в XML-файле с действиями). **Примечание:** если значение не задано или равно `null`, то при попытке вычислить отпечаток БД будет сгенерировано исключение.
identityInsertOptions	Коды сущностей, для которых нужно использовать принудительную вставку идентификаторов. **Примечание:** допустимые значения кодов определены в `Quantumart.QP8.Constants.EntityTypeCode`.

7.6.3.2 Метод ComputeFingerPrint

```
public string ComputeFingerPrint()
```

Вычисляет отпечаток БД.

Результат следует сравнить со значением отпечатка из XML-файла с действиями (атрибут `fingerprint` тэга `actions`) и, если значения не совпадут, то не выполнять действия.

7.6.3.3 Метод ReplayXml

```
public void ReplayXml(string xmlText)
```

Выполняет действия из XML-файла.

Содержимое файла в виде текста передаётся, как параметр метода. Выполнение действий осуществляется в рамках одной транзакции: если в процессе выполнения возникает исключение (как на уровне SQL, так и на уровне .NET), то отменяются все произведённые с начала транзакции изменения в БД.

7.6.4 Лог операций воспроизведения действий

Лог пишется в таблицу `XML_DB_UPDATE_ACTIONS` БД, с которой работает продукт. Структура таблицы:

Название	Тип	Описание
Id	int	Идентификатор операции воспроизведения действия
UpdateId	int	Идентификатор операции запуска воспроизведения. **Примечание:** ссылка на таблицу XML_DB_UPDATE.
Ids	nvarchar	Идентификаторы сущностей, для которых выполнялось действие (например, идентификатор статьи)
ParentId	int	Идентификатор родительской сущности (например, идентификатор контента, к которому относится статья)
Hash	nvarchar	Хэш действия
Applied	datetime	Дата воспроизведения действия
UserId	int	Идентификатор пользователя, выполнившего операцию. **Примечание:** в случае использования программы QuantumArt DbUpdate for QP8 значение равно 1.
SourceXml	nvarchar	Исходное действие из файла
ResultXml	nvarchar	Преобразованное действие (с учётом замены идентификаторов)

7.6.5 Лог операций запуска воспроизведения

Также ведётся лог операций запуска воспроизведения. Используется таблица `XML_DB_UPDATE` в БД, с которой работает продукт. Структура таблицы:

Название	Тип	Описание
Id	int	Идентификатор операции запуска воспроизведения
Applied	datetime	Дата воспроизведения
Hash	nvarchar	Хэш операции запуска воспроизведения
FileName	nvarchar	XML-файл или директория с XML-файлами (при использовании программы QuantumArt DbUpdate for QP8)
UserId	int	Идентификатор пользователя, выполнившего операцию. **Примечание:** в случае использования программы QuantumArt DbUpdate for QP8 значение равно 1.
Body	nvarchar	Содержимое XML-файлов, использованных в операции
Version	nvarchar	Поле не используется текущей версией продукта

7.7 Аудит

Раздел «Аудит» содержит различные служебные данные об использовании продукта пользователями. ГПИ доступен из контекстного меню корневого элемента в дереве сущностей.

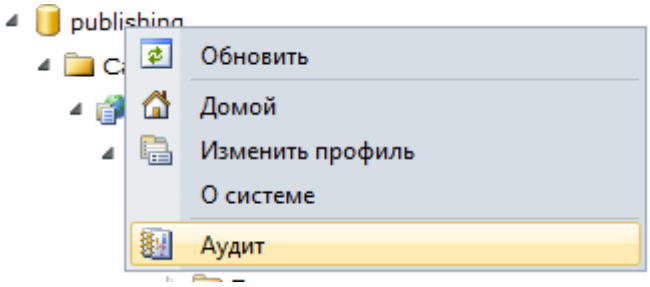


Рисунок 7.8. Аудит в контекстном меню.

Реализованы следующие представления аудита:

- Журнал действий (Actions Log),
- Пользовательские сессии (User Sessions),
- Неудачные попытки входа (Failed Logins).

Примечание: также существуют представления «QP7: Button Trace» и «QP7: Removed Entities». Они являются устаревшими.

7.7.1 Представление «Журнал действий»

Журнал действий содержит записи о всех произведённых в бэкенде значимых действиях над всеми сущностями. Журнал хранит данные о самом событии, затронутой сущности, а также IP и роли пользователя, выполнившего запрос.

Примечание: значимыми считаются действия, приводящие к изменению данных (создание, изменение, удаление).

Код клиента "qa_d...site_rus" - Аудит

Код клиента "qa_demosite_rus"

Обновить

Выберите фильтр

Найти Очистить

Имя действия	Тип действия	Тип сущности	ID сущности	Заголовок сущности	Родительский ID	API	Время	Логин	IP пользователя	Группы пользователя
Множественное удаление статей	Множественное удаление	Статья	741546	Тест	30756		22.04.2024 13:29:45	admin	[illegible]	Administrators, Everyone
Новая статья	Добавить	Статья	741546	Тест	30756		22.04.2024 13:29:27	admin	[illegible]	Administrators, Everyone
Удаление статьи	Удалить	Статья	741545	Тест	30756		22.04.2024 13:28:58	admin	[illegible]	Administrators, Everyone
Изменить настройки базы данных	Обновить	Код клиента	1		0		22.04.2024 13:28:49	admin	[illegible]	Administrators, Everyone
Новая статья	Добавить	Статья	741545	Тест	30756		22.04.2024 13:28:44	admin	[illegible]	Administrators, Everyone

Рисунок 7.9. Представление «Журнал действий».

7.7.2 Представление «Пользовательские сессии»

Представление содержит данные о периодах работы пользователей с бэкендом.

Код клиента "[illegible]"

Обновить

Логин	Время открытия	Время закрытия	Длительность	Авто. вход	IP клиента	Браузер	Имя сервера	QP7
[illegible]	24.07.2017 12:28:44				[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	SPBDEV01	
[illegible]	24.07.2017 10:53:42	24.07.2017 10:53:47	00:00:04		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:54.0) Gecko/20100101 Firefox/54.0	SPBDEV01	
[illegible]	24.07.2017 10:39:33				[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:54.0) Gecko/20100101 Firefox/54.0	SPBDEV01	
[illegible]	24.07.2017 10:25:01				[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	SPBDEV01	
[illegible]	24.07.2017 10:19:50	24.07.2017 10:24:48	00:04:58		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	SPBDEV01	
[illegible]	21.07.2017 18:16:04				[illegible]	websrv02 Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/42.0.2311.152 Safari/537.36	BACKEND03	

Рисунок 7.10. Представление «Пользовательские сессии».

7.7.3 Представление «Неудачные попытки входа»

Представление содержит сведения о зафиксированных событиях неудачной аутентификации в бэкенде.

Код клиента "[illegible]"

Обновить

Логин	Время	Авто. вход	IP клиента	Браузер	Имя сервера	QP7
[illegible]	21.07.2017 17:55:21		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	BACKEND03	
[illegible]	21.07.2017 17:08:09		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	BACKEND03	
[illegible]	20.07.2017 16:04:56		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	BACKEND03	
[illegible]	19.07.2017 14:13:27		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	BACKEND03	
[illegible]	19.07.2017 14:13:22		[illegible]	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36	BACKEND03	

Рисунок 7.11. Представление «Неудачные попытки входа».

7.8 Управление учётными записями пользователей

7.8.1 Пользователь

Учётная запись пользователя используется для аутентификации и авторизации персоны, работающей в бэкенде. Управление пользователями осуществляется в разделе «Пользователи» бэкенда.

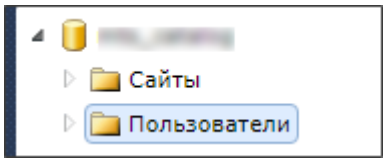


Рисунок 7.12. Раздел «Пользователи» в бэкенде.

Доступ к форме создания пользователя осуществляется одним из следующих способов:

- кнопка «Добавить нового пользователя» в разделе «Пользователи»,
- пункт «Новый пользователь» в контекстном меню раздела «Пользователи».

7.8.1.1 Свойства пользователя

Параметры входа (Login Parameters)

Название	Описание
Логин (Login)	Уникальный (в пределах БД) идентификатор пользователя. Используется для доступа в бэкенд.
Пароль (Password)	Набор символов, используемый для авторизации пользователя в бэкенде.
NT-логин (NT-login)	Идентификатор пользователя. Используется режимом единой авторизации для привязки пользователя QP к пользователю Windows.
Пользователь должен изменить пароль при следующей авторизации (User must change password at next logon)	Указатель, что при следующем входе, пользователь должен будет изменить пароль. Требования к паролю: • длина пароля от 7 до 20 символов; • пароль должен содержать три из четырех категорий: цифры, латинские прописные буквы, латинские строчные буквы, не алфавитно-цифровой символ.
Автоматический вход (Auto Login)	Указатель, что для пользователя включен режим единой авторизации с AD.
Заблокирован (Disabled)	Указатель, что пользователь деактивирован (нет прав на работу в бэкенде). **Примечание:** по умолчанию пользователь деактивирован.

Данные профиля (Profile data)

Название	Описание
Имя (First Name)	Персональные данные пользователя. Используются в ГПИ бэкенда при выводе данных, связанных с пользователем (например, данные о последнем изменении сущности).
Фамилия (Last Name)	
E-mail	Адрес электронной почты пользователя.
Язык (Language)	Язык для ГПИ бэкенда. **Внимание:** ОС на веб-сервере и на компьютере пользователя должны поддерживать выбранный язык.

Членство (Membership)

Название	Описание
Группы (Groups)	Группы пользователей, в которые входит пользователь.

Фильтрация по умолчанию (Default Filters)

Название	Описание
Сайт (Site)	Выбор сайта, в котором требуется применять правила фильтрации по умолчанию.
Контент (Content)	Выбор контента, в котором требуется применять правила фильтрации по умолчанию.

Статьи (Articles)	Выбор статей, в которых требуется применять правила фильтрации по умолчанию.

Настройки отображения (View settings)

Название	Описание
Разрешить группировку контентов в дереве бэкенда (Enable content grouping in backend tree)	Указатель, требуется ли использовать группы контентов в дереве сущностей.

Параметры OnScreen (OnScreen Parameters)

Внимание: режим OnScreen существует только в предыдущих версиях продукта.

Название	Описание
Разрешить для объектов (Allow for objects)	**Внимание:** опция поддерживается только ASP-сборкой в целях совместимости. Указатель, имеет ли пользователь право работать в режиме OnScreen с объектами QP.
Разрешить для полей (Allow for fields)	Указатель, имеет ли пользователь право работать в режиме OnScreen с содержимым полей.

7.8.2 Группа пользователей

QP позволяет Администратору объединять пользователей в группы для удобства управления правами доступа, когда нескольким пользователям необходимо назначить одни и те же права, и чтобы упростить решение задачи по добавлению нового пользователя, которому нужно дать такие же права, как и существующему. Один пользователь может быть добавлен во множество групп. При этом пользователь всегда наследует самый высокий уровень прав на конкретное действие среди всех групп, в которые он добавлен.

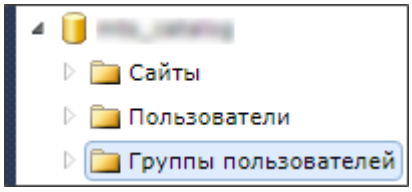


Рисунок 7.13. Раздел «Группы пользователей» в бэкенде.

Управление пользователями осуществляется в разделе «Группы пользователей» бэкенда. Доступ к форме создания группы пользователей осуществляется одним из следующих способов:

- кнопка «Добавить новую группу пользователей» в разделе «Группы пользователей»,
- пункт «Новая группа» в контекстном меню раздела «Группы пользователей».

Примечание: пользователь, входящий в группу пользователей «Администраторы», обладает полным доступом на все функциональные возможности бэкенда и правами на работу со всеми данными, содержащимися в бэкенде.

7.8.2.1 Свойства группы пользователей

Название	Описание
Имя (Name)	Имя группы пользователей.
Описание (Description)	Описание группы пользователей.
Общее владение (Shared Ownership)	Указатель, требуется ли предоставлять всем пользователям из группы полные права доступа к

	сущности, созданной любым из входящих в группу пользователем. По умолчанию выключено (полные права доступа на сущность имеет только пользователь, её создавший).
Параллельный Workflow (Parallel Workflow)	Указатель, требуется ли пользователям группы использовать параллельный Workflow.
Члены группы могут разблокировать сущности (Group members can unlock entities)	Указатель, определяющий, имеют ли пользователи право разблокировать сущности, заблокированные другими пользователями.
Члены группы могут управлять задачами по расписанию (Group members can manage scheduled tasks)	Указатель, определяющий, имеют ли пользователи право управлять задачами по расписанию. Подробнее см. Руководство разработчика
Пользователи (Users)	Список пользователей, включённых в группу пользователей.
Родительская группа (Parent Group)	Группа, от которой требуется наследовать права пользователей.

7.8.3 Связывание пользователя и группы пользователей

Привязку пользователя к группе и группы к пользователю можно выполнить из форм для создания/изменения:

- пользователя,
- группы пользователей.

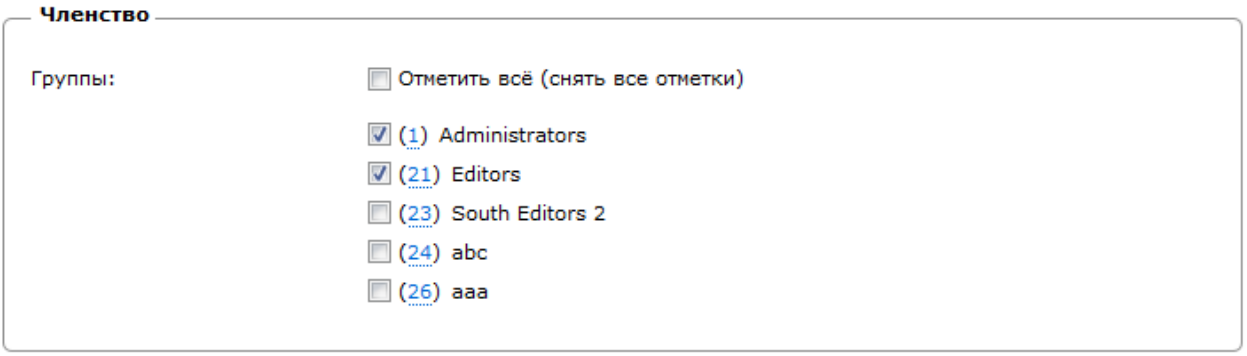


Рисунок 7.14. Управление группами пользователя.

В дереве сущностей бэкенда группы отображаются в соответствии с их иерархией.

7.8.4 Рекомендуемая последовательность при работе с пользователями

Рекомендуемая последовательность действий:

1) создать новую группу пользователей по образцу,
2) удалить из новой группы всех пользователей,
3) настроить права доступа для группы,
4) создать новых пользователей,
5) добавить новых пользователей в новую группу.

7.9 Импорт пользователей из AD

Функциональность предназначена для:

1) импорта новых пользователей в QP из AD,
2) обновления данных существующих пользователей QP актуальными данными из AD.

Выполняется синхронизация для всех `Customer Code`, зарегистрированных в QP. Исключение может быть задано с помощью параметра `ExceptCustomerCodes`. Для отмены синхронизации с помощью ГПИ бэкенда на странице «Настройки» («Settings») следует отключить опцию «Включить синхронизацию пользователей с внешней системой» («Turn on users synchronization with external system»).

7.9.1 Настройка синхронизации

Внимание: пользователь, под которым запущен бэкенд QP, должен обладать достаточными правами для выполнения LDAP-запросов к AD.

- Подробнее об интеграции с AD – см. Интеграция с AD.
- Параметры конфигурации – см. Конфигурационный файл веб-сайта (секция Ldap).

7.9.2 Правила синхронизации

1. Для группы пользователей QP в качестве значения поля `NT-группа` должно быть задано имя группы в AD.
2. Для пользователя QP в качестве значения поля `NT-логин` должно быть задано имя пользователя в AD.
3. При работе службы в первую очередь выполняется сопоставление иерархии групп в QP и AD. В случае обнаружения несовпадений в лог-файле указываются:
    - группы, которые существуют в QP, но отсутствуют в AD;
    - группы с несопоставимой иерархией.

   Такие группы не участвуют в синхронизации. Служба выполняет синхронизацию пользователей, но не групп. Поэтому необходимо заранее настроить группы в QP, чтобы они соответствовали группам AD.
4. Новые пользователи (есть в AD, нет в QP) создаются, у них задаются значения параметров профиля:
    - логин (часть значения `NT-login` от последнего символа `\`),
    - `NT-логин`,
    - E-mail,
    - Имя,
    - Фамилия.

   Также выполняются следующие операции:
    - генерация нового пароля (QP не получает пароли пользователей из AD),
    - установка флага «Автоматический вход».

   Если какие-либо необходимые для QP параметры в AD не заданы, то в это поле в QP вставляется значение `Undefined`.

   Примечание: если пользователь в AD находится в состоянии `disabled`, то он не копируется в QP.
5. Для существующих пользователей (есть в AD, есть в QP) обновляются значения параметров профиля:

- E-mail,
- Имя,
- Фамилия.

Изменение пароля не выполняется.

Если какие-либо необходимые для QP параметры в AD не заданы, то в это поле в QP вставляется значение `Undefined`.

Примечание: если пользователь в AD находится в состоянии `disabled`, то параметры переносятся в QP и для пользователя QP выставляется флаг «Блокировать».

Также выполняется синхронизация списка групп, в которые входит пользователь.

6. Для отсутствующих в AD пользователей (нет в AD, есть в QP) выставляется флаг «Блокировать». При этом выполняется проверка, в какие еще группы QP входит пользователь. По группам, которые содержат значение `NT-группа`, проверяются группы AD. Но если при этой проверке обнаруживается, что пользователь имеет статус `disabled` в AD, то выставляется флаг «Блокировать» в QP. И только если пользователь входит еще куда-либо и не имеет статуса `disabled` в AD, то для него не выставляется флаг «Блокировать».

7.10 Настройка прав доступа

7.10.1 Уровни доступа

Существуют следующие уровни доступа:

Название	Описание
Полный доступ (Full Access)	Разрешены все действия: создание, чтение, изменение и удаление.
Изменение (Modify)	Чтение и изменение разрешены. Создание, удаление запрещены.
Чтение (Read)	Чтение разрешено. Создание, изменение и удаление запрещены.
Список (List)	Разрешено чтение детального списка сущностей. Доступ к форме изменения сущности запрещён.
Запретить (Deny)	Любой доступ запрещён.

7.10.2 Настройка доступа к сущностям

Права доступа к сущностям представляют из себя дочерние типы сущностей по отношению к типам, к которым они предоставляют доступ.

7.10.2.1 Настройка прав доступа к самой сущности

Настройка прав доступа поддерживается для следующих типов сущностей:

- «Сайт»,
- «Контент»,
- «Статья»,
- «Workflow»,
- «Библиотека сайта».

ГПИ для управления правами доступа вызывается из контекстного меню для сущности (раздел «Права доступа»):

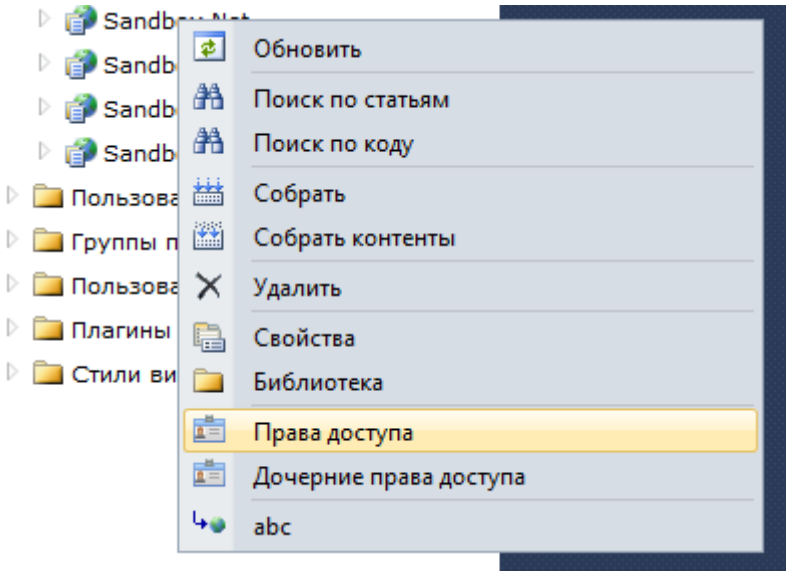


Рисунок 7.15. Права доступа в контекстном меню.

Каждая запись для права доступа содержит следующие данные:

1) кому предоставляется доступ (пользователь или группа);
2) на что предоставляется доступ (одна из указанных ранее сущностей; не задаётся в явном виде, определяется контекстом вызова);
3) какой уровень доступа предоставляется.

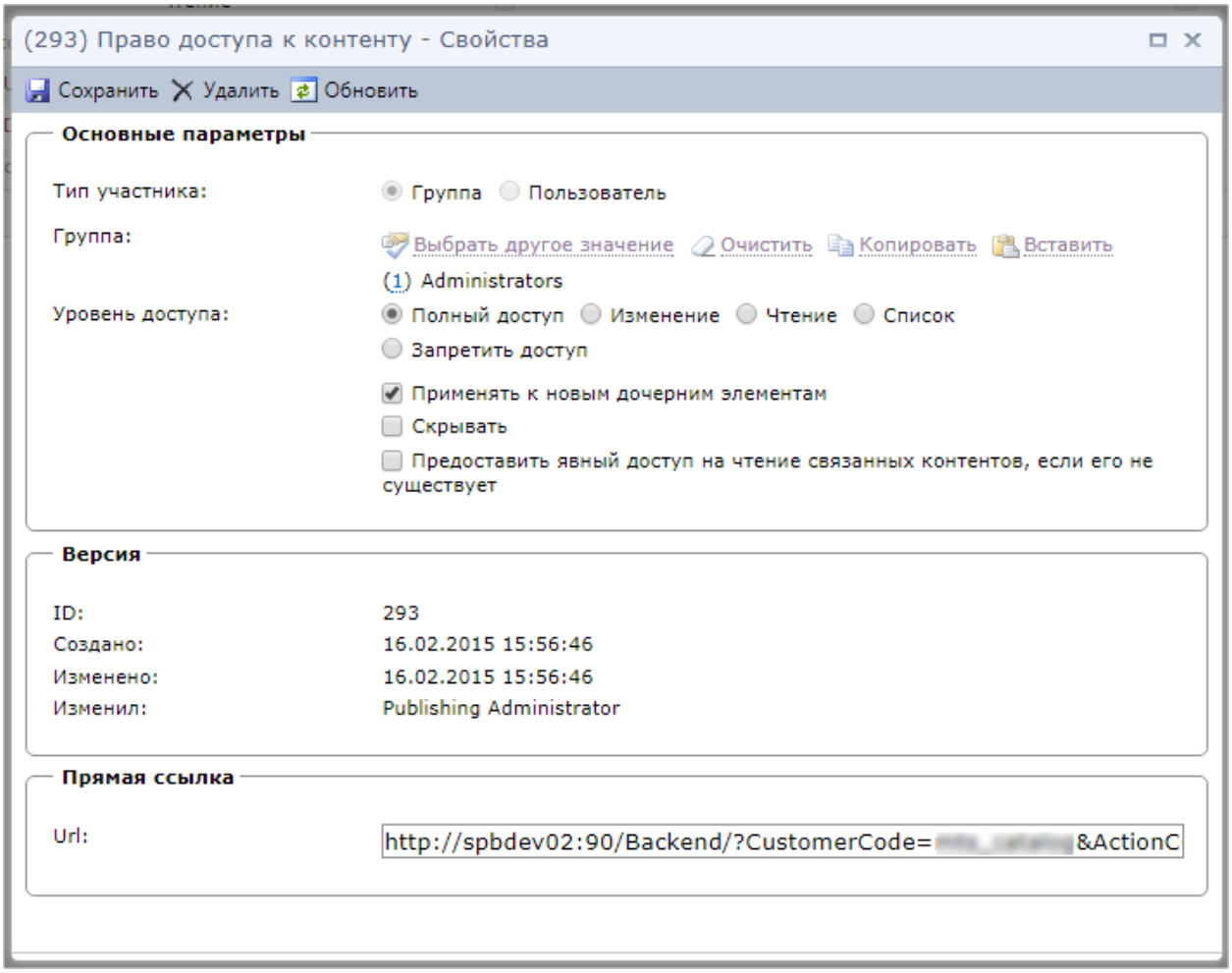


Рисунок 7.16. Экран «Права доступа к контенту».

При настройке прав доступа к контентам и сайтам можно задать опцию «Применять к новым дочерним элементам». Если она установлена, то при создании:

- нового контента будет скопировано право доступа сайта, содержащего контент;
- новой статьи будет скопировано право доступа контента, содержащего статью.

Опция «Скрывать» позволяет скрыть контент в дереве контентов. При этом права на контент и возможность его редактирования по связям остаются.

7.10.2.2 Настройка прав доступа к дочерним сущностям

ГПИ явной настройки прав доступа к сущности плохо подходит для массовых изменений. Для решения подобной задачи следует использовать ГПИ настройки прав доступа к дочерним сущностям. Для сайта он позволяет управлять правами доступа сразу ко всем его контентам, а для контента – ко всем его статьям. ГПИ вызывается из контекстного меню родительской сущности (раздел «Дочерние права доступа»):

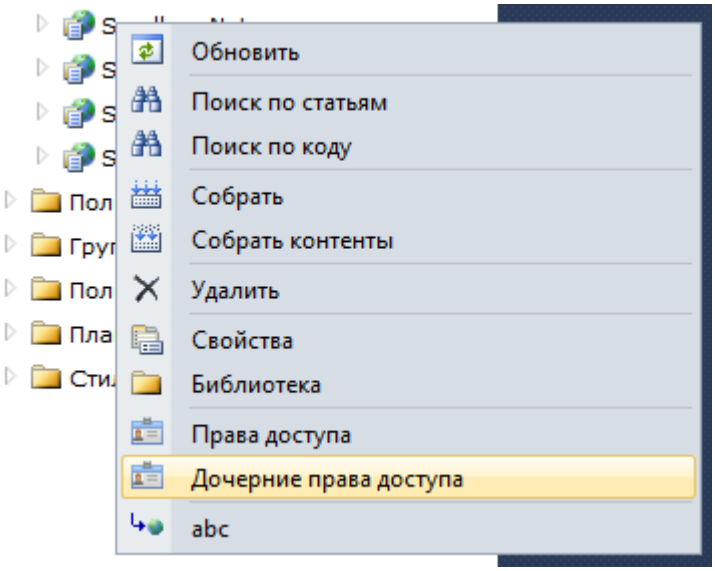


Рисунок 7.17. Дочерние права доступа в контекстном меню.

В первую очередь следует выбрать субъекта (пользователя или группу пользователей), для которого требуется настроить права.

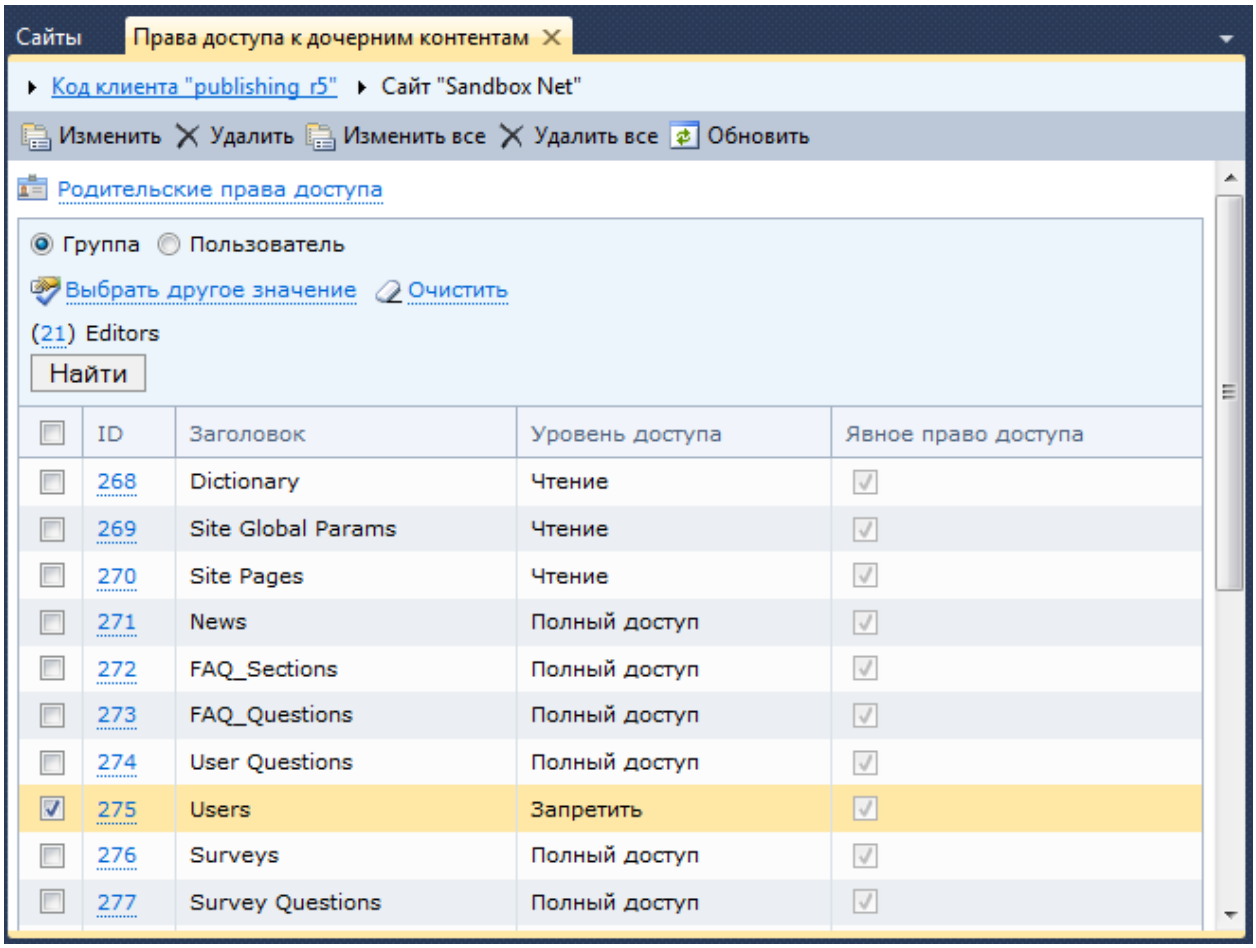


Рисунок 7.18. Выбор субъекта.

После выбора в ГПИ выводятся следующие данные:

- имена сущностей (контентов или статей);
- уровень доступа, заданный для выбранного на каждую сущность;
- указатель, является ли право доступа явным для субъекта (или же оно унаследовано от родительской группы);

  Примечание: явные права доступа имеют более высокий приоритет, чем унаследованные.

- значение опции «Применять к новым дочерним элементам»,
- значение флага «Скрывать».

ГПИ позволяет выполнять изменение и удаление прав доступа. Эти операции могут выполняться:

- над одним элементом (из контекстного меню, на панели инструментов);
- над несколькими выбранными элементами сущности (на панели инструментов),
- над всеми дочерними элементам (кнопки «Изменить все» и «Удалить все» на панели инструментов).

Операция изменения прав доступа выполняется в отношении явных прав доступа:

- изменяет существующие явные права доступа;
- создаёт новые явные права доступа с требуемым уровнем доступа, если они не существуют.

При изменении доступны следующие возможности:

- скопировать право доступа из родительского элемента и применить его к выбранным элементам (опция «Копировать родительское право доступа»);
- задать уровень доступа самостоятельно с возможностью:
  - использовать опцию «Применять к новым дочерним элементам»),
  - выставить флаг «Скрывать»,
  - использовать опцию «Предоставить явный доступ на чтение связанных контентов, если его не существует».

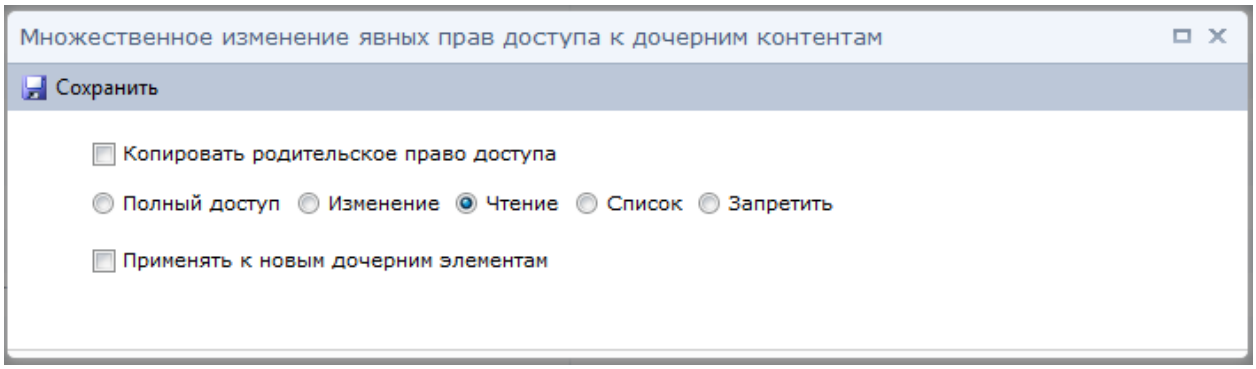


Рисунок 7.19. Множественное изменение явных прав доступа.

Операция удаления прав доступа выполняется в отношении явных прав доступа:

- удаляет явные права доступа, если они существуют;
- не производит удаления, если существуют только неявные права доступа.

Кнопка «Родительские права доступа» позволяет перейти к ГПИ для настройки прав доступа для родительской сущности (сайта или контента).

Опция «Предоставить явный доступ на чтение связанных контентов, если его не существует»

Опция позволяет упростить настройку прав доступа для контентов, использующих поля «Связь».

Если Администратору требуется дать права доступа группе пользователей на изменение статей контента, то для этого ему необходимо удостовериться, что у группы есть, как минимум, право доступа на чтение всех связанных контентов (если такого права нет, то дать его). После включения опции описанная задача будет выполняться автоматически.

7.10.2.3 Применение прав доступа к сущностям

Применение прав доступа к сущностям может осуществляться на разных этапах работы приложения:

Название	Описание
Генерация нового документа и определение доступных кнопок на панели инструментов	Применяется, если на данном этапе чётко определена сущность, к которой относится действие. **Примечание:** например, действие «Удалить» в панели инструментов для формы изменения статьи.
Изменение выбранного элемента и определение доступных кнопок на панели инструментов и пунктов контекстного меню	Применяется, если действие относится к одному выбранному элементу. **Примечание:** например, действие «Удалить» из контекстного меню списка статей.

Непосредственное выполнение действия	Применяется, если действие относится к нескольким выбранным элементам и в качестве повторной проверки в остальных случаях. **Примечание:** например, действие «Удалить» из панели инструментов списка статей (множественное удаление).

7.10.2.4 Вычисление эффективных прав доступа пользователя на сущность

1. Если пользователь входит в группу «Администраторы», то он получает уровень доступа «Полный доступ».
2. Если ищутся права для статьи, а для контента отключена опция «Использовать права доступа статей», то используются права, установленные для контента.

   Примечание. Если для полей контента настроено использование прав доступа на связи (описано далее), то проверяются все права доступа для связи:

   1) находятся все связанные статьи для полей с активным свойством «Использовать связанные права доступа»;

   2) для каждой связанной статьи вычисляется эффективное право доступа.

   Результатом вычисления является минимальное право доступа из всех прав доступа связанных статей.
3. Ищутся явные права для данного пользователя.
4. Если ничего не найдено, то ищутся явные права для групп, в которые входит пользователь. Если права доступа найдены сразу для нескольких групп, то учитывается максимальный уровень доступа.
5. Если ничего не найдено, то выполняется движение вверх по иерархии групп (при этом ранее рассмотренные группы не учитываются). Процесс заканчивается при нахождении результата (явных прав доступа) либо при достижении корня по всем цепочкам поиска.
6. Если на одном из трёх предыдущих шагов найдены явные права доступа, то они принимаются за результат вычисления. Если ничего не найдено, то пользователь получает уровень доступа «Запретить».

   Примечание. При вычислении прав доступа на директории сайта, если для некоторой директории не найдено ни одного результата, при этом у данной директории имеется родительская директория, то вычисление прав доступа выполняется рекурсивно для родителя. Выход из рекурсии осуществляется одним из следующих способов:

   - при нахождении первого подходящего результата он используется в качестве итогового результата вычисления;
   - если достигнута корневая директория Библиотеки сайта и не найдено ни одного результата, то пользователь получает уровень доступа «Запретить».

Если требуется определить, может ли пользователь выполнить какое-либо действие, то итоговый результат проверки прав доступа также зависит от прав доступа на действия и является пересечением прав доступа на сущности и прав доступа на действия.

7.10.3 Настройка доступа к действиям

Настройка доступа к действиям вызывается из контекстного меню корневого узла дерева сущностей (раздел «Список прав доступа»):

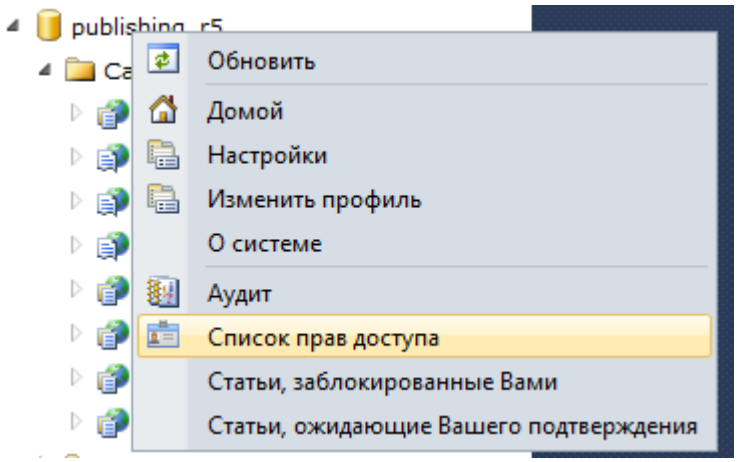


Рисунок 7.20. Настройка доступа к действиям в контекстном меню.

В первую очередь следует выбрать субъекта (пользователя или группу пользователей), для которого требуется настроить права.

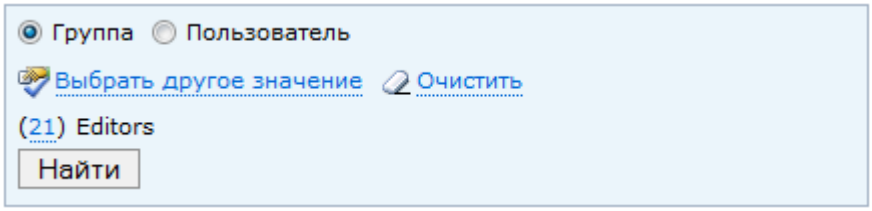


Рисунок 7.21. Выбор субъекта.

После выбора субъекта в ГПИ выводится двухуровневое дерево. Первый уровень содержит типы сущностей, второй уровень – возможные действия над этими типами сущностей. Уровни введены для более удобной настройки. Конечная цель настройки – действия, но часто бывает необходимо задать единый уровень доступа для группы логически связанных действий. В качестве такой группы выступает тип сущности.

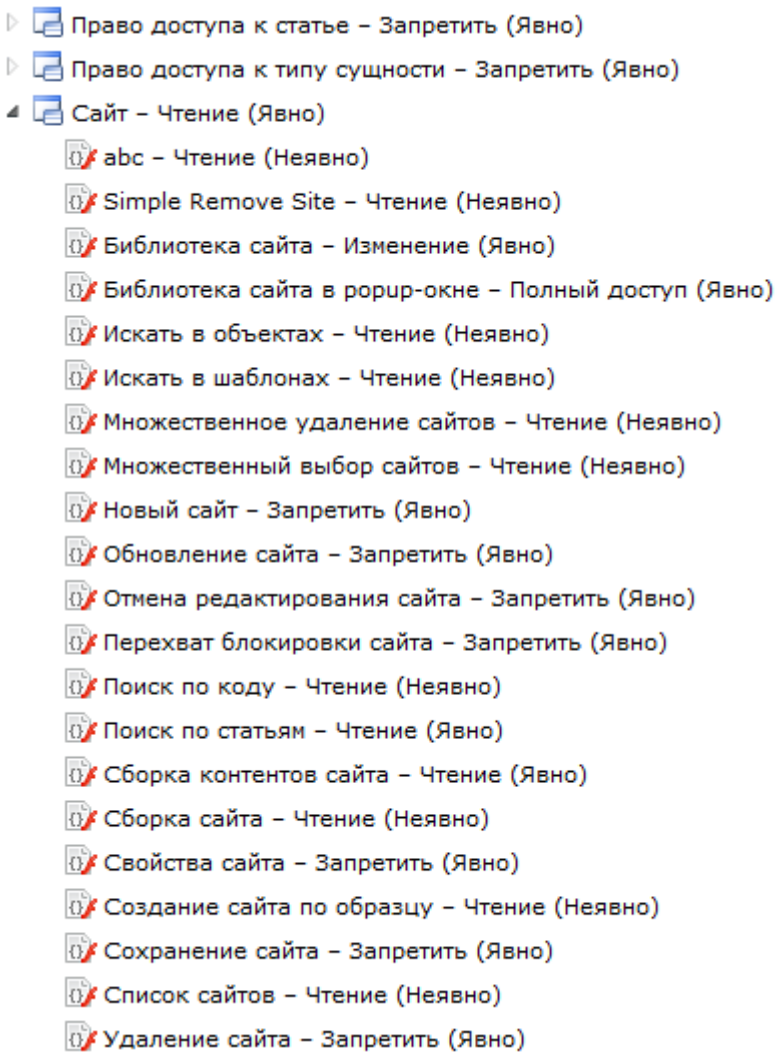


Рисунок 7.22. Дерево доступа к действиям.

Пример типовой задачи: пользователь должен иметь право доступа «Чтение» на все действия с контентами. При такой настройке все действия, которые требуют право доступа «Чтение» или «Список» будут ему доступны, а те действия, которые требует право доступа «Изменение» или «Полный доступ», будут недоступны.

При вычислении прав доступа на действие учитывается иерархия: сначала проверяется наличие явных прав доступа на действие и, если таковые не найдены, то проверяется наличие прав доступа для родительского типа сущности.

7.10.3.1 Настройка прав доступа к действиям через дерево

Для каждого узла дерева выводятся следующие данные:

- заголовок;
- уровень доступа для пользователя (группы);
- указатель, является ли этот уровень доступа явным.

Изменить уровень доступа непосредственно в дереве можно с помощью вызова контекстного меню на узле дерева.

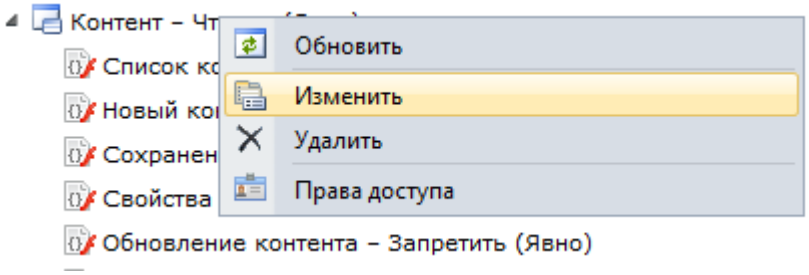


Рисунок 7.23. Контекстное меню для узла дерева доступа к действиям.

Аналогично настройке прав доступа к дочерним сущностям, действие изменения позволяет:

1) изменить явное право доступа, если оно существует;
2) создать явное право доступа в случае его отсутствия.

Действие удаления позволяет удалить явное право доступа, если оно существует.

7.10.3.2 Настройка прав доступа к действиям через список

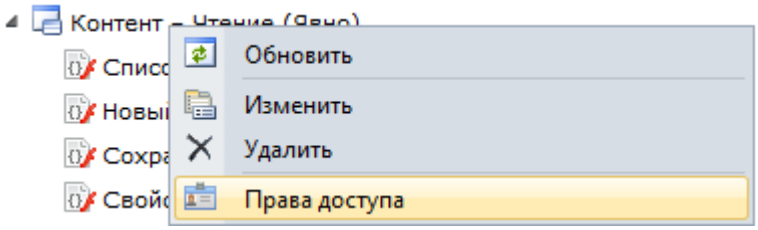


Рисунок 7.24. Настройка прав доступа к действиям через список.

При выборе пункта контекстного меню «Права доступа» на любом узле дерева показывается список всех прав доступа, определённых для данного узла с типовым интерфейсом настройки.

Таким образом, при настройке прав доступа через дерево или список работа ведётся с различными срезами одних и тех же данных:

- при настройке через дерево доступны все права, определённые для субъекта;
- при настройке через список доступны все права, определенные для действия или типа сущностей.

7.10.3.3 Применение прав доступа к действиям

Права на действия работают следующим образом – при выводе контекстного меню или панели инструментов:

- определяется, какие действия входят в состав меню или панели;
- для каждого действия определяется требуемый уровень доступа (для встроенных действий уровень определён разработчиками QP, для пользовательских действий уровень определён разработчиками этих действий);
- для каждого действия определяется, какой эффективный уровень доступа к данному действию имеет пользователь.

Если уровень доступа пользователя меньше запрашиваемого, то действие пользователю недоступно.

Кроме того, такая проверка выполняется для запрашиваемого действия при попытке его выполнить.

7.10.4 Права доступа на связи

Права доступа на связи являются облегчённой версией прав доступа на статьи.

Для контентов с большим объёмом данных проявляются следующие недостатки прав доступа на статьи:

- существенно замедляется работа с контентом,
- настройка прав доступа для каждой статьи занимает у Администратора много времени.

Права доступа на связи позволяют избежать указанных недостатков. Решение основано на том, что обычно нужно дать права на часть контента по определённому признаку. Данная функциональная возможность позволяет использовать в качестве такого признака значение поля связи с некоторой категорией. Включение функциональной возможности для такого поля связи (O2M или M2M) выполняется с использованием свойства «Использовать связанные права доступа».

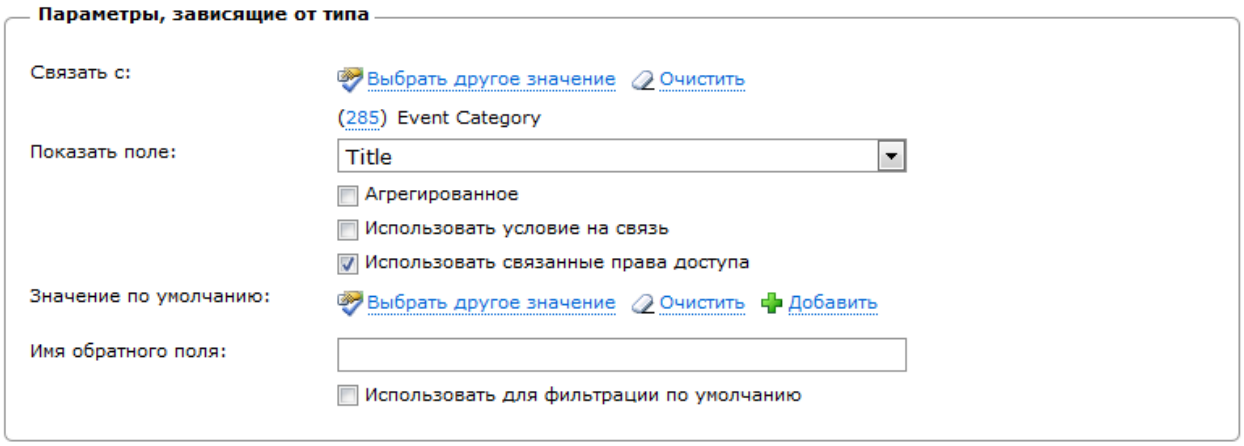


Рисунок 7.25. Включение свойства «Использовать связанные права доступа».

После его включения настройка права доступа ограничится только контентом категорий.

Права доступа на связи применяются:

- в кратких и детальных списках статей (необходим уровень доступа «Список» хотя бы на одну связанную категорию);
- при изменении статьи (необходим уровень доступа «Изменить» на все связанные категории).

При использовании прав доступа на связи совместно с иерархическими контентами может быть полезна опция «Копировать права доступа при создании дочерних элементов». Если опция установлена, то при создании нового дочернего элемента будут скопированы все права доступа от родительского.

7.10.4.1 Фильтрация по умолчанию

Функциональная возможность предназначена для ситуаций, когда не нужно полностью ограничивать доступ Контент-менеджера только определёнными категориями статей (права доступа на связи). Но в то же время у Контент-менеджера есть определённые предпочтительные категории статей, с которыми он работает в контенте чаще всего. В этом случае можно настроить фильтрацию по умолчанию.

Первичная настройка выполняется в ГПИ бэкенда на уровне пользователя Контент-менеджера: выбирается сайт, контент и статьи, задающие требуемые категории.

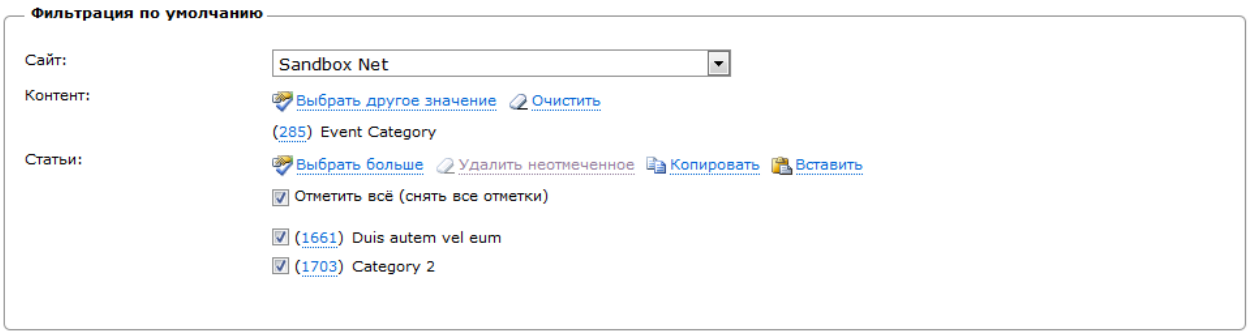


Рисунок 7.26. ГПИ для возможности «Фильтрация по умолчанию».

После этого фильтрацию нужно включить на уровне конкретного поля, так как в контенте может быть несколько подходящих полей. Это делается с помощью свойства «Использовать для фильтрации по умолчанию».

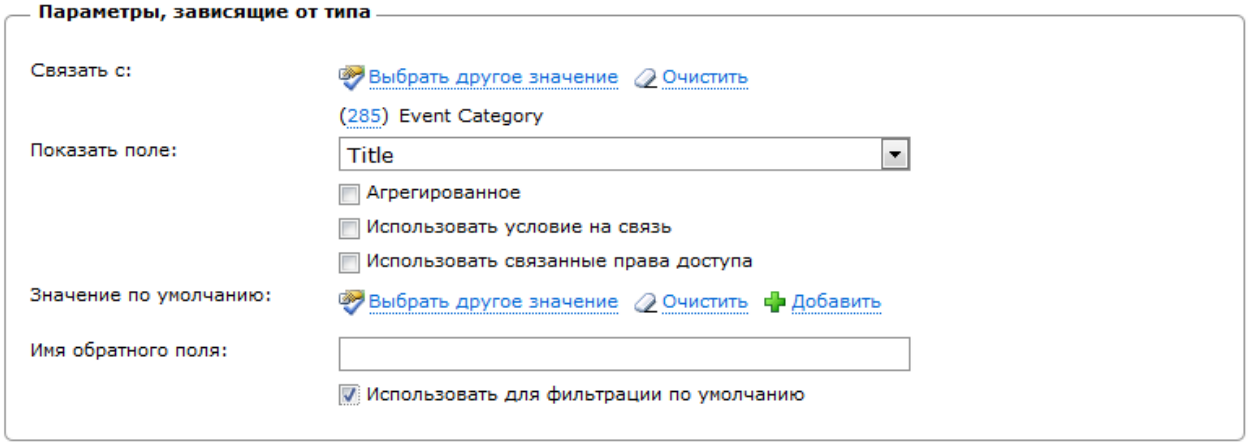


Рисунок 7.27. Свойство «Использовать для фильтрации по умолчанию» для поля.

Фильтрация по умолчанию применяется при открытии списка статей. Параметры фильтрации можно изменить, если развернуть панель поиска/фильтрации, затем удалить/изменить фильтр по умолчанию или добавить другой фильтр. Чтобы вернуть фильтр по умолчанию, следует нажать кнопку «По умолчанию».

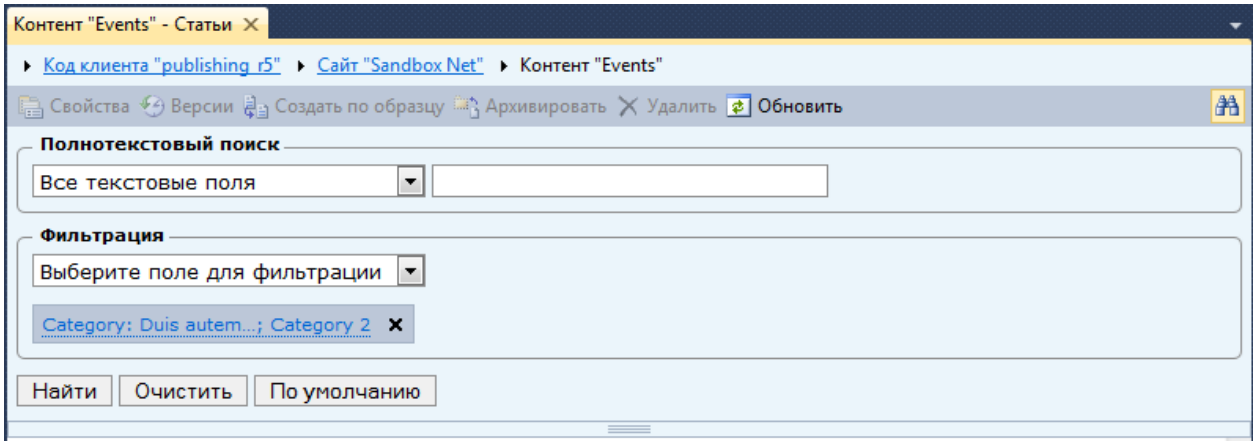


Рисунок 7.28. Результат использования фильтрации по умолчанию.

7.10.5 Права доступа по типу продукта

Примечание: возможность используется в продукте DPC.

Для поля типа «Классификатор» доступна возможность указать, что права доступа на контент-расширение с данными по продукту определённого типа должны определять права на продукт

данного типа. Для включения возможности в свойствах поля требуется активировать флаг для параметра «Использовать права доступа по типам».

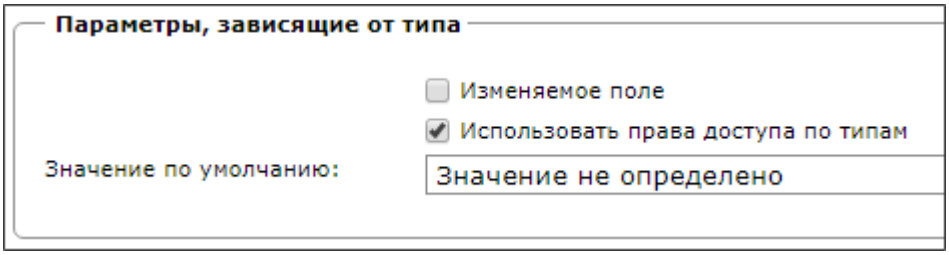


Рисунок 7.29. Параметр «Использовать права доступа по типам».

По умолчанию возможность отключена (права на контент-расширение игнорируются).

8 Задачи по расписанию (Scheduled Tasks)

В QP есть возможность выполнять задачи по расписанию.

Примеры задач по расписанию:

- DisableUsersJob – задача по блокировке пользователей, которые не заходили в QP более определенного периода времени;
- EnableUsersJob – задача по разблокировке пользователей;
- InterfaceCleanupJob – задача по очистке отправленных уведомлений;
- InterfaceNotificationJob – задача по отправке уведомлений;
- UsersSynchronizationJob – задача по синхронизации пользователей с AD;
- CleanSystemFoldersJob – задача по очистке системных папок.

Для просмотра списка задач необходимо нажать на CustomerCode правой кнопкой мыши, а далее выбрать из списка Scheduled Tasks (Рисунок 8.1).

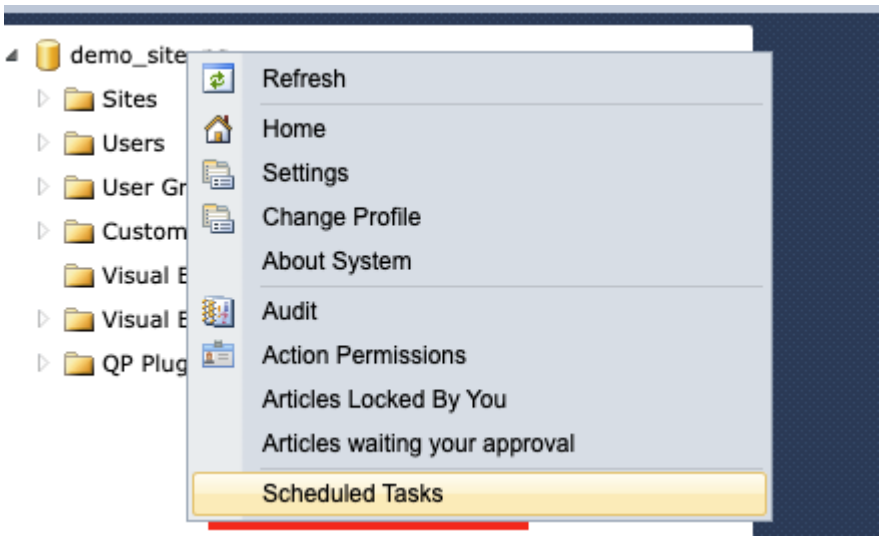


Рисунок 8.1 Задачи по расписанию

На странице перечислены задачи: название задачи, ее описание, расписание, время последнего запуска, время следующего запуска, а также возможность запустить задачу вручную («Run Job») для принудительного запуска задачи вне расписания.

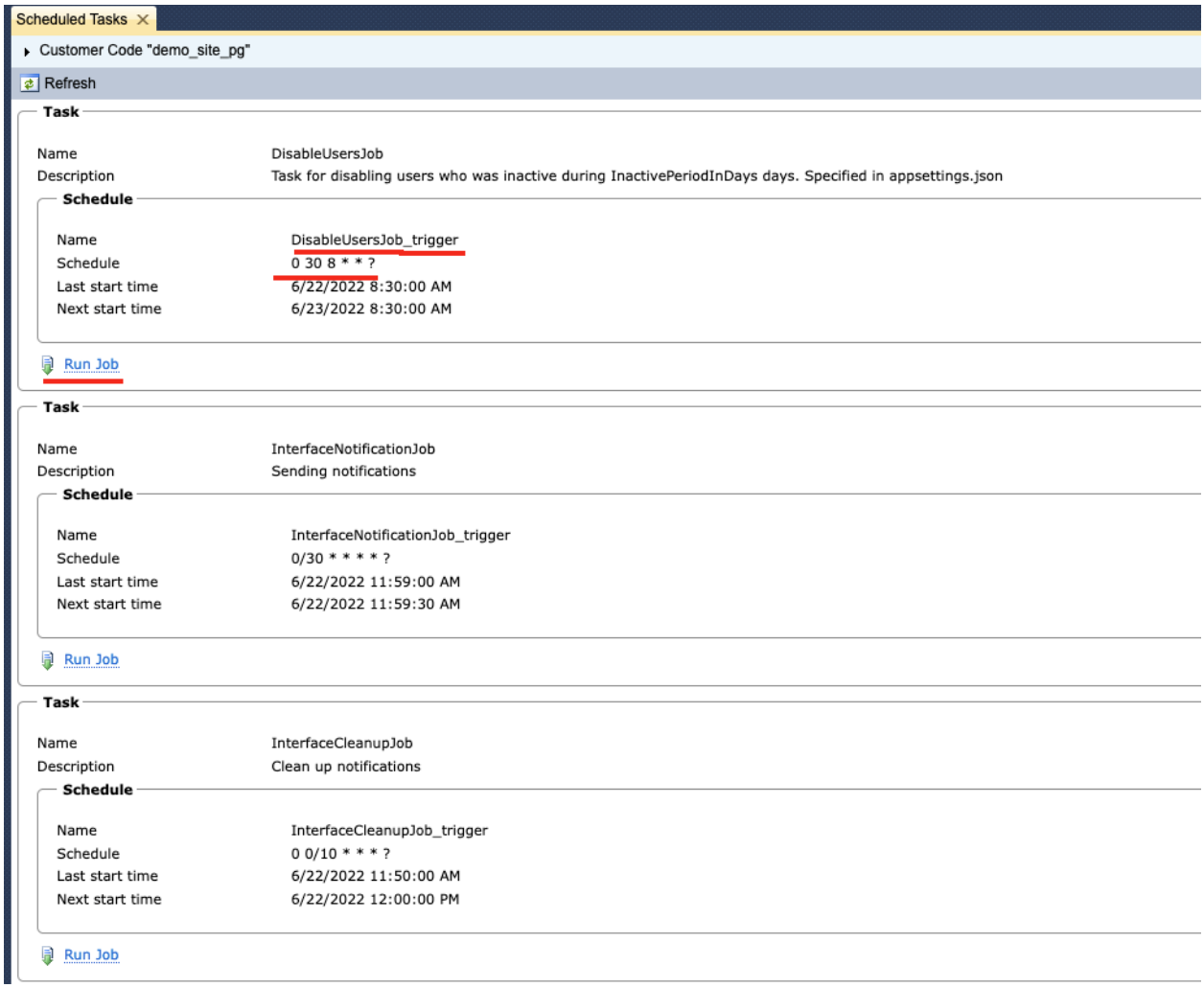


Рисунок 8.2 Список задач по расписанию

Настраиваются задачи в конфигурационном файле QP:

```
"CommonScheduler": {
    "Name" : "CommonScheduler",
    "DefaultUserId": 1,
    "DefaultLanguageId": 1,
    "InterfaceNotificationJob" : {
      "Schedule": "0/30 * * * * ?",
      "Description" : "Sending notifications"
    },
    "InterfaceCleanupJob": {
      "Schedule": "0 0/10 * * * ?",
      "Description" : "Clean up notifications"
    },
    "DisableUsersJob": {
      "Schedule": "0 30 8 * * ?",
      "Description" : "Task for disabling users who was inactive during InactivePeriodInDays
days. Specified in appsettings.json",
      "SpecifiedConditions":
      {
        "InactivePeriodInDays": "90"
      }
    }
  },
```

Предоставляются возможности: добавить новую задачу, удалить задачу. В случае если добавляется принципиально новый вид задачи, необходимо написать обработчик.

Также предоставлена возможность разделять задачи по CustomerCode: есть задачи, которые выполняются только для определенного CustomerCode, а есть которые выполняются для всех CustomerCode.

9 Интеграции с внешними системами

9.1 Интеграция с MinIO

Данная интеграция позволяет хранить статические файлы в S3-совместимом хранилище (MinIO) вместо файловой системы сервера. Это затрагивает следующую функциональность:

- библиотека сайта (см. раздел 4.18.1 Руководства разработчика);
- библиотека контента (см. раздел 4.18.2 Руководства разработчика);
- поля типов "Файл", "Изображение", "Динамическое изображение" (см. раздел 4.7.1 Руководства разработчика).

В дистрибутиве ПО доступны скрипты, позволяющие выполнить миграцию файлов из файловой системы в MinIO.

9.1.1 Миграция

9.1.1.1 Установка MinIO Client

Установка MinIO Client выполняется на ту машину, с которой будут выполняться скрипты миграции. Для работы QP8 данное ПО не требуется.

- В соответствии с официальной документацией, описывающей установку на Windows, необходимо скачать файл клиента MinIO и разместить его в каталоге, доступном в переменной окружения PATH. После этого для проверки можно выполнить команду `mc --help`.
- Создаём алиас для актуального соединения MinIO с помощью команды `mc alias set ALIAS HOSTNAME ACCESS_KEY SECRET_KEY`, где:
  - `ALIAS` – создаваемый алиас, например `minio`
  - `HOSTNAME` – адрес MinIO, например `http://minio01:9000`
  - `ACCESS_KEY` – ключ доступа к MinIO (логин)
  - `SECRET_KEY` – секрет доступа к MinIO (пароль).

  Проверить алиас можно с помощью команды `mc ls minio`.

9.1.1.2 Установка модуля SqlServer

Установка модуля SqlServer для Powershell осуществляется на ту машину, с которой будет выполняться скрипт миграции. Для работы QP8 данное ПО не требуется.

В соответствии с официальной документацией, необходимо установить модуль SqlServer командой `Install-Module -Name SqlServer`. Проверить установленный модуль можно командой `Get-Module SqlServer -ListAvailable`.

9.1.1.3 Загрузка файлов

Загрузка файлов выполняется с той машины, куда ранее были установлены MinIO Client и модуль SqlServer.

Создаём бакет `qp8` с помощью команды `mc mb minio/qp8`. В качестве альтернативы можно создать бакет через MinIO UI.

Для выполнения загрузки файлов необходимо выполнить скрипт `minio.ps1`, задав параметры `src` и `dest`.

Примечание: скрипт `minio.ps1` загружает файлы только в том случае, если файлы отсутствуют по соответствующему пути в MinIO.

9.1.1.4 Исправление регистра файлов

Исправление регистра файлов выполняется с той машины, куда ранее были установлены MinIO Client и модуль SqlServer.

Для исправления регистра файлов необходимо выполнить скрипт `fix_filenames.ps1`, задав параметры `path`, `isMinioPath`, `dbserver`, `dbname`, `dblogin`, `dbpassword`.

В параметре `path` передаётся путь к папке, файлы в которой необходимо обработать. Это может быть путь как в MinIO, так и в файловой системе. Булев параметр `isMinioPath` позволяет задать тип пути. Четыре параметра (`dbserver`, `dbname`, `dblogin` и `dbpassword`), задают конкретную БД, в которой необходимо выполнить корректировку регистра файлов.

Скрипт ищет ссылки на файлы в неправильном регистре и исправляет их в соответствии с тем, как файл называется на самом деле. Для ускорения поиска используется полнотекстовый индекс, поэтому для корректной работы скрипта он должен быть актуальным для БД.

Примечание: данный скрипт можно выполнять как после загрузки файлов в MinIO, так и до неё.

9.1.1.5 Изменение URL

Изменение URL выполняется с той машины, куда ранее были установлены MinIO Client и модуль SqlServer.

Стоит учесть, что при выдаче файлов по HTTP с помощью самого MinIO для использования на сайте URL будет отличаться от того, что есть сейчас, потому что в MinIO в качестве первого сегмента URL используется название бакета. Чтобы обойти это неудобство, рекомендуется для доступа к файлам по HTTP использовать балансировщик (например, nginx) с рерайтом путей. Это позволит обеспечить бесшовный переход с сохранением основного URL.

Однако некоторые URL придётся менять в любом случае. Например, это URL с предыдущей версии сайта. Для того чтобы изменить URL можно использовать скрипт `fix_url.ps1`, приложенный в поставку. Для исправления регистра необходимо перейти в папку со скриптом и выполнить его, задав параметры `src`, `dest`, `dbserver`, `dbname`, `dblogin`, `dbpassword` и `search` (опционально).

- В параметре `src` задаётся заменяемый URL (или его часть).
- В параметре `dest` задаётся URL (или его часть), который мы хотим получить в итоге.
- Параметр `search` используется для ускорения поиска; в нём указывается строка поиска, замена URL будет выполняться только в тех записях таблицы `content_data`, которые соответствуют условиям поиска. Если параметр `search` не задан, то его значение берётся из параметра `src`.

Четыре параметра (`dbserver`, `dbname`, `dblogin` и `dbpassword`), задают конкретную БД, в которой необходимо выполнить замену URL.

9.1.2 Настройка QP8 на использование MinIO

Глобально доступ к использованию MinIO настраивается в `appsettings.json` в отдельной секции `S3` (см. Конфигурационный файл веб-сайта).

Далее на всех базах, которые должны работать с новым QP8, необходимо выполнить скрипт `current.sql`, поставляющийся вместе с QP8. Это необходимо для поддержки новых настроек в UI.

После этого каждый кастомер-код необходимо переключить по отдельности через UI на странице «Настройки» (Settings) (Рисунок 7.5, Рисунок 9.1):

Рисунок 9.1. Прочие настройки

Затем следует перенастроить пути на MinIO в свойствах сайта (Рисунок 9.2, Рисунок 9.3):

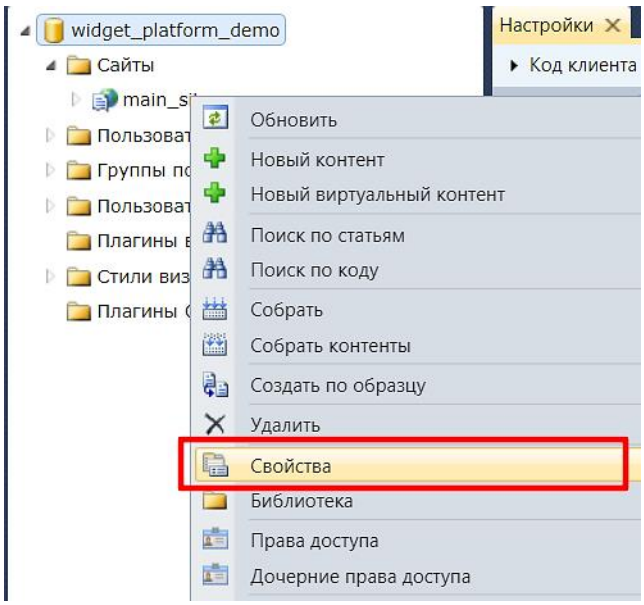


Рисунок 9.2. Контекстное меню сайта – Свойства

Примечание: подробнее о свойствах сайта – см. раздел 4.4.1 Руководства разработчика.

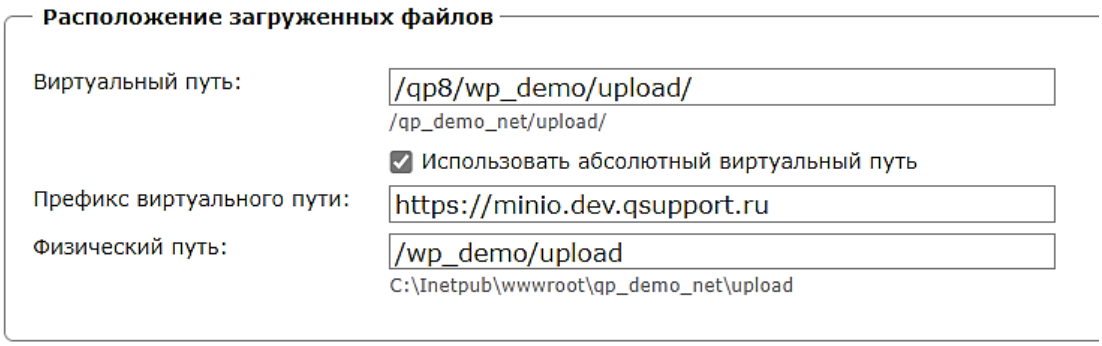


Рисунок 9.3. Расположение загруженных файлов

Далее при необходимости (если переключение между старым и новым storage непрозрачное с точки зрения URL) необходимо изменить URL-адреса в БД (см. выше).

9.2 Интеграция с Camunda

Интеграция QP8.CMS с платформой Camunda позволяет автоматизировать ряд процессов и управлять задачами пользователя.

Примечание: подробное описание работы платформы Camunda, а также инструкции по ее использованию, можно найти здесь.

Для активации интеграции QP8.CMS с платформой Camunda требуется произвести ряд настроек как в файле конфигурации QP, так и в ГПИ QP.

Настройка конфигурации заключается в добавлении новой секции параметров «Camunda» в файл конфигурации `appsettings.json` (см. Конфигурационный файл веб-сайта).

Дальнейшие настройки выполняются через пользовательский интерфейс QP.

9.2.1 Настройки в ГПИ QP

Для корректной интеграции с платформой Camunda необходимо последовательно выполнить все описанные ниже действия.

9.2.1.1 Создание контентов

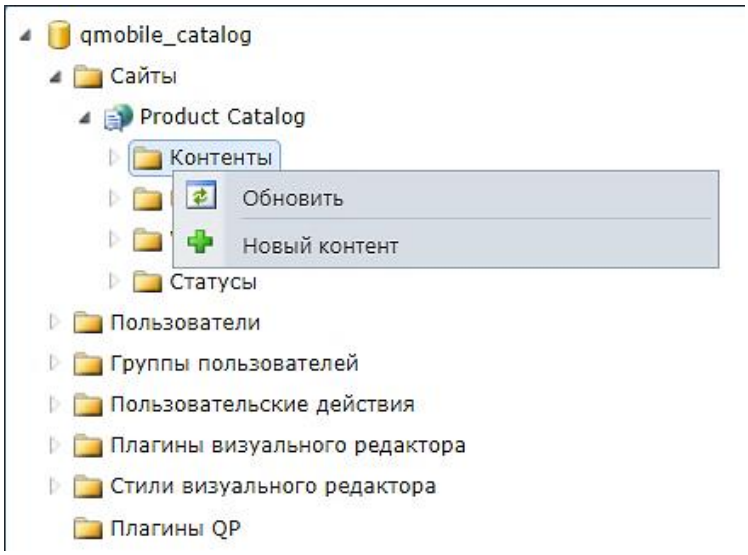


Рисунок 9.4. Создание нового контента через контекстное меню

В QP8.CMS необходимо создать три контента (Рисунок 9.4) со следующим набором полей:

Контент		Название поля	Тип поля	Описание поля
1.	**Схемы workflow** Содержит схемы Camunda. Название может быть задано произвольно.	Title	Текстовое	Понятное для пользователя название схемы
		SchemaFile	Файл	Файл схемы. Название может быть задано произвольно.
		SchemaId	Строка	Идентификатор схемы, с которым схема будет загружена в Camunda. Название поля обязательное.
		IsMainSchema	Булевое	Признак главной схемы. Если процесс состоит из нескольких схем, то этот признак проставляется как `true` у основной схемы; у связанных он проставляется как `false`. Если процесс состоит из одной схемы, то у неё этот параметр должен соответствовать `true`. Название поля обязательное.

2.	**Workflow** Процессы. Название может быть задано произвольно.	Title	Текстовое	Понятное для пользователя название процесса
		WorkflowID	Текстовое	Идентификатор процесса, с которым процесс будет опубликован в Camunda. Название обязательное; значение должно совпадать с ID процесса, указанным в схеме.
		WorkflowSchemas	Связь «многие ко многим»	Связь на контент «Схемы workflow». Позволяет привязать к процессу одну или несколько схем. К процессу должна быть привязана *одна* схема со значением `true` в поле IsMainSchema. Название обязательное.
3.	**Назначения workflow** Привязка workflow к контенту, для статей которого будет запускаться workflow. Название может быть задано произвольно.	Title	Текстовое	Понятное для пользователя название назначения
		ContentId	Связь «один ко многим»	Связь на контент «Contents». Позволяет указать, для какого контента будет запущен процесс. Название обязательное.
		Workflow	Связь «один ко многим»	Связь на контент «Workflow». Позволяет указать, какой процесс следует запускать в Camunda. Название обязательное.

Внимание! Если в контенте присутствует поле «Классификатор» (Classifier), то при запуске workflow контент будет определяться по значению данного поля (а не по Id самого контента). Классификатор позволяет, например, привязать разные workflow к разным типам одного продукта.

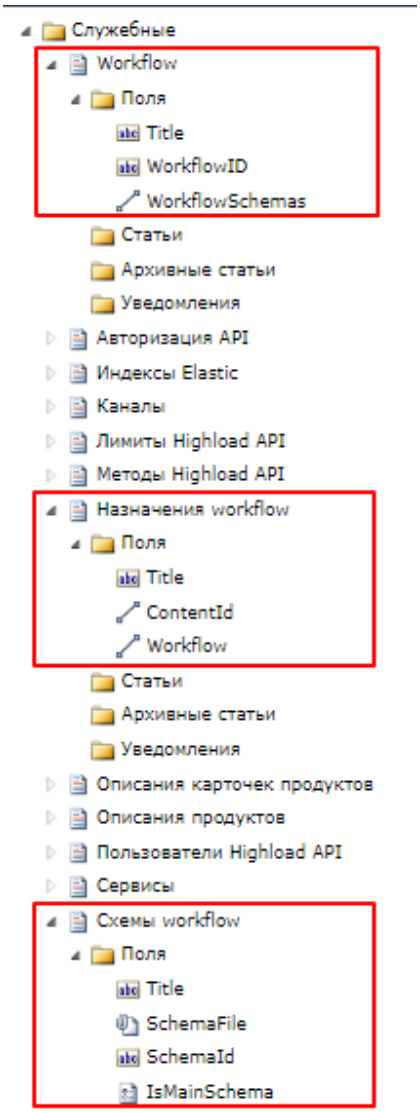


Рисунок 9.5. Контенты, необходимые для привязки workflow, и их поля

9.2.1.2 Настройка кода клиента (Customer code)

Далее следует перейти в настройки Customer Code (Рисунок 9.6).

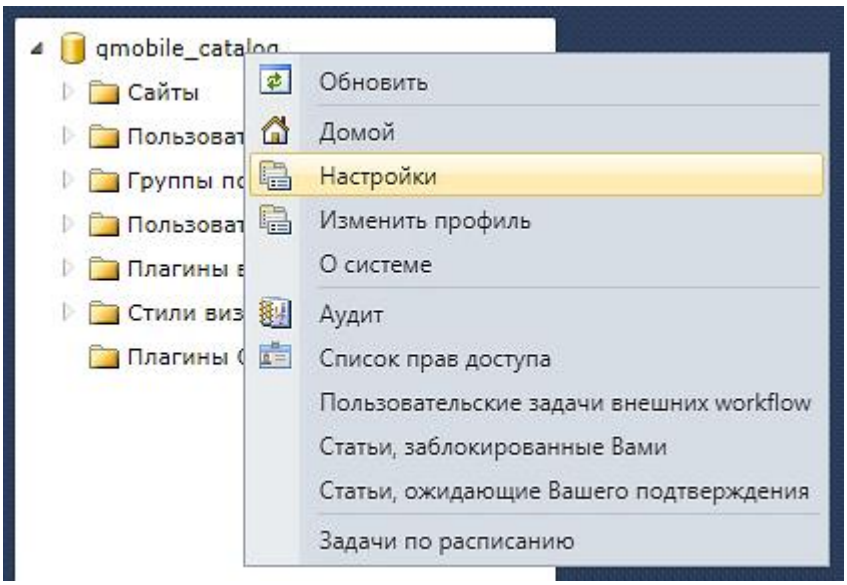


Рисунок 9.6. Переход к настройкам Customer Code через контекстное меню

В секции «Прочие настройки» в списке «Настройки приложения» необходимо добавить следующие настройки и задать им соответствующие значения (Рисунок 9.7):

1. `EXTERNAL_WORKFLOW` – `true` (для включения интеграции с Camunda на уровне текущего кода клиента);
2. `EXTERNAL_WORKFLOW_CONTENT_ID` – Id контента «Назначения workflow»;
3. `EXTERNAL_WORKFLOW_SCHEMA_FILE_FIELD_ID` – Id поля с файлом схемы контента «Схемы workflow»;
4. `EXTERNAL_WORKFLOW_SCHEMA_NAME_FIELD_ID` – Id поля Title контента «Схемы workflow».

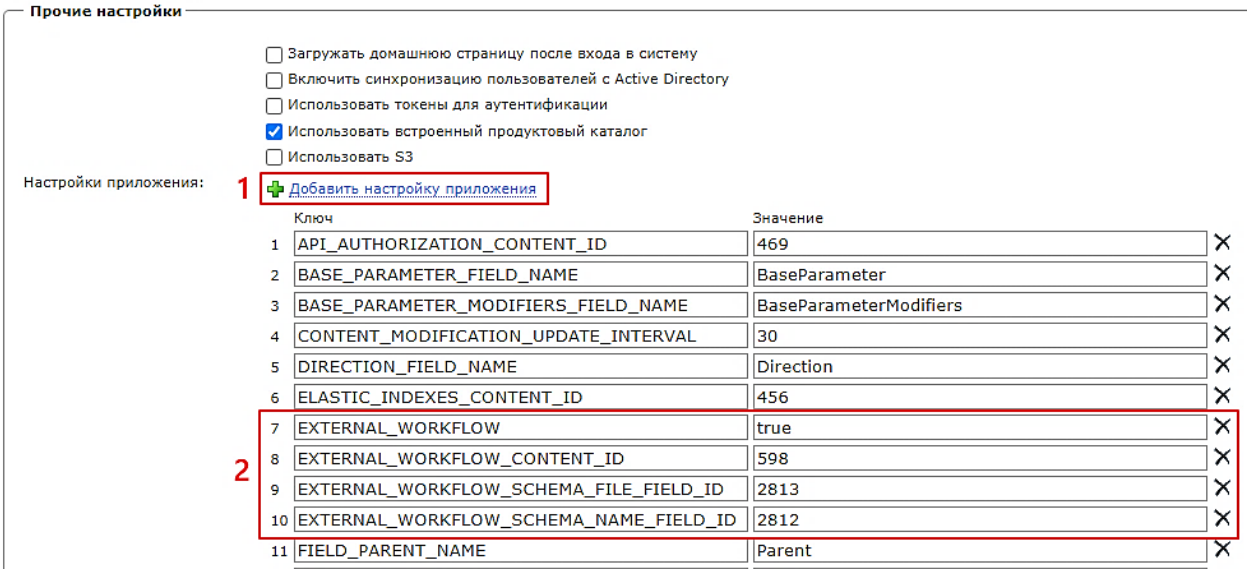


Рисунок 9.7. Добавление настроек приложения (1) и пример заполнения настроек внешнего workflow (2)

9.2.1.3 Создание пользовательских действий

Далее необходимо создать пользовательские действия публикации процесса и запуска процесса (Рисунок 9.8).

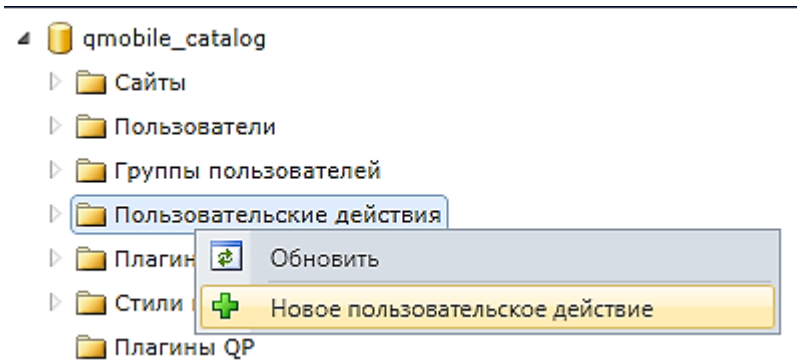


Рисунок 9.8. Создание пользовательского действия через контекстное меню

Параметры двух созданных действий должны быть заданы согласно следующим требованиям:

Параметр	Действие публикации процесса	Действие запуска процесса
Имя	Понятное пользователю название, например «Опубликовать workflow»	Понятное пользователю название, например «Запустить workflow»
Псевдоним	Понятный алиас процесса, например «PublishWorkflow»	Понятный алиас процесса, например «StartWorkflow»

Описание	Произвольно, по желанию	
Тип действия	Экспорт	
Тип сущности	Статья	
Режим выбора сайтов	Скрывать на всех сайтах, кроме выбранных	
Выбранные сайты	«Выбрать больше» > в списке сайтов выбираем нужный сайт	
Режим выбора контентов	Скрывать во всех контентах, кроме выбранных	
Выбранные контенты	«Выбрать больше» > в списке контентов выбираем контент «Workflow»	«Выбрать больше» > в списке контентов выбираем контент(-ы), для которого(-ых) будет запускаться процесс **Внимание:** при выборе всех имеющихся контентов кнопка запуска появится у каждого из них, однако при её нажатии будет выводиться сообщение о том, что на контент не назначен workflow.
URL	`https://qp/ExternalWorkflow/publishWorkflow`, где qp – это адрес QP	`https://qp/ExternalWorkflow/startWorkflow`, где qp – это адрес QP
Url иконки	Название иконки (если используются внутренние) или адрес внешней, например `green_arrow_up.png` для стрелки вверх	Название иконки (если используются внутренние) или адрес внешней, например `start.gif` для стрелки вправо
Порядок	Желаемый порядковый номер относительно остальных действий статьи, например 99 (чтобы действие располагалось в конце списка)	
Показывать в контекстном меню	По усмотрению (необходимо выбрать хотя бы один из вариантов отображения)	
Показывать на панели инструментов		
Интерфейс	Оставляем не отмеченным (важно!)	
Фраза подтверждения	Текст запроса подтверждения действия (произвольно, по желанию)	
Есть предварительное действие	–	

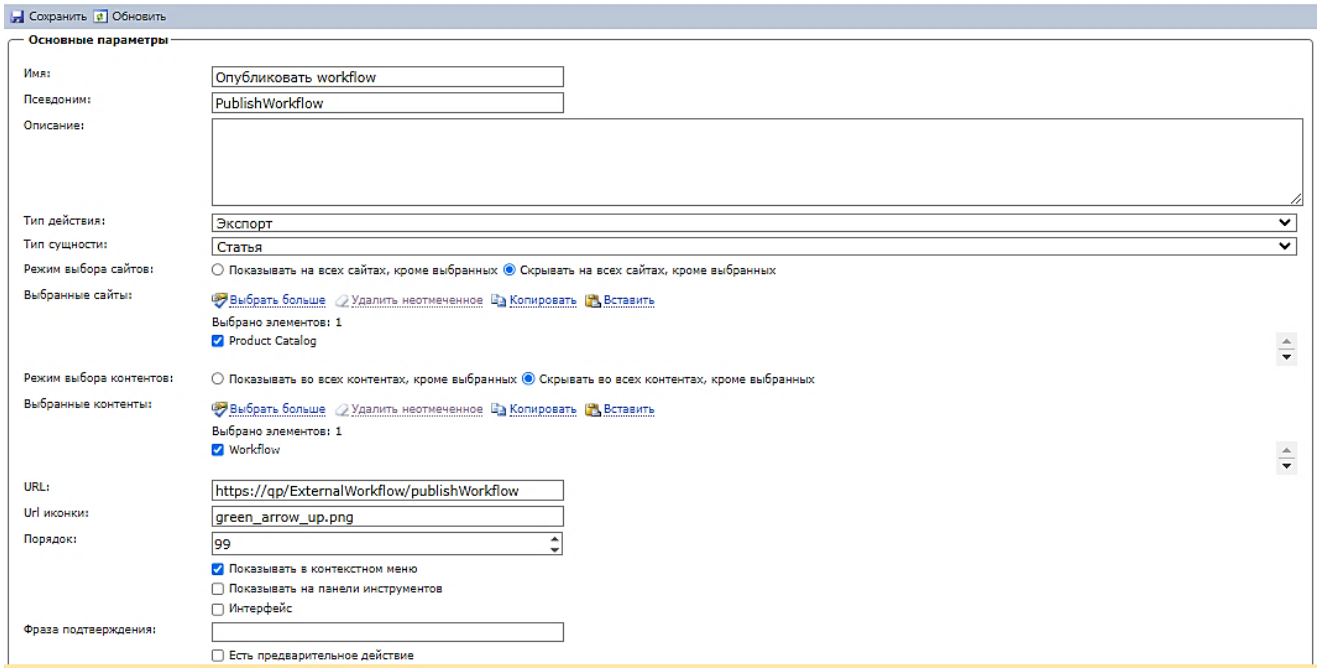


Рисунок 9.9. Добавление пользовательского действия (на примере действия публикации процесса)

После заполнения параметров требуется сохранить внесенные изменения.

9.2.1.4 Создание виртуальных контентов

Далее следует создать виртуальные контенты для просмотра состояния запущенных процессов и их истории (Рисунок 9.10).

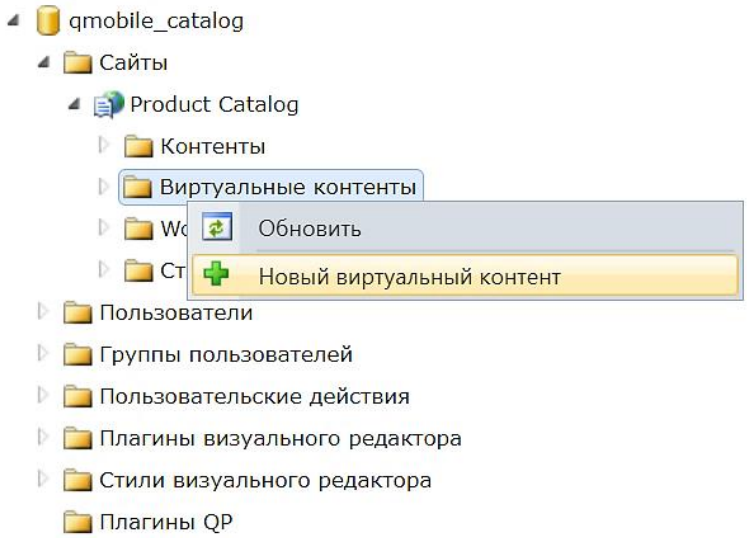


Рисунок 9.10. Создание нового виртуального контента через контекстное меню

1. Создаём новый виртуальный контент с именем «Все процессы», и типом «User Query». В поле «Запрос» вводим:

```
select distinct on (w.id)
w.id as content_item_id,
w.process_id as "id процесса",
w.created,
w.created_by as "кем запущен",
w.article_name as "статья",
w.article_id as "id статьи",
w.workflow_name as "название процесса",
```

```
w.workflow_id as "id статьи процесса",
s.status as "статус процесса",
s.created as modified,
s.created_by as "кем обновлён статус",
cast(125 as numeric(18, 0)) as status_type_id,
cast(1 as numeric(18, 0)) as visible,
cast(0 as numeric(18, 0)) as archive,
cast(1 as numeric(18, 0)) as last_modified_by
from external_workflow w
join external_workflow_status s on s.external_workflow_id = w.id
order by w.id, s.id desc
```

Примечание: остальные параметры виртуального контента заполняются произвольно.

После заполнения параметров требуется сохранить внесенные изменения.

2. Создаём новый виртуальный контент с именем «Запущенные процессы» и типом «User Query». В поле «Запрос» вводим:

```
select w.id as content_item_id,
w.article_id as "id статьи",
w.process_id as "id процесса",
w.created as created,
w.created_by as "кем запущен",
w.article_name as "статья",
w.workflow_name as "название процесса",
w.workflow_id as "id статьи процесса",
s.status as "статус процесса",
s.created as modified,
s.created_by as "кем обновлён статус",
cast(1 as numeric(18, 0)) as last_modified_by,
cast(125 as numeric(18, 0)) as status_type_id,
cast(1 as numeric(18, 0)) as visible,
cast(0 as numeric(18, 0)) as archive
from external_workflow_in_progress p
join external_workflow_status s on p.current_status = s.id
join external_workflow w on p.external_workflow_id = w.id
order by w.created desc
```

Примечание: остальные параметры виртуального контента заполняются произвольно.

После заполнения параметров требуется сохранить внесенные изменения.

3. Создаём новый виртуальный контент с именем «История процессов» и типом «User Query». В поле «Запрос» вводим:

```
select s.id as content_item_id,
w.process_id as "id процесса",
w.created as created,
w.created_by as "кем запущен",
w.article_name as "статья",
w.article_id as "id статьи",
w.workflow_name as "название процесса",
w.workflow_id as "id статьи процесса",
s.status as "статус процесса",
s.created as modified,
s.created_by as "кем обновлён статус",
cast(125 as numeric(18, 0)) as status_type_id,
cast(1 as numeric(18, 0)) as visible,
cast(0 as numeric(18, 0)) as archive,
```

```
cast(1 as numeric(18, 0)) as last_modified_by
from external_workflow w
join external_workflow_status s on s.external_workflow_id = w.id
order by w.id desc, s.id asc
```

Примечание: остальные параметры виртуального контента заполняются произвольно.

После заполнения параметров требуется сохранить внесенные изменения.

4. Далее разворачиваем поля созданных контентов и активируем флаг «Просмотреть в списке» в свойствах тех полей, которые необходимо отображать в списке статей данного контента. Через параметр «Разместить» можно задать порядок расположения полей (Рисунок 9.11).

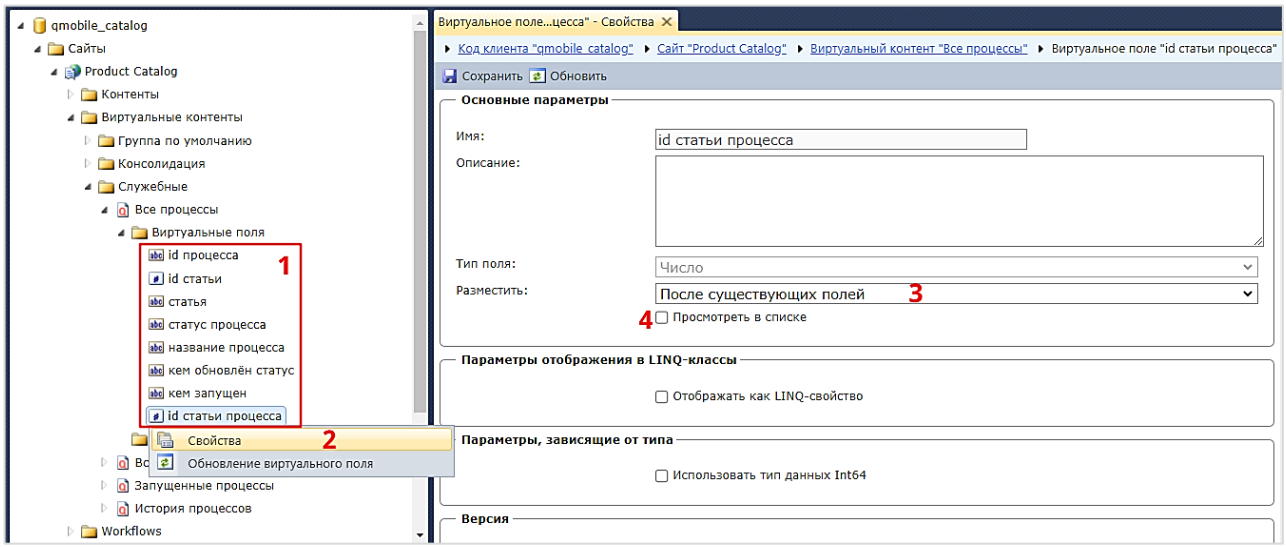


Рисунок 9.11. Поля виртуальных контентов (1), их свойства (2), параметры сортировки (3) и признак отображения (4)

После заполнения параметров требуется сохранить внесенные изменения.

На этом настройка интеграции с Camunda завершена. Процесс создания схем на платформе Camunda описан в Руководстве Разработчика QP8.CMS.

10 Удаление продукта

10.1 Удаление продукта на Windows

Для удаления продукта необходимо выполнить следующие действия:

1) Запустить `UninstallQP8Interactive.cmd`.
2) Задать имя сайта (по умолчанию – *QP8*).
3) Выполнится процедура удаления:
   - удалится сайт *QP8,*
   - удалится связанный с ним пул,
   - удалятся файлы сайта.

10.2 Удаление продукта на Linux

10.2.1 Удаление продукта на Linux (с использованием Docker)

1) Если устанавливался nginx на docker, то перейти в папку `/etc/qpconfig/nginx` и выполнить команду `sudo docker-compose down`.
2) Перейти в папку `/etc/qpconfig/compose` и выполнить команду `sudo docker-compose down.`
3) Очистить неиспользуемые *images* и *volumes* в docker для освобождения ресурсов системы.

10.2.2 Удаление продукта на Linux (без использования Docker)

1) Останавливаем сервисы и убираем их из автозапуска, выполнив команды:

```
systemctl stop qp.service
systemctl disable qp.service
systemctl stop qp-storage.service
systemctl disable qp-storage.service
systemctl stop qp-config.service
systemctl disable qp-config.service
```

2) Удаляем папки приложений:

```
/home/qp/QP
/home/qp/QP.Configuration.Service
/home/qp/QP.Storage
```

3) Удаляем пользователя `qp` и группу `qp.`

10.2.3 Удаление продукта на Linux (с использованием Kubernetes)

Перейти в папку `/etc/qpconfig/kubernetes` и выполнить команды:

```
kubectl delete -f ing.yaml
kubectl delete -f qp.yaml
kubectl delete -f pv.yaml
```

11 Прочее

11.1 Плагин «Яндекс.Спеллер» для визуального редактора

Существует плагин визуального редактора, позволяющий выполнять орфографическую проверку значения поля «Визуальный редактор» с использованием сторонней службы «Яндекс.Спеллер».

Для работы плагина необходимо, чтобы с веб-сервера, на котором расположен бэкенд QP, было разрешено формировать исходящие запросы на `http://speller.yandex.net:80`.

Подключение плагина осуществляется в свойствах сайта или поля (пункт «Yandex Spell Checking» в разделе «Команды»).

12 Партнёрство и сертификация

Спасибо за проявленный интерес к продуктам нашей компании!

Приглашаем вас стать нашим сертифицированным партнёром!

Сертификация проводится по различным направлениям: администратор, разработчик или контент-редактор. Запросы на сертификацию принимаются по адресу: **info@quantumart.ru**