­

Программное обеспечение

Система обогащения данных POLYCORE

Руководство администратора

Аннотация

Настоящий документ является руководством администратора Системы обогащения данных POLYCORE.

Документ разработан в соответствии с требованиями ГОСТ Р 59795-2021 «Требования к содержанию документов».

Содержание

[Введение 5](#_Toc194956531)

[1 Назначение и условия применения 10](#_Toc194956532)

[1.1 Назначение системы 10](#_Toc194956533)

[1.1.1 Функции 10](#_Toc194956534)

[**1.2 Условия применения** 12](#_Toc194956535)

[**1.2.1 Серверная часть** 12](#_Toc194956536)

[1.2.2 Локальная сеть 13](#_Toc194956537)

[2 Подготовка к работе 15](#_Toc194956538)

[2.1 Состав программного обеспечения 15](#_Toc194956539)

[2.1.1 Установка и настройка Docker 15](#_Toc194956540)

[2.1.2 Установка компонентов на один сервер 16](#_Toc194956541)

[2.1.3 Вспомогательный функционал 18](#_Toc194956542)

[2.2 Порядок проверки работоспособности 18](#_Toc194956543)

[2.3 Порядок обновления системы 19](#_Toc194956544)

[3 Описание операций администрирования 21](#_Toc194956545)

[3.1 Определения и сокращения POLYCORE 21](#_Toc194956546)

[3.2 Резервное копирование 22](#_Toc194956547)

[3.2.1 Создание резервной копии 22](#_Toc194956548)

[3.2.2 Восстановление из резервной копии 22](#_Toc194956549)

[3.3 Настройки компонентов системы 22](#_Toc194956550)

[3.3.1 Общие настройки компонентов системы 22](#_Toc194956551)

[3.3.2 Конфигурационные файлы 23](#_Toc194956552)

[3.3.3 Ограничение ресурсов для контейнера 30](#_Toc194956553)

[3.4 Настройка объектов системы 31](#_Toc194956554)

[3.4.1 Провайдеры 32](#_Toc194956555)

[3.4.2 Пользователи 38](#_Toc194956556)

[4 Аварийные ситуации 58](#_Toc194956557)

[5 Рекомендации по освоению 60](#_Toc194956558)

# Введение

Модуль POLYCORE (далее «Система») представляет собой решение для управления нормативно-справочной информацией и ручного обогащения данных во внешних источниках в привычном для бизнес-пользователей интерфейсе электронных таблиц без хранения этих данных внутри.

Уровень подготовки персонала, необходимого для работы с ИАС, предполагает наличие следующих групп пользователей:

* Служба эксплуатации ИАС,
* Администраторы,
* Инженеры,
* Операторы.

Служба эксплуатации ИАС.

В службу эксплуатации ИАС входят специалисты следующих категорий: «Администратор защиты (безопасности) информации», «Администратор операционных систем», «Администратор баз данных».

* Администратор защиты (безопасности) информации обеспечивает:
  + Формирование списка пользователей, допущенных к работе с Системой.
  + Настройку учетных записей пользователей и управление ролями доступа.
  + Формирование матрицы доступа к ресурсам Системы и данным, а также изменение прав доступа.
* Администратор операционных систем отвечает за:
  + Установку компонентов платформы, активацию и первоначальную настройку.
  + Сопровождение ИАС (тестирование работоспособности, восстановление и т.п.), обновление версий (анализ необходимости перехода на новые версии, разработку перечня мероприятий по переводу на новую версию).
* Администратор баз данных отвечает за:
  + Генерацию систем управления базами данных.
  + Сопровождение и управление информационными ресурсами.
  + Сохранение резервных копий, восстановление искаженной информации, архивирование информации и организацию поступления информации из архива.
  + Обработку и анализ статистической информации о характере и интенсивности использования данных, о распределении нагрузки на различные компоненты структуры баз данных, внесение изменений в структуру баз данных в процессе эксплуатации Системы с целью повышения производительности.

Служба эксплуатации обеспечивает функционирование в штатном режиме технических и программных средств АИС, отслеживает процессы наполнения АИС данными.

Поддержка функционирования Системы должна осуществляться силами действующей Службы эксплуатации АИС, состоящей из специалистов, обладающих знаниями в области информационных и сетевых платформ, на которых реализована АИС, и опытом администрирования баз данных.

Администраторы.

Администраторами являются специалисты, которые участвуют в процессах настройки объектов Системы (Моделей, Форм, Тегов), предоставления доступов к Системе и её объектам, проведения анализа работы настроенных экземпляров объектов, выявления аномалий и их причин.

Обязанности:

* Настройка системы, управление пользователями, правами доступа, провайдерами данных.
* Установка, обновление и сопровождение системы.
* Интеграция с внешними источниками данных (PostgreSQL, ODBC, JDBC).

Требования:

* Знания:
  + Глубокое понимание архитектуры и функционала Системы.
  + Знание принципов работы СУБД (PostgreSQL), включая настройку и выполнение SQL-запросов.
  + Понимание ролевой модели доступа (роли: Читатель, Оператор, Редактор, Дизайнер, Администратор).
  + Знание протоколов интеграции (ODBC, JDBC) и настройки провайдеров.
* Навыки:
  + Продвинутая работа с интерфейсом системы.
  + Управление пользователями.
  + Настройка прав доступа к объектам Системы (Модели, Формы, Теги, Провайдеры).
  + Настройка валидации данных и интеграционных процессов.
  + Анализ и оптимизация производительности системы.
* Опыт:
  + Работа с данными (включая опыт роли Оператора).
  + Администрирование баз данных и систем консолидации данных.

Инженеры.

Инженерами являются специалисты, которые участвуют в процессах настройки объектов Системы (Модели, Формы, Теги), проведения анализа работы настроенных экземпляров объектов, выявления аномалий и их причин.

Обязанности:

* Настройка Моделей, Форм, Тегов и Провайдеров.
* Проверка работоспособности объектов системы.
* Реализация схем данных («Звезда», «Снежинка»).

Требования:

* Знания:
  + Общее понимание назначения Системы её компонентов.
  + Концепции организации данных: схемы «Звезда» и «Снежинка».
  + Основы JSON (чтение, извлечение данных).
  + Принципы интеграции с внешними источниками через Провайдеры.
* Навыки:
  + Создание и настройка Форм в табличном интерфейсе (аналогично Excel).
  + Работа с формулами, условным форматированием, справочниками.
  + Настройка расчетных показателей и промежуточных расчетов.
  + Валидация данных и настройка правил форматно-логического контроля.
* Опыт:
  + Свободное владение Excel или аналогами (LibreOffice, Google Sheets).
  + Проектирование структур данных для хранилищ.

Операторы.

Операторами являются специалисты, которые участвуют в процессах ввода данных в объекты типа Формы в Системе.

Обязанности:

* Ввод и редактирование данных в Формах.
* Работа с крупными объемами данных (до сотен тысяч строк).
* Соблюдение правил валидации и форматно-логического контроля.

Требования:

* Знания:
  + Базовое понимание функционала Системы.
  + Правила ввода данных, включая использование справочников и условного форматирования.
* Навыки:
  + Работа с интерфейсом Форм (аналогично Excel).
  + Копирование/вставка данных через буфер обмена.
  + Использование комментариев и истории изменений для рецензирования.
* Опыт:
  + Базовые навыки работы с электронными таблицами (Excel, LibreOffice).
  + Ввод данных в высоконагруженных системах с поддержкой многопользовательского доступа.

Для работы с Системой необходимо ознакомиться со следующим набором эксплуатационной документации:

* Руководство администратора;
* Руководство пользователя.

1. Назначение и условия применения
   1. Назначение системы

Система является решением для управления нормативно-справочной информацией и ручного обогащения данных во внешних источниках в привычном для бизнес-пользователей интерфейсе электронных таблиц без хранения этих данных внутри Системы.

* + 1. Функции

Система позволяет управлять следующей функциональностью:

* + - 1. Управление данными и их обогащение
* Работа с данными в Формах:
  + Просмотр, ввод и редактирование данных в Формах.
  + Поддержка структуры электронных таблиц, включая копирование данных через буфер обмена.
  + Работа с крупными объемами данных (до сотен тысяч строк).
* Настройка Форм:
  + Использование справочников для выбора значений при вводе.
  + Условное форматирование ячеек (цвет, шрифт, правила отображения).
  + Размещение в свободных областях Форм:
    - графиков и диаграмм,
    - справочной информации,
    - вспомогательных расчетов (формулы, скрипты).
* Интеграция и контроль:
  + Сохранение данных во внешние источники через настроенные провайдеры (PostgreSQL, ODBC, JDBC).
  + Экспорт данных из Форм в форматы: .xlsx, .ods, .pdf.
  + Рецензирование данных (комментарии, история изменений).
  + Использование расчетных показателей на основе введенных данных.
* Производительность:
  + Оптимизация для работы с высоконагруженными Формами.
  + Поддержка многопользовательского доступа без потери скорости отклика.
    - 1. Совместное редактирование
* Редактирование одной Формы несколькими пользователями в режиме реального времени.
* Отображение активности пользователей: кто редактирует Форму и какие элементы изменяет.
  + - 1. Форматно-логический контроль
* Настройка правил проверки вводимых данных (типы значений, условия).
* Мгновенная валидация при выходе из ячейки.
* Гибкие сценарии: блокировка неверных данных или уведомления с возможностью подтверждения.
  + - 1. Ролевая модель доступа
* Читатель: просмотр Форм и Тегов.
* Оператор: редактирование данных без изменения структуры Форм.
* Редактор: управление Формами, Тегами и доступом на основе существующих Моделей.
* Дизайнер: создание Моделей, Форм, Тегов, настройка Моделей на основе существующих Провайдеров.
* Администратор: полный контроль над Системой, включая управление пользователями и подключениями к внешним системам.
  + - 1. Базовые сценарии использования

Система предоставляет взаимосвязанные процессы настройки и ввода данных для пользователей, в соответствии со следующими основными сценариями:

* Сценарий использования системы для предварительной настройки:
  + Создание и настройка Моделей данных с использованием схем «Звезда» и «Снежинка».
  + Формирование Форм в виде электронных таблиц, связывание их с источниками данных через Провайдеры.
  + Настройка правил валидации, промежуточных расчетов, Тегов для категоризации Форм.
  + Управление доступом пользователей к объектам системы (Моделям, Формам, Тегам).
* Сценарий использования системы для ввода данных:
  + Поиск Форм через иерархическое дерево Тегов.
  + Ввод и редактирование данных в Формах, включая добавление графиков, справочной информации и расчетных показателей.
  + Сохранение изменений в источниках данных через настроенные Провайдеры.
  1. **Условия применения**

Для функционирования АИС необходимо следующее программно-аппаратное обеспечение:

* + 1. **Серверная часть**

Минимальные требования к серверному оборудованию следующие:

* 8 vCPU (2.8 ГГц+)
* 12GB RAM
* 90GB HDD

Из которых:

* Сервис app:
  + 2 vCPU
  + 4GB RAM
  + 10GB HDD
* Сервис grid:
  + 4 vCPU
  + 4GB RAM
  + 30GB HDD
* Внутренняя база данных системы (redis, postgres):
  + 2 vCPU
  + 4GB RAM
  + 30GB HDD

Ориентировочная формула для подсчета конфигурации в зависимости от количества пользователей: дополнительно к минимальным системным требованиям необходимо RAM 256-512МБ CPU 0.1 vCPU в среднем на каждого пользователя. Конечная конфигурация уточняется в каждом случае отдельно.

Операционная система: Astra Linux Special Edition 1.6 (Воронеж) или аналог.

Права пользователя, разворачивающего приложение: user - non-root with sudo privileges.

Дополнительные требования к установленным приложениям: Docker версии 20.10.0 и до 25, Docker -compose версия 1.29 и выше (для Docker с версии 26 необходимо использовать плагин Docker-compose).

* + 1. Локальная сеть

Все компоненты платформы должны находиться в одной подсети или должна обеспечиваться прозрачная маршрутизация. Не рекомендуется использовать NAT. В рамках ознакомления рекомендуется отключить брандмауэры. Внутри локальной сети между всеми компонентами не должно быть ограничений по передаче данных. Для доступа из внешней сети достаточно открыть порт, используемый POLYCORE (порт задается при установке). При использовании системы с установленными антивирусами или комплексными системами защиты необходимо обеспечить свободную работу, сетевую активность и взаимодействие компонентов.

1. Подготовка к работе
   1. Состав программного обеспечения

Модуль POLYCORE поставляется в виде нескольких файлов:

* образы Docker, содержащие в себе все компоненты с уже настроенным окружением и всеми внутренними зависимостями;
* файлы, поставляемые в дистрибутиве:
  + docker-compose\_prod.yml.tmpl
  + .env.tmpl
  + prod.nginx.conf.tmpl
  + coolwsd.xml.tmpl.

Примечание: Docker — программное обеспечение для автоматизации развёртывания и управления приложениями в среде виртуализации на уровне операционной системы. Суть и одно из предназначений Docker такое же, как и у виртуальных машин — это изоляция работы различных конфликтующих программ внутри одного сервера. Наглядно увидеть отличия между виртуальной машиной и контейнером можно, пройдя по ссылке: https://www.docker.com/what-docker.

Таким образом, установка сводится к двум шагам:

1. Установка docker-engine внутри операционной системы;

2. Настройке файлов, поставляемых в дистрибутиве.

Основные операции с интерфейсом, а также настройка объектов системы описаны в документе «Руководство пользователя».

* + 1. Установка и настройка Docker

1. Установить Docker в соответствии с инструкцией:

<https://docs.docker.com/install/linux/docker-ce/ubuntu/>.

1. Установить Docker Compose в соответствии с инструкцией:

<https://docs.docker.com/compose/install/>.

* + 1. Установка компонентов на один сервер

1. Создать директорию для файлов образов

cd ~ && mkdir install

1. Скопировать из полученного дистрибутива в созданную директорию архивы базовых образов (где <version> - версия приложения):

* r5\_polycore\_grid:<version>.tar
* r5\_polycore\_app:<version>.tar
* r5\_redis:7-alpine.tar
* r5\_postgres13:1.0.tar

1. Выполнить команды:

docker load -i ~/install/r5\_polycore\_grid:<version>.tar

docker load -i ~/install/r5\_polycore\_app:<version>.tar

docker load -i ~/install/r5\_redis:6-alpine.tar

docker load -i ~/install/r5\_postgres13:1.0.tar

1. Создать директорию сервиса

cd ~ && mkdir polycore && cd polycore

1. Скопировать из полученного дистрибутива в созданную директорию инициализирующие компоненты

* docker-compose\_prod.yml.tmpl
* .env.tmpl
* prod.nginx.conf.tmpl
* coolwsd.xml.tmpl l

1. Настроить компоненты:
   1. переименовать docker-compose\_prod.yml.tmpl в docker-compose.yml
   2. заменить в docker-compose.yml все переменные вида {{...}} на актуальные значения
   3. переименовать .env.tmpl в .env
   4. заменить в .env все значения, представленные как {{...}}, на актуальные
   5. переименовать prod.nginx.conf.tmpl в nginx.conf
   6. заменить в nginx.conf все значения, представленные как {{...}}, на актуальные
   7. переименовать coolwsd.xml.tmpl в coolwsd.xml
   8. заменить в coolwsd.xml все значения, представленные как {{...}}, на актуальные
   9. создать директорию файлов БД, указанную в docker-compose.yml в качестве тома
   10. запустить сервис базы данных, предварительно указав желаемый пароль вместо {{ sa\_password }} и полный путь для тома база данных бота на хосте вместо {{ /polycore/database/volume/path }} (тот же, что был задан в docker-compose.yml)
   11. если сервис не должен использовать https:
       1. убрать из docker-compose.yml секреты
       2. убрать из nginx.conf сертификаты и параметр ssl.
2. выполнить

docker run -it --rm -d --name polycore-db -v {{ /polycore/database/volume/path }}:/var/lib/postgresql/data -e POSTGRES\_DB=appdb -e POSTGRES\_USER=sa -e POSTGRES\_PASSWORD={{sa\_password}} -e PGDATA=/var/lib/postgresql/data registry.polyanalitika.ru/r5/r5\_postgres13:1.0

1. выполнить скрипт, предварительно заменив в нем {{ polycore\_password }} на соответствующий пароль из .env

docker exec -it --user postgres polycore-db psql -U sa -d appdb -c "create user polycore with password '{{ polycore\_password }}';create schema polycore;alter schema polycore owner to polycore;alter user polycore set search\_path to polycore;revoke create on schema public from polycore;"

1. остановить контейнер

docker stop polycore-db

1. выполнить миграции структуры базы данных и инициализирующих данных, указав email и пароль администратора сервиса sa

docker compose run --rm -e POLYCORE\_SA\_EMAIL=sa@example.com -e POLYCORE\_SA\_PASSWORD=changeme app flask --app polycore.app.py db upgrade

1. запустить сервис командой docker compose up -d
   * 1. Вспомогательный функционал

Зашифровать строку (полученное значение следует использовать с префиксом fernet:)

docker exec -i polycore-app flask encrypt --text=somepassword

* 1. Порядок проверки работоспособности

Для проверки работы POLYCORE необходимо произвести аутентификацию в системе. Для этого введите в браузере адрес машины, на которой установлено решение (уточняется у администратора), например [https://polycore.server:8000](https://polycore.server:8000/) (порт указывается тот, который был задан при установке). На появившейся форме укажите свой логин и пароль (пароль администратора задаётся при установке системы, логин/пароль пользователя – получаются у администратора), и нажмите кнопку «Вход» в соответствии с Рисунок 1.

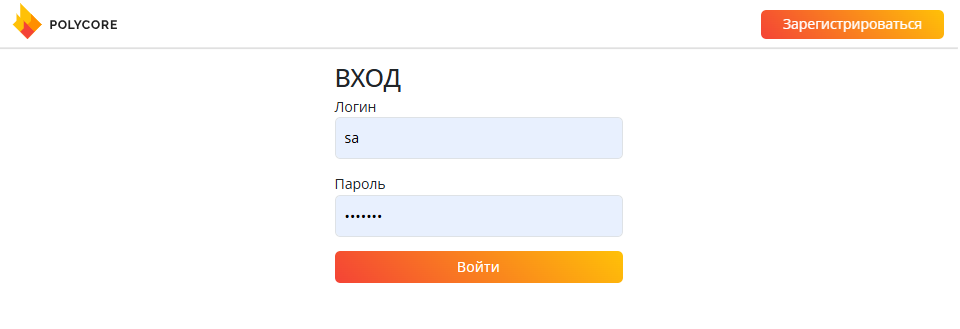


Рисунок . Начальная страница

После успешной аутентификации откроется главная страница приложения, со списком доступных пунктов меню (Рисунок 2). Необходимо выбрать один из них и перейти на форму работы с выбранным объектом, кликнув на соответствующем пункте меню (Рисунок 3).

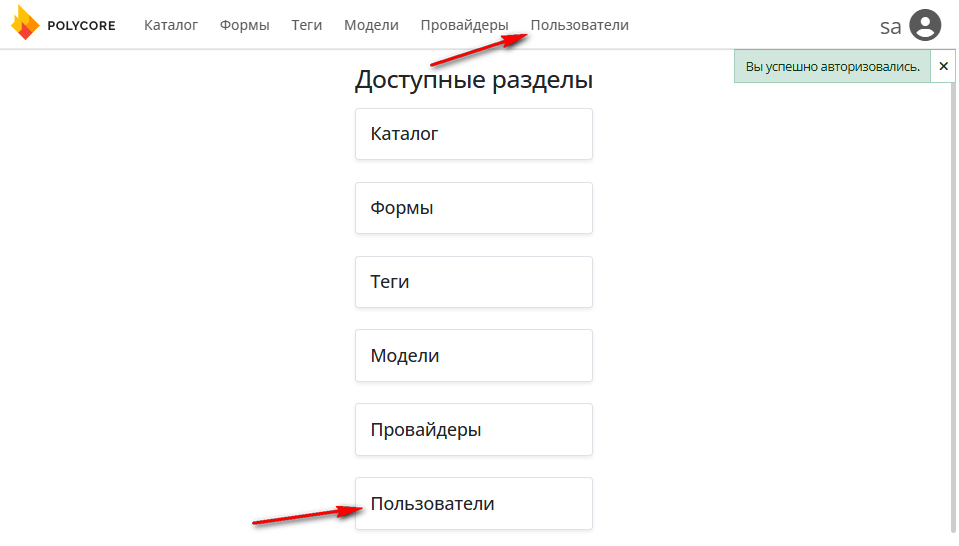


Рисунок . Главная форма приложения

Будет открыта форма информации и управления выбранным объектом. Для проверки используется учетная запись, для которой доступны пункт меню «Провайдеры», «Пользователи» (Рисунок 3) (в данном случае учетная запись администратора sa).

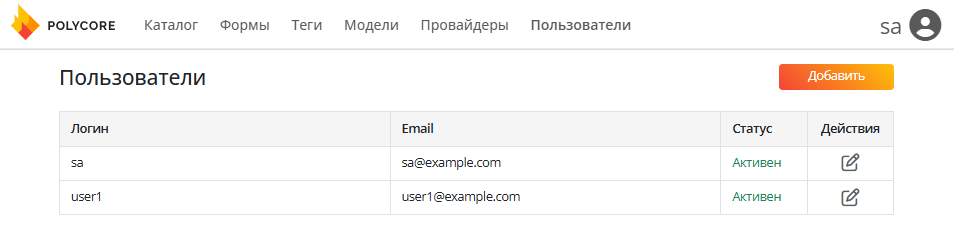


Рисунок . Форма управления пользователями

* 1. Порядок обновления системы

При обновлении необходимо обеспечить наличие в системе образов приложения новой версии, аналогично как при установке.

Ниже указан общий порядок обновления:

1. убедиться, что сервис остановлен
2. обновить версии образов в файле docker-compose\_prod.yml
3. запустить сервис
4. в случае необходимости (уточняется в описании к конкретной версии), выполнить обновление системных таблиц: docker exec -i polycore-app flask db upgrade

Перед обновлением рекомендуется создать резервную копию БД.

1. Описание операций администрирования
   1. Определения и сокращения POLYCORE

Определения и сокращения POLYCORE представлены в Таблица 1.

Таблица . Определения POLYCORE

|  |  |
| --- | --- |
| Термин/Сокращение | Определение |
| Система | Сокращенное наименование Системы консолидации данных POLYCORE |
| Модель | Объект Системы, позволяющий описать связь между сущностями источника и его представлением в системе |
| Форма | Объект Системы, который позволяет настроить интерфейс представления в табличном виде для работы с данными |
| Тег | Объект Системы, который используется для классификации и структурирования Форм и Тегов |
| Провайдер | Объект Системы, предназначенный для настройки подключения к внешним источникам данных |
| Каталог | Раздел Системы, позволяющий просмотр доступных Форм и Тегов в древовидной структуре, с возможностью выполнять действия в соответствии с ролевой моделью пользователя |
| Схема «Звезда» | Модель организации данных в хранилищах, где:   * Центральная таблица фактов содержит ключевые метрики (например, объем продаж). * Таблицы измерений связаны с ней через внешние ключи (например, товары, клиенты, даты). |
| Схема «Снежинка» | Модель организации данных, где:   * Таблицы измерений нормализованы и могут иметь иерархические подтаблицы (например, регион - страна - город). |

* 1. Резервное копирование

Для резервной копии требуется наличие свободного дискового пространства на:

* целевом устройстве под хранение резервной копии
* контейнере polycore-db (директория /tmp).

Функционал резервного копирования и восстановления БД предназначен для использования на одном и том же экземпляре сервиса.

* + 1. Создание резервной копии

Для создания резервной копии БД, выполнить

docker stop polycore-app && \

mkdir -p backups && \

docker exec -t polycore-db pg\_dump -U sa -d appdb > backups/backup\_$(date +%Y-%m-%d).sql && \

docker start polycore-app

* + 1. Восстановление из резервной копии

Для восстановления из резервной копии БД, выполнить

docker stop polycore-app && \

cat backups/backup\_<BACKUP DATE>.sql | docker exec -i polycore-db psql -U sa -d appdb && \

docker exec -i polycore-app flask db upgrade && \

docker start polycore-app

* 1. Настройки компонентов системы
     1. Общие настройки компонентов системы

Перед установкой необходимо проверить наличие файлов ключей и сертификатов key.pem и cert.pem, пути до которых необходимо будет указать в ходе установки.

* + 1. Конфигурационные файлы

Конфигурационные файлы создаются на этапе деплоя или обновлений, и определяют предстартовое окружение и установки системы.

В конфигурационные файлы системы входят:

* docker-compose.yml - настройки конфигурации сервисов,
* .env – файл переменных окружения сервисов,
* nginx.conf - конфигурационный файл для Nginx, обратного прокси-сервера фронтенда,
* coolwsd.xml – файл настроек сервиса grid.

Ниже приведен примеры конфигурационных файлов с описанием заданных настроек в виде комментариев (#).

* + - 1. docker-compose.yml

Пример содержимого файла настройки конфигурации сервисов:

# ======================

# Основные сервисы системы

# ======================

services:

# Сервис базы данных PostgreSQL

db:

image: registry.polyanalitika.ru/r5/r5\_postgres13:1.0 # Образ PostgreSQL

container\_name: polycore-db # Имя контейнера

hostname: polycore-db # Сетевой идентификатор

restart: always # Автоперезапуск при сбоях

ports:

- 2345:5432 # Проброс порта: хост:контейнер

healthcheck: # Проверка работоспособности БД

test: pg\_isready -U polycore -d appdb # Команда проверки

interval: 10s # Интервал проверок

volumes:

- dbdata:/var/lib/postgresql/data # Постоянное хранение данных

networks:

- tier # Подключение к внутренней сети

# Сервис кэширования Redis

redis:

image: registry.polyanalitika.ru/r5/redis:7-alpine

healthcheck:

test: redis-cli ping # Проверка доступности Redis

sysctls:

- net.core.somaxconn=511 # Максимальное число соединений

# Сервис для работы с таблицами

grid:

image: registry.polyanalitika.ru/r5/r5\_polycore\_grid:0.4.3

depends\_on:

db: # Зависимость от работоспособности БД

condition: service\_healthy

environment:

- username=user # Учетные данные для доступа

- password=userpasswd

volumes:

- ./coolwsd.xml:/etc/coolwsd/coolwsd.xml # Конфиг веб-редактора

# Основное приложение

app:

image: registry.polyanalitika.ru/r5/r5\_polycore\_app:0.4.3

depends\_on: # Зависимости от других сервисов

grid:

condition: service\_healthy

redis:

condition: service\_healthy

env\_file:

- ./.env # Файл с переменными окружения

secrets: # SSL-сертификаты

- key

- cert

ports:

- "8800:8000" # Основной порт приложения

volumes:

- ./nginx.conf:/etc/nginx/conf.d/app.conf # Конфиг Nginx

command: bash -c "/app/start\_https.sh" # Скрипт запуска с HTTPS

environment:

- WTF\_CSRF\_SSL\_STRICT=False # Отключение строгой проверки SSL для CSRF

extra\_hosts: # Пользовательские DNS-записи

- "polycore.polyanalitika.ru:111.222.333.444"

# ======================

# Инфраструктурные настройки

# ======================

networks:

tier: # Внутренняя сеть для коммуникации сервисов

driver: bridge # Тип сетевого драйвера

secrets: # Управление SSL-сертификатами

key:

file: key.pem # Приватный ключ

cert:

file: cert.pem # Сертификат

volumes:

dbdata: # Постоянное хранилище для PostgreSQL

driver: local

driver\_opts:

type: 'none'

o: 'bind'

device: '/home/visiology/polycore/volume/data' # Путь к данным на хосте

* + - 1. .env

Пример содержимого файла переменных окружения бэкенда:

# ==================================

# Настройки производительности и инфраструктуры

# ==================================

FLASK\_WORKERS=4 # Количество процессов обработки запросов Flask (оптимально для 4-ядерного CPU)

# Подключение к PostgreSQL:

# пользователь:пароль@хост/база\_данных

DATABASE\_URL=postgresql://user:passwd@polycore-db/appdb

# ==================================

# Настройки безопасности и шифрования

# ==================================

# Ключ для подписи сессий и CSRF-токенов

SECRET\_KEY=XoMu13cASkBvucnAShsGYVReYWyOEGR6vrvWGXYJ\_WQ

# Ключ для шифрования данных (Fernet)

FERNET\_KEY=tolDjRfrXlIZJAJEDJ1hO1kOi5FIw-JgC59QJ5pelrk=

# ==================================

# Настройки приложения и логирования

# ==================================

# Уровень детализации логов (DEBUG/INFO/WARNING/ERROR/CRITICAL)

LOG\_LEVEL=DEBUG

# Время кеширования статических файлов (1 год в секундах)

SEND\_FILE\_MAX\_AGE\_DEFAULT=31556926

# ==================================

# Доменные настройки

# ==================================

# Основной домен системы

POLYCORE\_DOMAIN=polycore.polyanalitika.ru

# Используемый протокол (http/https)

POLYCORE\_PROTOCOL=https

* + - 1. nginx.conf

Пример содержимого конфигурационного файла для Nginx:

# ======================

# Глобальные настройки сервера

# ======================

server {

# Прослушивание HTTPS на порту 8000

listen 8000 ssl;

# Доменное имя сервера

server\_name polycore.polyanalitika.ru;

# Корневая директория статики

root /app/static/build;

# Дефолтный индексный файл

index /;

# Лог доступа

access\_log /app/logs/nginx-access.log;

# Лог ошибок

error\_log /app/logs/nginx-error.log;

# SSL-сертификат из Docker-секретов

ssl\_certificate /run/secrets/cert;

# Приватный ключ из Docker-секретов

ssl\_certificate\_key /run/secrets/key;

# Макс. размер занятых буферов

proxy\_busy\_buffers\_size 512k;

# Число и размер буферов ответа

proxy\_buffers 4 512k;

# Размер буфера для заголовков

proxy\_buffer\_size 256k;

# ======================

# Маршрутизация запросов

# ======================

# Обработка запросов к сервису Grid

location /grid {

# Проксирование на grid-сервис

proxy\_pass http://polycore-grid:9980;

# WebSocket-совместимые заголовки

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection "Upgrade";

# Передача информации о клиенте

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

# Таймаут для долгих операций (10 часов)

proxy\_read\_timeout 36000s;

}

# Основное проксирование на backend-приложение

location / {

# Проксирование на основное приложение

proxy\_pass https://polycore-app:5000;

# Стандартные заголовки для прокси

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

}

* + - 1. coolwsd.xml

coolwsd.xml — конфигурационный файл для сервиса Grid (веб-редактор документов).

Он определяет параметры работы с документами в браузерном интерфейсе системы POLYCORE.

Ключевые настройки:

1. Сетевая конфигурация:

* Порт и адрес для подключения к сервису.
* Настройки WebSocket-соединений для реального времени.
* Интеграция с основным приложением POLYCORE через REST API.

1. Безопасность:

* SSL-сертификаты для HTTPS.
* Ограничения доступа (IP-фильтрация, токены аутентификации).
* Политики CORS для кросс-доменных запросов.

1. Ограничения документов:

* Максимальный размер файлов.
* Поддерживаемые форматы (ODS, XLSX, DOCX и др.).
* Лимиты на использование памяти и CPU.

1. Интеграция с POLYCORE:

* Пути для сохранения данных во внешние хранилища через Провайдеры.
* Настройки совместного редактирования (многопользовательский режим).
* Логирование событий в общую систему мониторинга.

1. Оптимизация:

* Кэширование документов.
* Таймауты обработки запросов.
* Настройки пула потоков для параллельной работы.

Роль в системе:

* Обеспечивает взаимодействие между Формами POLYCORE и веб-редактором документов.
* Управляет конвертацией файлов, совместным редактированием и отображением данных в табличном виде.
* Связан с сервисом app через прокси в Nginx (см. конфиг /grid).

Файл монтируется в Docker-контейнер polycore-grid и требуется для старта сервиса.

Изменения в конфиге применяются только после перезапуска контейнера.

* + 1. Ограничение ресурсов для контейнера

Для ограничения ресурсов контейнера необходимо внести изменения в файл docker-compose.yml:

* resources: настраивает ограничения физических ресурсов для запуска контейнера на платформе. Эти ограничения могут быть настроены как:
  + limits: платформа должна запрещать контейнеру выделять больше;
  + reservations: платформа должна гарантировать, что контейнер может выделить как минимум заданный объем;

services:

frontend:

image: awesome/webapp

deploy:

resources:

limits:

cpus: '0.50'

memory: 50M

pids: 1

reservations:

cpus: '0.25'

memory: 20M

* cpus: настраивает ограничение или резервирование того, сколько доступных ресурсов ЦП (как количество ядер) может использовать контейнер
  + memory: настраивает ограничение или резервирование объема памяти, который может выделять контейнер, в виде строки, выражающей значение байта;
  + pids: настраивает ограничение PID контейнера, заданное как целое число.
  1. Настройка объектов системы

В зависимости от пользователя и его роли интерфейс Системы может отличаться. Данный раздел описывает операции, доступные только пользователям с ролью «Администратор» (Таблица 2). При отсутствии разделов и некоторых операций Системы ознакомьтесь с вашей ролевой моделью (Таблица 3, Таблица 4).

Администратор обладает полным контролем над системой, включая управление провайдерами и пользователями, которые доступны только ему (Рисунок 4).

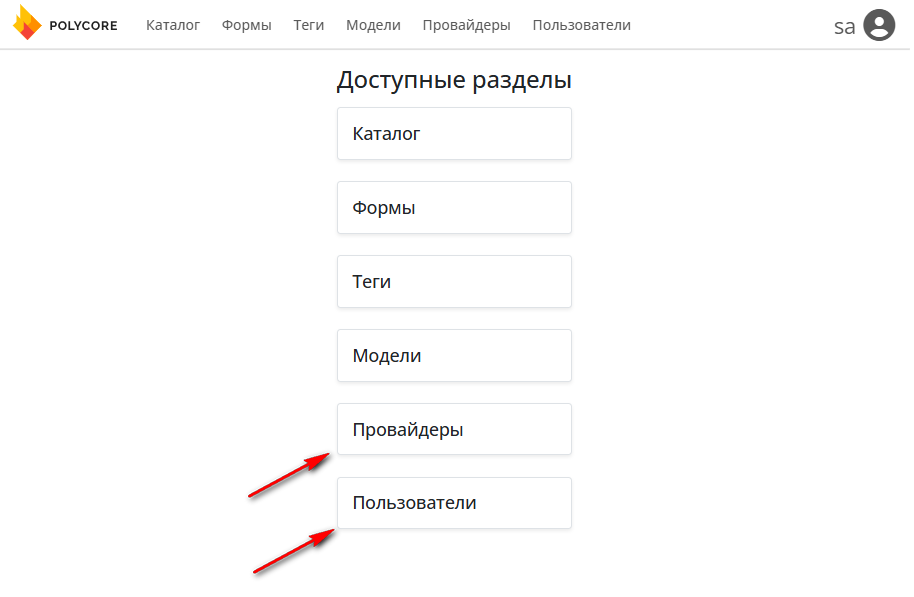


Рисунок . Доступные администратору разделы системы

* + 1. Провайдеры

Провайдер – объект Системы, предназначенный для подключения к внешним источникам данных. Раздел Провайдеры позволяет настроить такие подключения. Список доступных типов провайдеров:

* MS SQL - провайдер для подключения к базе данных MS SQL
* PostgreSQL – провайдер для подключения к базе данных PostgreSQL
* ODBC - провайдер для подключения к базе данных с поддержкой ODBC
* JDBC – провайдер для обращения к базе данных с поддержкой JDBC

Для перехода в раздел нажмите в верхнем меню или на главной форме на пункт «Провайдеры» (Рисунок 5).

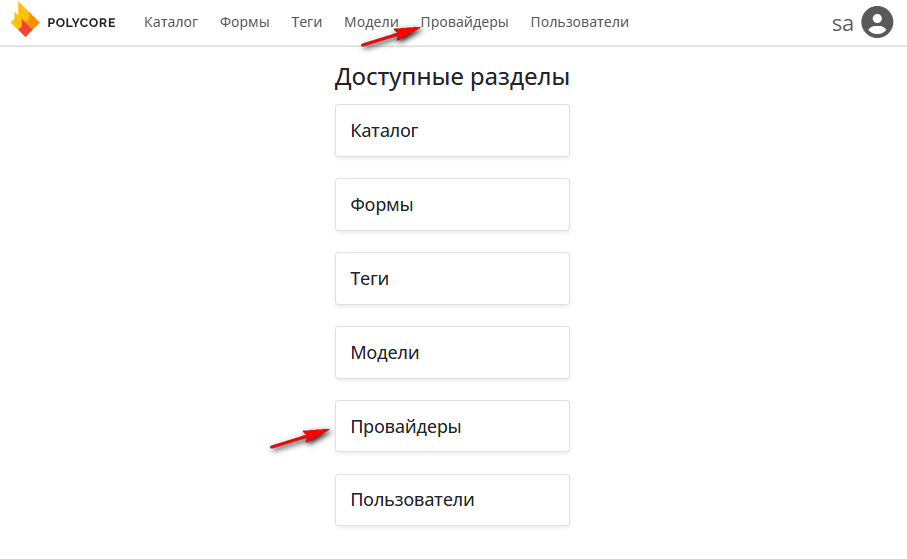


Рисунок . Пункты меню провайдеры

Доступ к разделу, доступен только авторизированному пользователю с ролью «Администратор». Ознакомиться с функциональными возможностями ролей возможно в Таблица 4.

При переходе в раздел вам будет предоставлен список имеющихся провайдеров в системе (Рисунок 6).

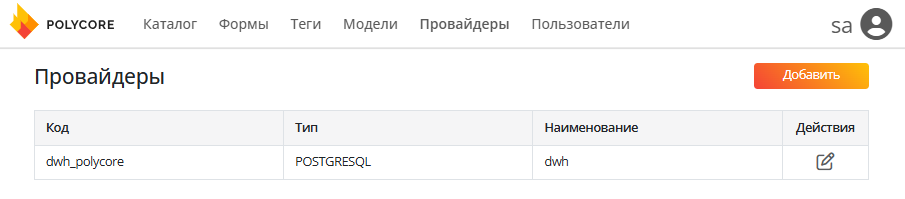


Рисунок . Доступные провайдеры

* + - 1. Создание провайдера

1. Для создания провайдера нажмите на кнопку «Добавить» в верхнем правом углу таблицы списка провайдеров (Рисунок 7).

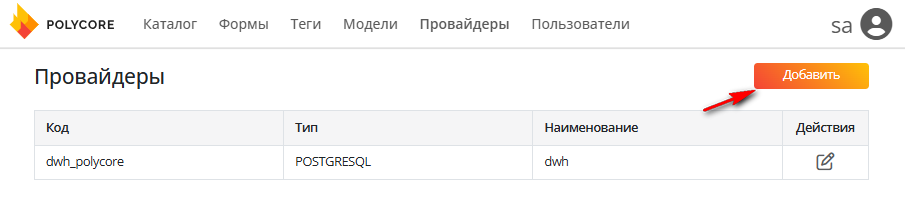


Рисунок . Переход на форму добавления провайдера

1. В открывшемся окне (Рисунок 8) необходимо заполнить обязательные поля:

* Код: введите код, который является уникальным и будет идентифицировать провайдер в системе.
* Тип: укажите тип подключения из доступного списка.
* Наименование: укажите наименование для отображения провайдера в системе.
* URL: укажите адрес, для подключения к провайдеру.
* Хост: укажите хост, по которому необходимо совершить подключение.
* Порт: укажите порт.
* Схема: укажите схему в базе данных, в которой хранятся данные.
* База данных: укажите базу данных, с таблицами которых мы взаимодействуем.
* Пользователь: укажите пользователя, под которым нужно получить доступ к ресурсам источника.
* Пароль: укажите пароль от сервиса.

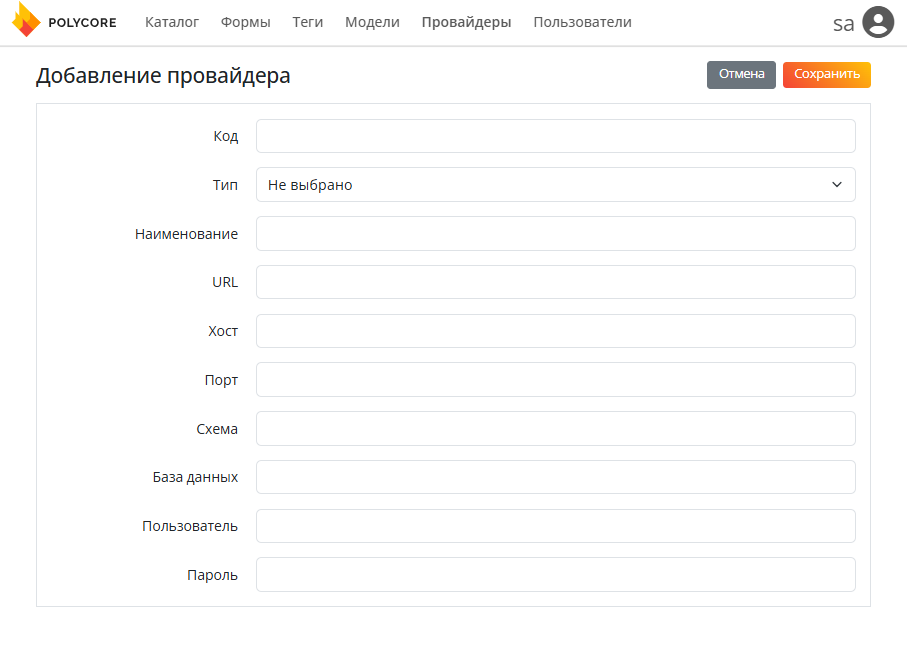


Рисунок . Форма добавления провайдера

1. Далее после нажатия кнопки «Сохранить» (Рисунок 9) в списке появится созданный провайдер (Рисунок 10).

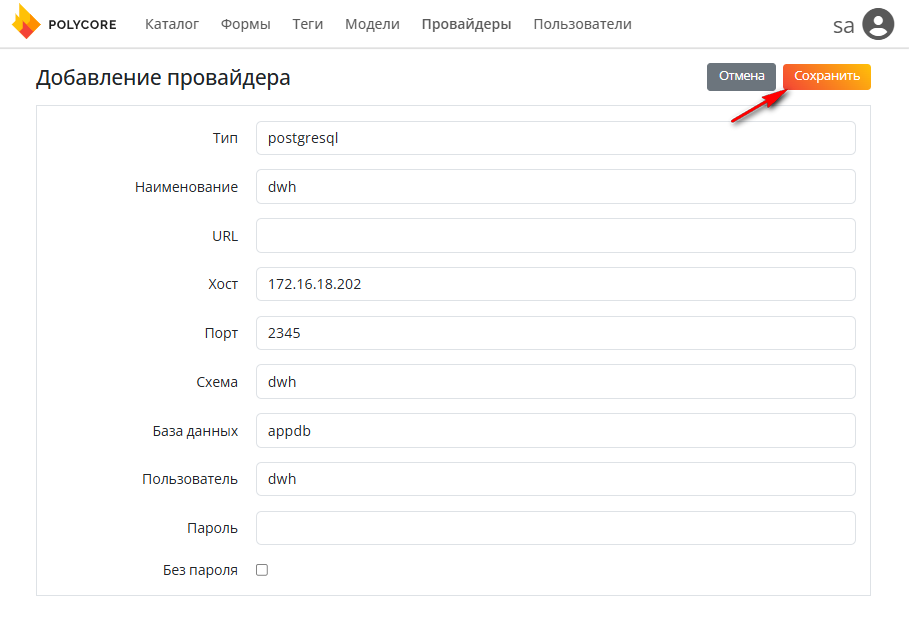


Рисунок . Сохранение параметров провайдера

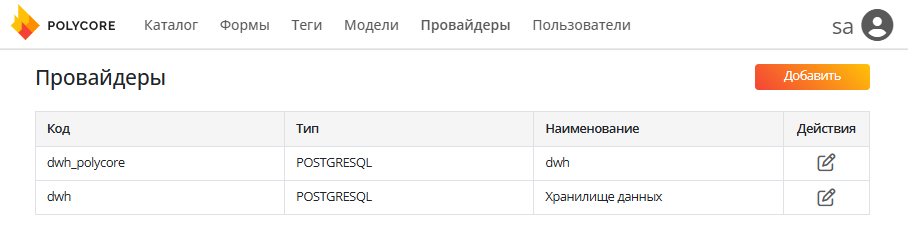


Рисунок . Добавленный провайдер в списке провайдеров

* + - 1. Редактирование провайдера

Форма редактирования идентична форме создания Провайдера за исключением поля «Код». Если вам необходимо сменить указанные данные провайдера, нажмите на кнопку «Редактировать» в панели действий с провайдером (Рисунок 11), на открывшейся форме внесите изменения (Рисунок 12) и нажмите кнопку «Сохранить» (Рисунок 13).

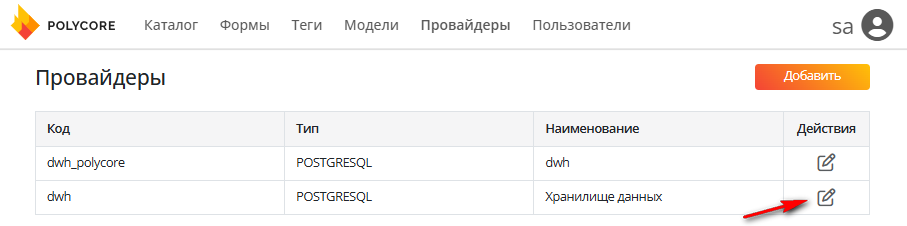


Рисунок 11. Переход на форму редактирования провайдера

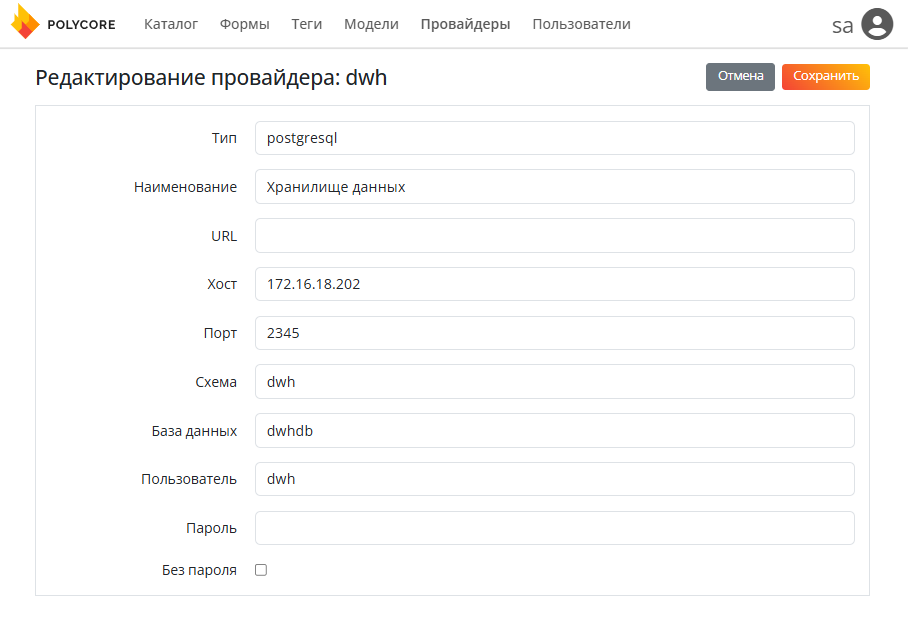


Рисунок 12. Форма редактирования провайдера

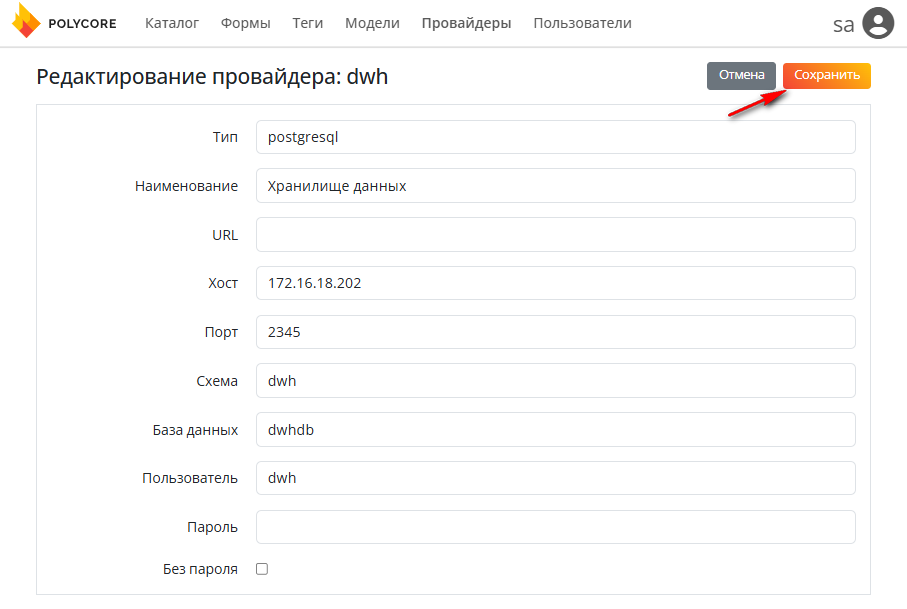


Рисунок 13. Сохранение параметров провайдера

* + 1. Пользователи

Функционал предназначен для работы с пользователями системы. Доступны следующие операции:

* создание новых пользователей,
* редактирование существующих пользователей,
* назначение пользователям ролей,
* активация и деактивация учётных записей,
* выдача прав,
* смена пароля.

Для перехода в раздел нажмите в верхнем меню или на главной форме на пункт «Пользователи» (Рисунок 14).

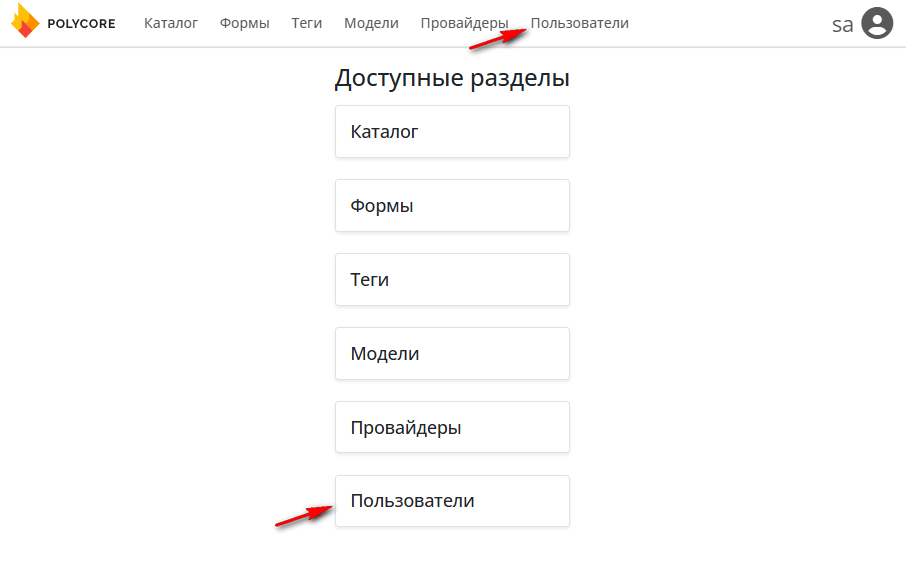


Рисунок . Пункты меню пользователи

Доступ к разделу, доступен только авторизированному пользователю с ролью «Администратор». Ознакомиться с функциональными возможностями ролей возможно в Таблица 3, Таблица 4.

При переходе в раздел вам будет предоставлен список имеющихся пользователей в системе (Рисунок 15).

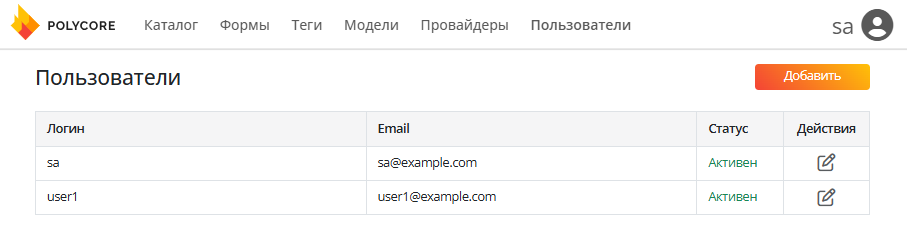


Рисунок . Список пользователей

Для более детального ознакомления в карточкой пользователя нажмите на кнопку в столбце «Действия» - «Редактировать» (Рисунок 16, Рисунок 17).

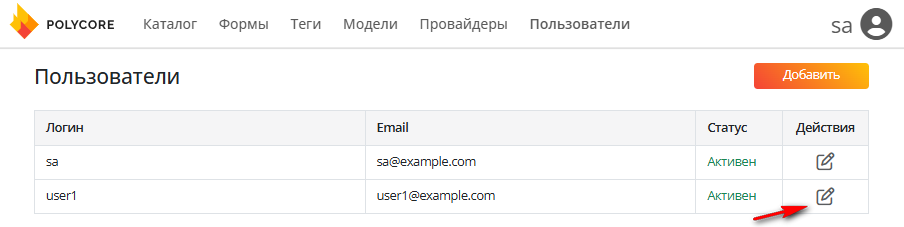


Рисунок . Переход на форму редактирования пользователя

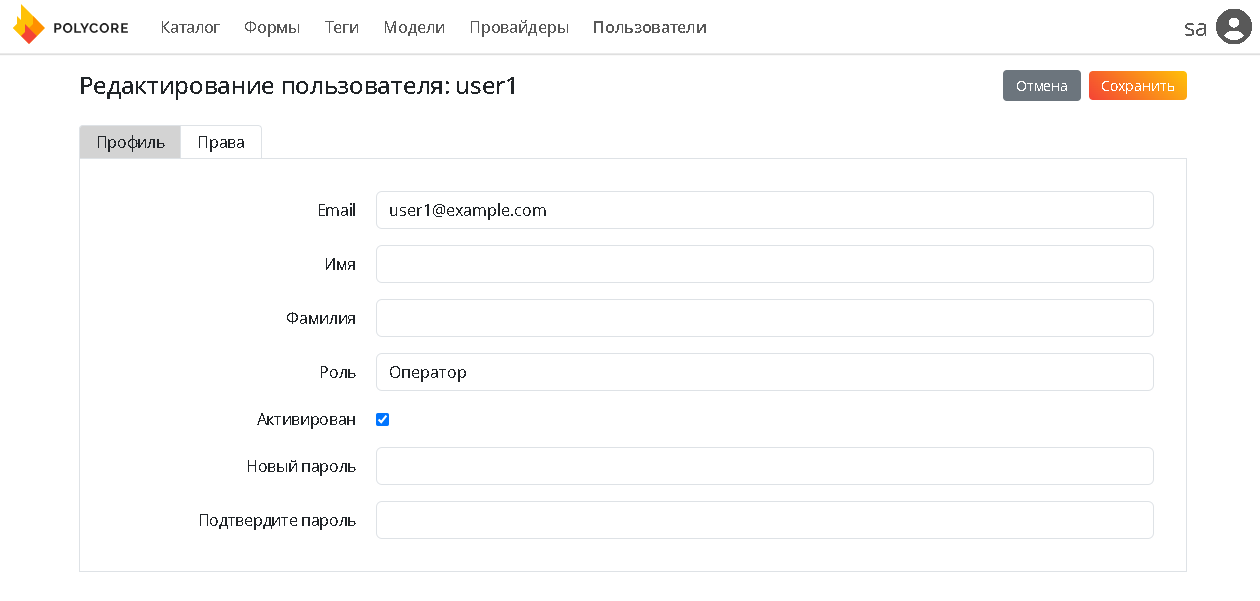


Рисунок 17. Форма редактирования пользователя

* + - 1. Ролевая модель

Предусмотрена ролевая модель доступа к функциональным возможностям Системы. Список и описание ролей приведены в Таблица 2.

Таблица . Роли пользователей

| Термин/Сокращение | Определение |
| --- | --- |
| Читатель | Роль для просмотра доступных Форм и данных в них |
| Оператор | Роль для работы только с Формами и данными в них |
| Редактор | Роль для формирования Форм и шаблонов к ним на основе готовых Моделей, имеющихся в системе |
| Дизайнер | Роль для формирования Моделей данных и описания зависимостей с внешними источниками и представлением Форм |
| Администратор | Роль для управления пользователями и подключения к внешним источникам |

В зависимости от роли пользователя различается доступ к разделам (Таблица 3) и функциям Системы (Таблица 4).

Таблица . Доступные разделы системы и роли пользователей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Интерфейс | Администратор | Дизайнер | Редактор | Оператор | Читатель |
| Каталог | + | + | + | + | + |
| Формы | + | + | + |  |  |
| Модели | + | + |  |  |  |
| Теги | + | + | + |  |  |
| Провайдеры | + |  |  |  |  |
| Пользователи | + |  |  |  |  |

Таблица . Доступные функциональные возможности и роли пользователей

| Функциональные возможности | Администратор | Дизайнер | Редактор | Оператор | Читатель |
| --- | --- | --- | --- | --- | --- |
| Просмотр данных в Форме | + | + | + | + | + |
| Ввод и редактирование данных в Форме | + | + | + | + |  |
| Создание Форм | + | + | + |  |  |
| Редактирование Форм (привязка модели, описание, привязка тегов) | + | + | + |  |  |
| Редактирование шаблонов Форм | + | + | + |  |  |
|  |  |  |  |  |  |
| Удаление Форм | + | + | + |  |  |
| Добавление Тегов | + | + |  |  |  |
| Редактирование Тегов | + | + |  |  |  |
| Добавление моделей | + | + |  |  |  |
| Редактирование Моделей (привязка провайдера, описание модели) | + | + |  |  |  |
| Удаление Моделей | + | + |  |  |  |
| Добавление Провайдеров | + |  |  |  |  |
| Редактирование провайдеров | + |  |  |  |  |
| Добавление пользователей | + |  |  |  |  |
| Редактирование пользователей | + |  |  |  |  |
| Активация/ деактивация пользователей | + |  |  |  |  |
| Предоставление прав на объекты пользователям | + | + (владелец объекта) | + (владелец объекта) |  |  |

* + - 1. Создание пользователя

Для создания пользователя нажмите на кнопку «Добавить» над таблицей списка пользователей (Рисунок 18) и на открывшейся форме создания пользователя (Рисунок 19) заполните поля соответствующими данными.

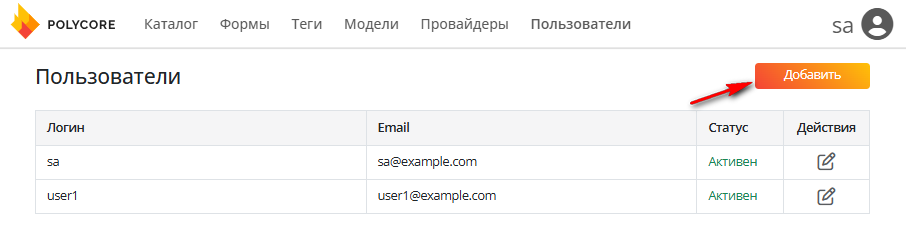


Рисунок . Переход на форму добавления пользователя

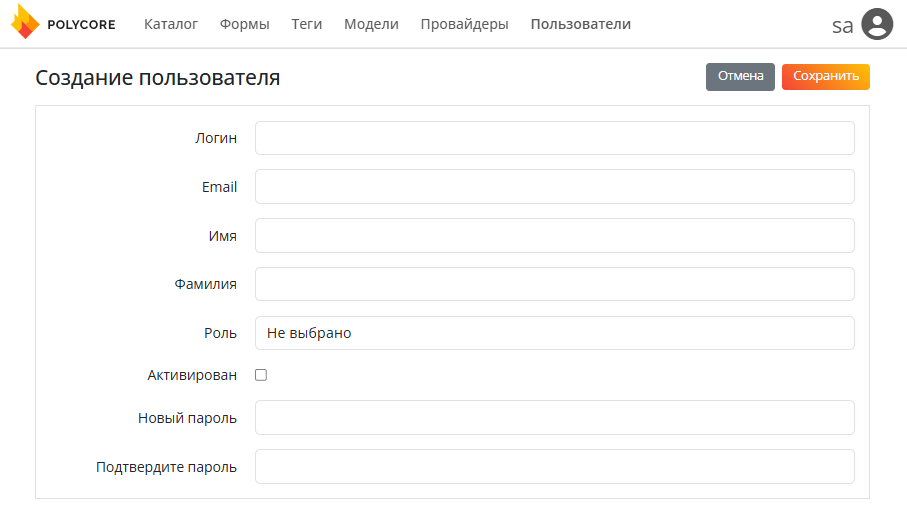


Рисунок . Форма создания пользователя

Поля, доступные для заполнения:

* Логин: введите логин, который будет использоваться пользователем для авторизации в системе,
* Email: введите Email пользователя, для идентификации пользователя в системе
* Имя: введите имя сотрудника
* Фамилию: введите фамилию сотрудника
* Роль: выберите роль из списка, который будет доступен пользователю. Описание функциональных возможностей каждой роли указано в Таблица 4.
* Пароль: введите пароль, который будет использоваться пользователем для авторизации в системе,
* Подтвердите пароль: подтвердите пароль, который был указан в поле Пароль
* Активирован: проставьте чек бокс, если пользователю должен быть активен и иметь возможность авторизоваться в системе.

После завершения заполнения полей нажмите кнопку «Сохранить» (Рисунок 20). Запись о созданном пользователе отобразится в таблице списка доступных пользователей раздела (Рисунок 21).

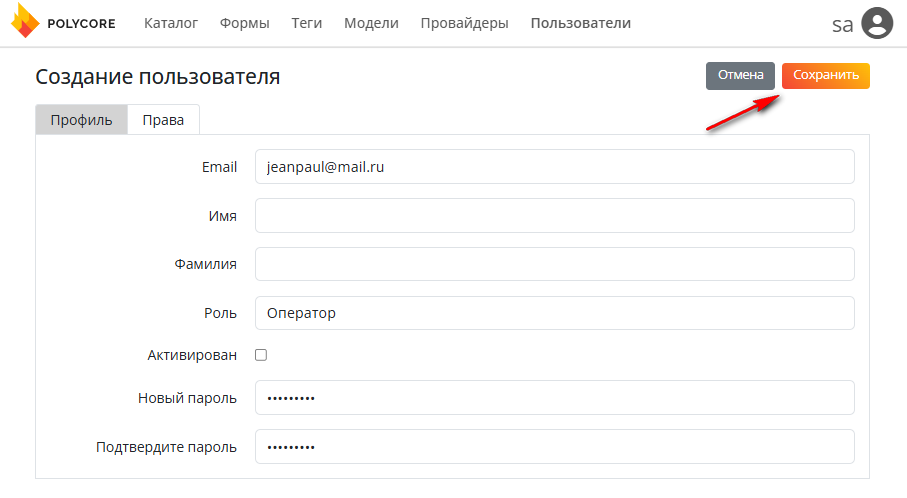


Рисунок . Сохранение данных пользователя

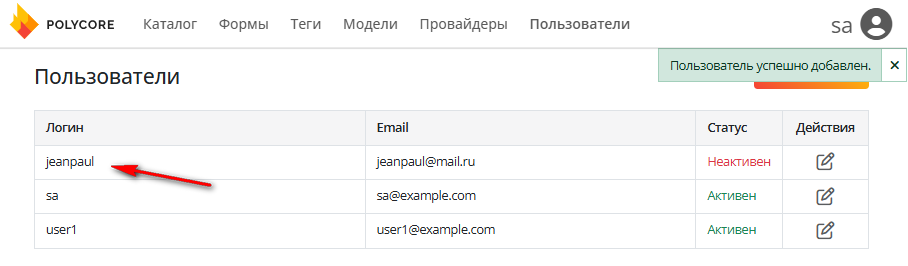


Рисунок . Созданный пользователь в списке пользователей

* + - 1. Редактирование пользователей

Для редактирования пользователя нажмите на кнопку «Редактировать» в строке в панели «Действия» (Рисунок 22) и внесите необходимые изменения в открывшейся форме и нажмите кнопку «Сохранить» (Рисунок 23). После чего информация по пользователю будет обновлена (Рисунок 24).

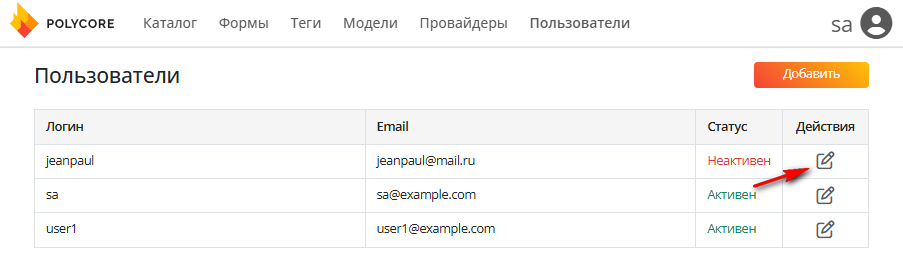


Рисунок . Переход на форму редактирования пользователя

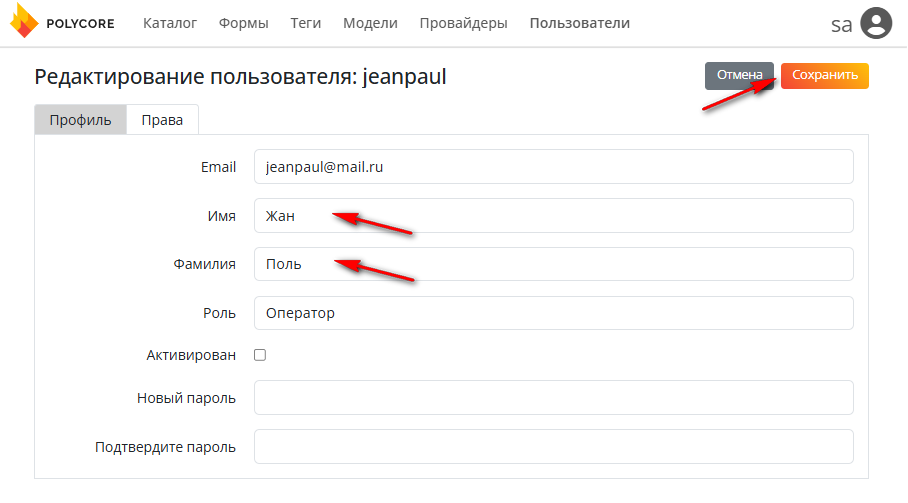


Рисунок . Обновление информации по пользователю

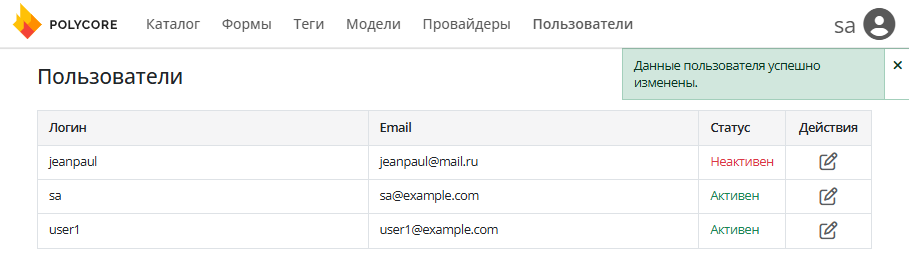


Рисунок . Успешное обновление информации по пользователю

* + - 1. Активация и деактивация пользователей

Для активации или деактивация сотрудника, которому нужно предоставить или отключить доступ откройте пользователя на редактирование и установите или уберите галочку с чекбокса в поле Активирован (Рисунок 25). После нажатия на кнопку «Сохранить» изменения будут применены (Рисунок 26).

При деактивации сотрудника, доступ в систему будет отозван до следующей активации сотрудника (Рисунок 27).

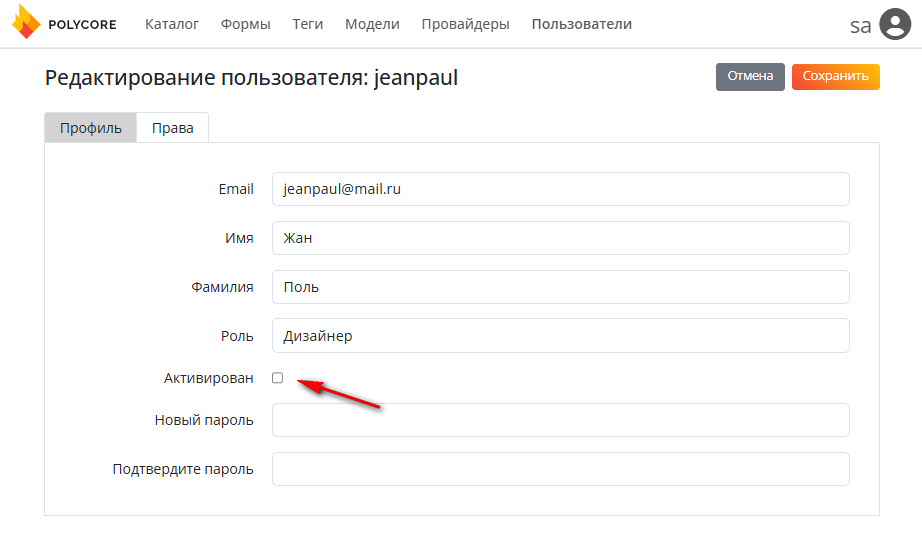


Рисунок . Деактивация пользователя

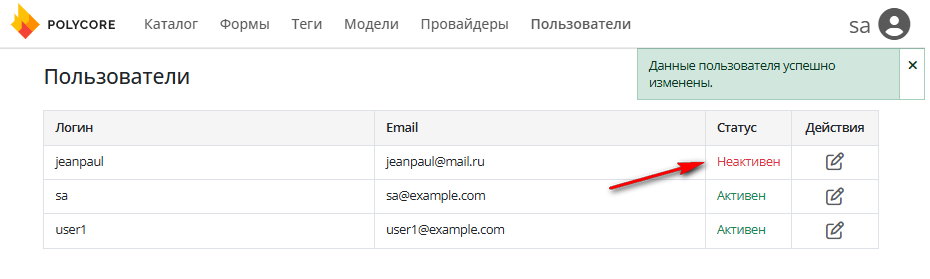


Рисунок . Успешная активация пользователя

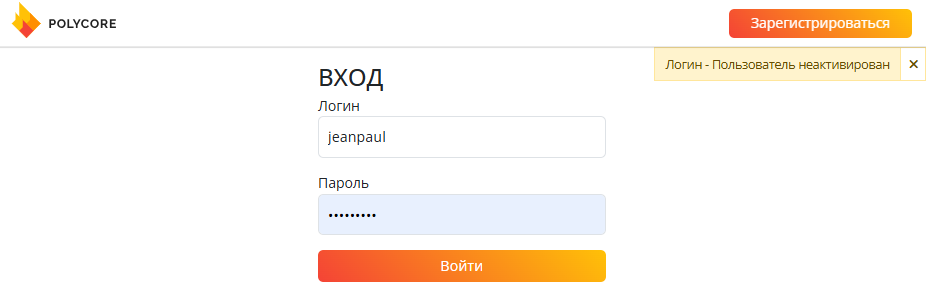


Рисунок . Попытка входа пользователя с неактивной учетной записью

* + - 1. Выдача прав

Для предоставления прав на объекты системы необходимо перейти на форму редактирования пользователя по кнопке «Редактировать» в панели действий (Рисунок 28).

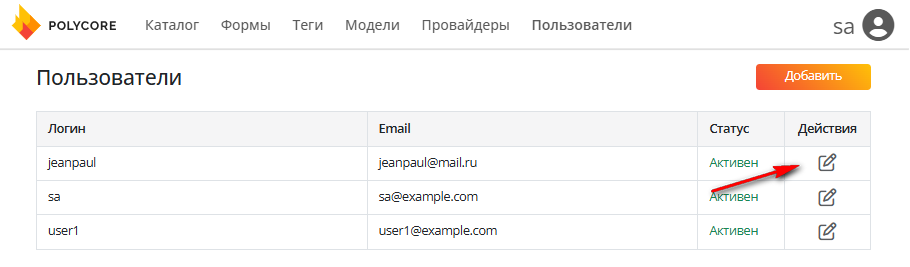


Рисунок . Переход на форму редактирования пользователя

На открывшейся форме редактирования, доступна вкладка Права (Рисунок 29).

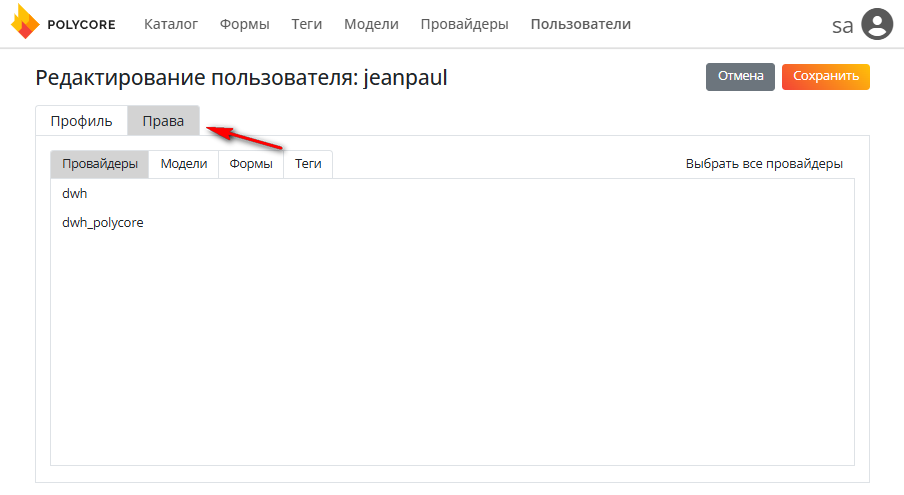


Рисунок . Права пользователя на объекты системы

На данной вкладке доступны все объекты принадлежащие к разделам и отсортированные по вкладкам (Рисунок 30, Рисунок 31, Рисунок 32, Рисунок 33).

Для выбора нескольких объектов вам нужно зажать кнопку Ctrl и выбрать кликом доступные для пользователя объекты.

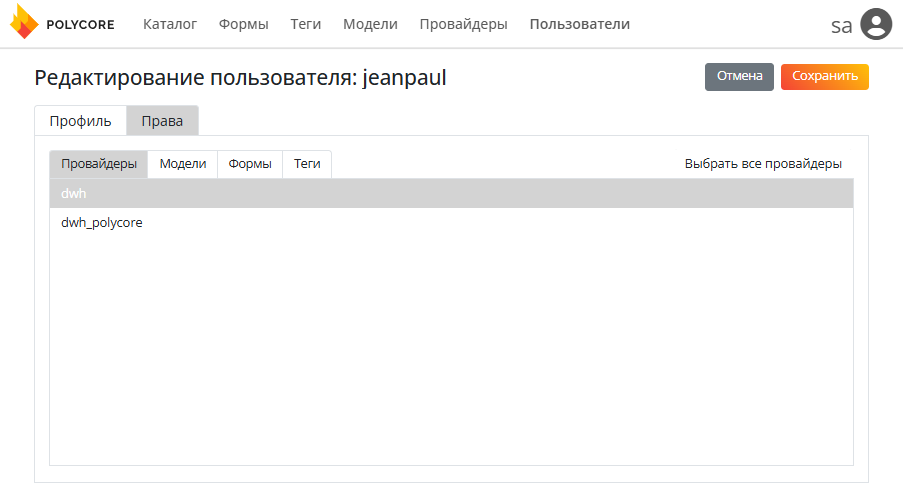


Рисунок . Добавление пользователю права на провайдеры

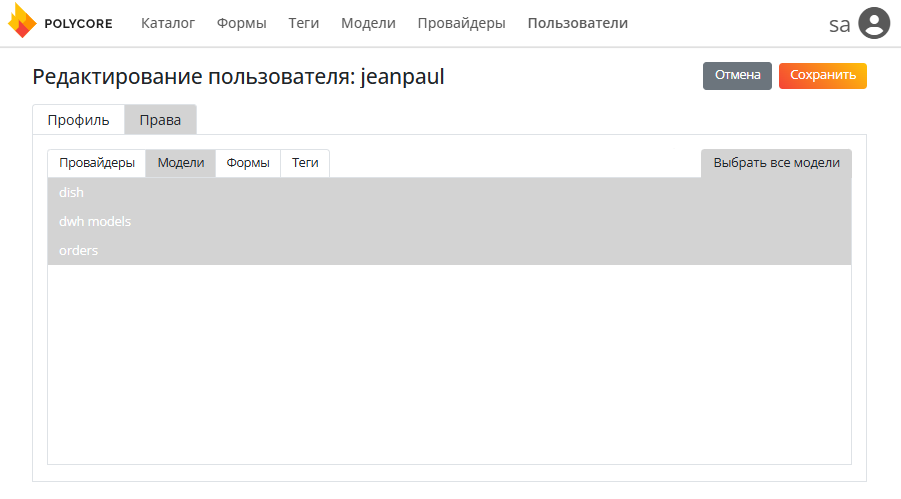


Рисунок . Добавление пользователю права на модели

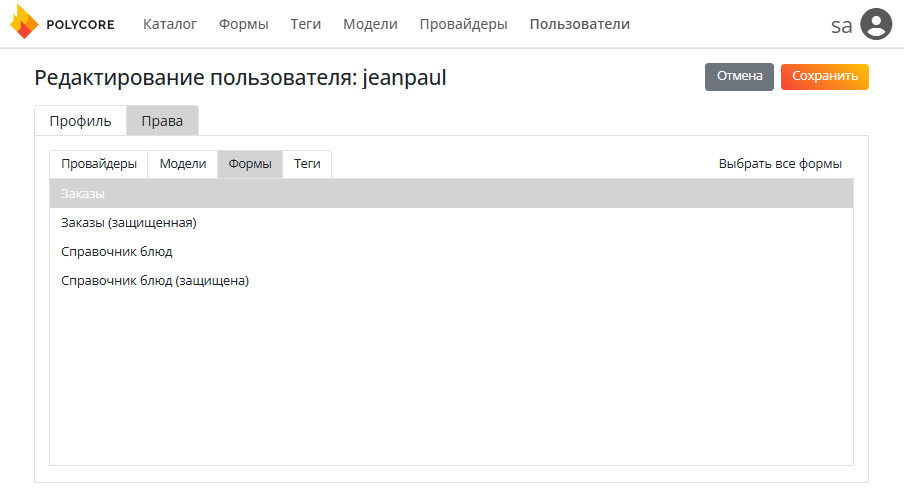


Рисунок . Добавление пользователю права на формы

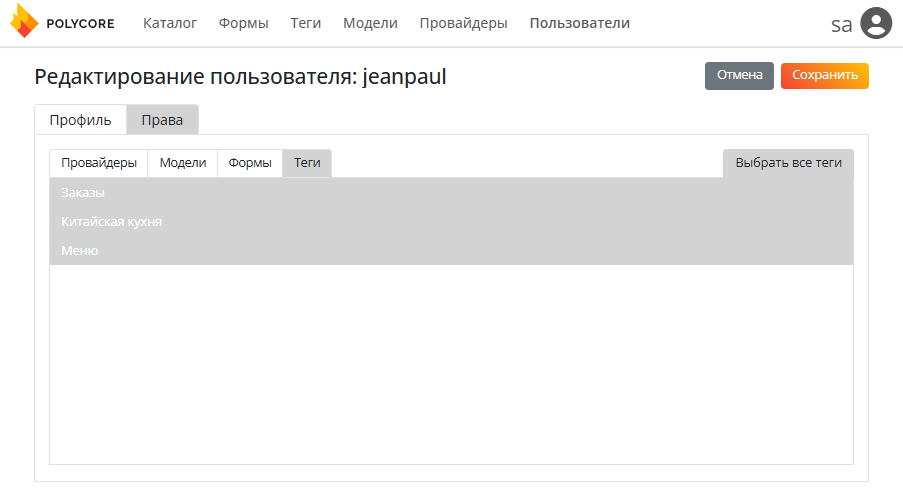


Рисунок . Добавление пользователю права на теги

Для сохранения изменений необходимо нажать на кнопку «Сохранить» (Рисунок 34, Рисунок 35).

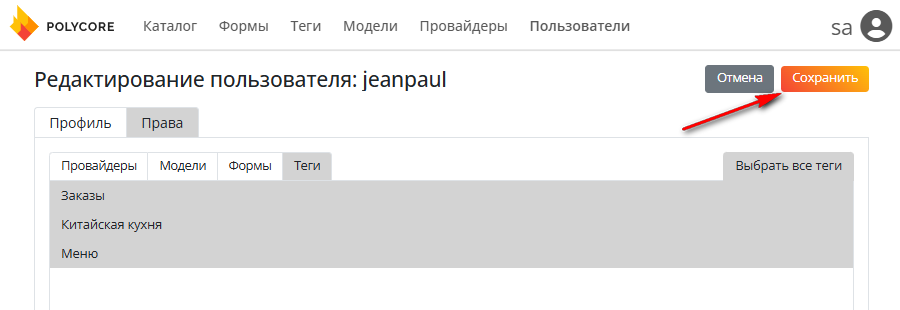


Рисунок . Сохранение изменений по правам

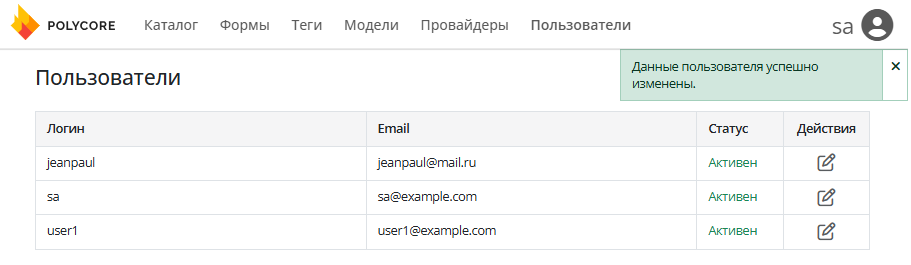


Рисунок . Успешное сохранение изменений по правам

**Важно**: Если пользователь не имеет доступ к разделу, то объекты на которые ему предоставлен доступ не будут доступны, до смены роли, в которой функциональные возможности доступны. Например, пользователю с ролью Оператор (Рисунок 36), доступен только раздел Каталог на главной форме приложения (Рисунок 37).

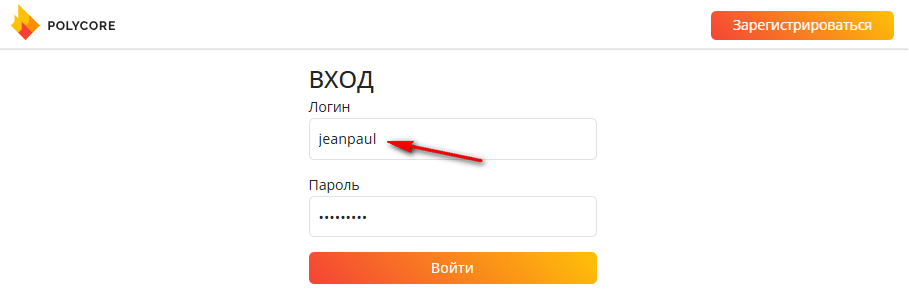


Рисунок . Авторизация под пользователем jeanpaul (роль оператор)

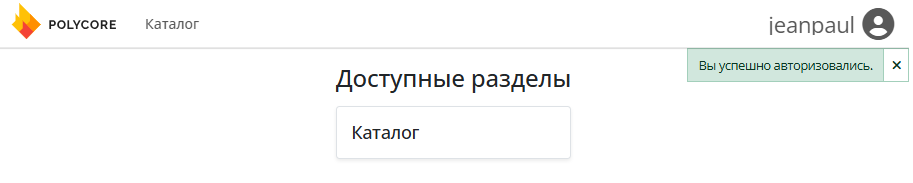


Рисунок . Разделы доступные для пользователя jeanpaul (роль оператор)

После изменения роли пользователя, список доступных разделов также меняется (Рисунок 38, Рисунок 39, Рисунок 40).

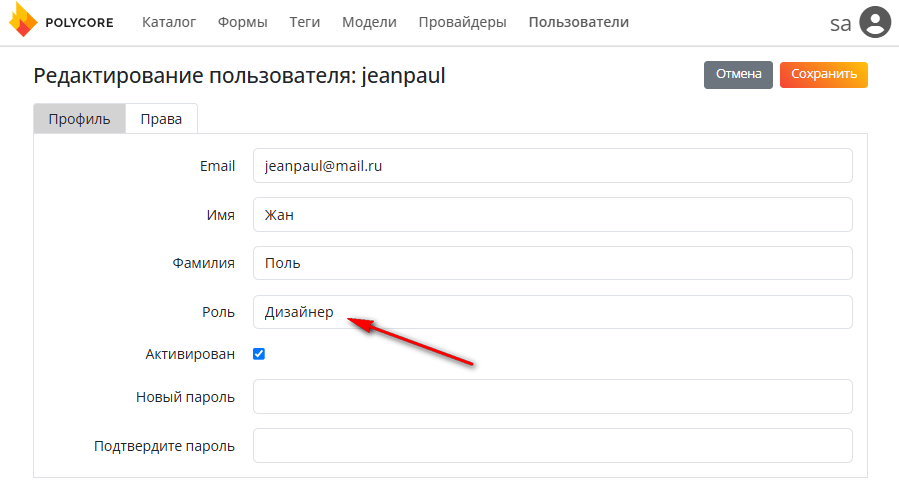


Рисунок . Повышение пользователя jeanpaul до роли дизайнера

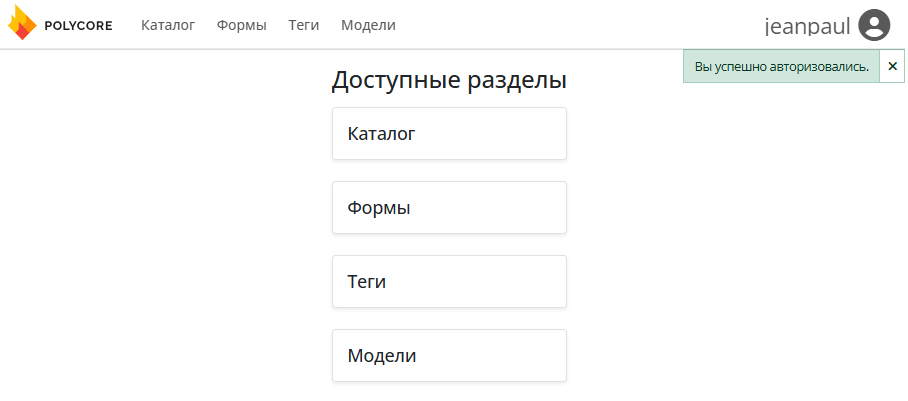


Рисунок . Разделы доступные для пользователя jeanpaul (роль дизайнер)

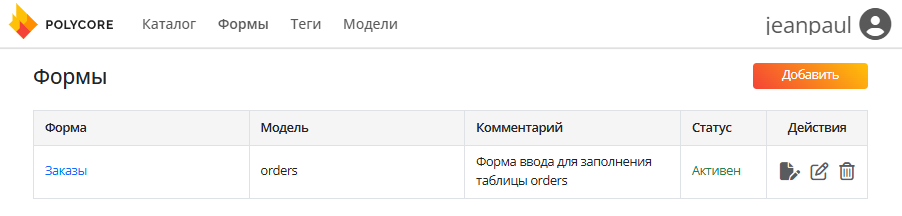


Рисунок . Формы доступные для пользователя jeanpaul (роль оператор)

* + - 1. Редактирования профиля пользователя

Каждый пользователь Системы может отредактировать свой профиль.

Для перехода в профиль пользователя нажмите в правом верхнем углу на свой логин и выберите из списка доступных «Мой профиль» (Рисунок 41).

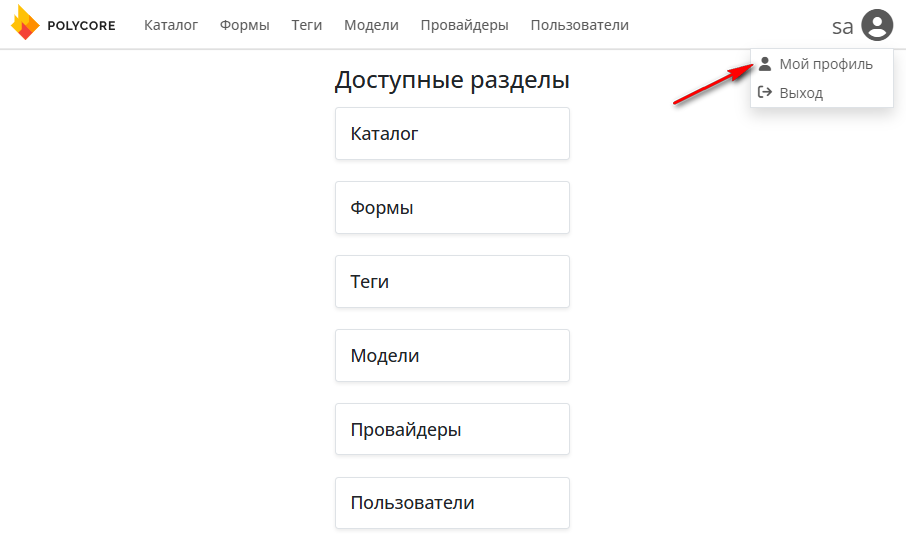


Рисунок . Переход на форму профиля пользователя

Пользователь может поменять данные по (Рисунок 42):

* Email
* Имя
* Фамилия
* Пароль: для сброса пароля, пользователю необходимо ввести свой старый пароль и подтвердить новый, после чего сохранить изменения, нажав кнопку «Сохранить» (Рисунок 43, Рисунок 44).

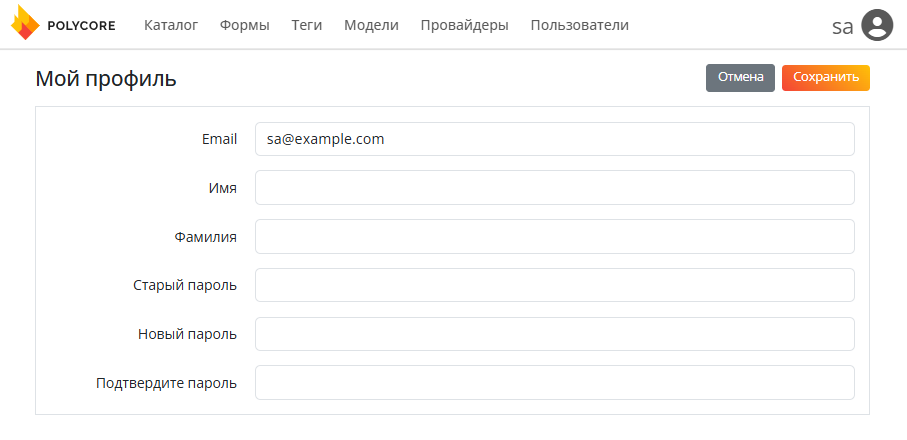


Рисунок . Форма профиля пользователя

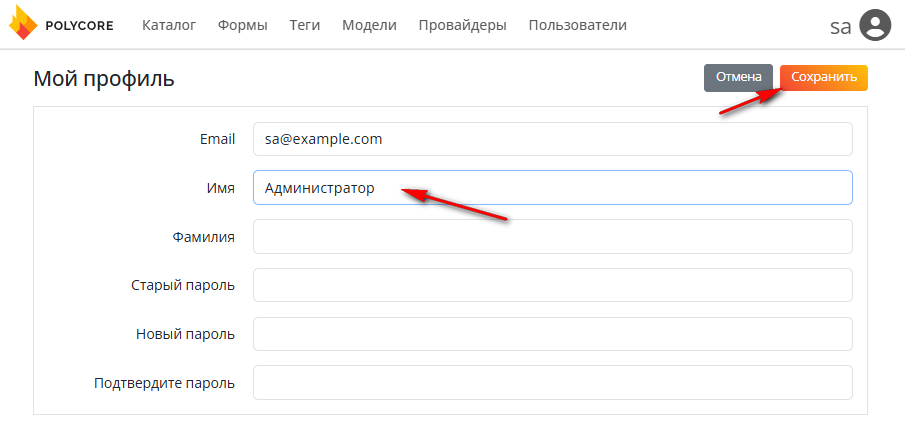


Рисунок . Сохранение изменений профиля пользователя

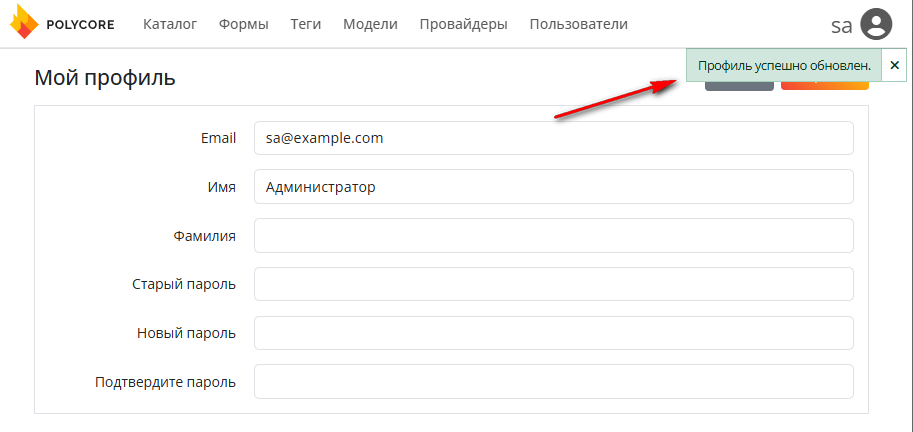


Рисунок . Успешное сохранение изменений профиля пользователя

1. Аварийные ситуации

Для Системы определены следующие режимы функционирования:

* штатный;
* аварийный.

Аварийный режим функционирования Системы используется при отказе одного или нескольких компонент программного и (или) технического обеспечения.

При переходе в аварийный режим в Системе предусмотрено формирование соответствующего информационного сообщения.

После выдачи сообщения, администратору необходимо выполнить комплекс мероприятий по устранению причины перехода Системы в аварийный режим.

При работе с АИС могут возникнуть следующие неисправности, приводящие к аварийным ситуациям:

* Превышение нагрузки на АИС. В этом случае необходимо ограничить количество тяжело-нагруженных процессов или общее их количество;
* Недостаток свободной оперативной памяти на сервере. В этом случае необходимо ограничить ресурсы для контейнера.
* Другие неисправности. В случае нарушения технологического процесса или при длительных отказах технических средств администратор системы обязан сообщить о возникшей проблеме в службу технической поддержки, провести диагностику работы Системы, определить вероятную причину неисправности и передать лог-файлы из соответствующего docker-контейнера. Чтобы связаться с службой поддержки необходимо сообщить о возникшей неисправности по электронному адресу: support@polyanalitika.ru.

1. Рекомендации по освоению

Основным источником информации, используемым при освоении Системы, является данное руководство и документ «Руководство пользователя».

Начинать работу с Системой следует со знакомства с разделами руководства «Подготовка к работе», «Описание операций администрирования» и разделов «Подготовка к работе», «Описание операций» из документа «Руководство пользователя».

Для обеспечения успешной работы пользователям необходимо обладать основными навыками работы с веб-приложениями, опубликованными в сети Интернет.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Лист регистрации изменений*** | | | | | | | | | |
| *Изм.* | *Номера листов (страниц)* | | | | *Всего листов (страниц) в доку-менте* | *Номер доку-мента* | *Входящий номер сопроводи-тельного документа и дата* | *Подпись* | *Дата* |
| *изменен-ных* | *заменен-ных* | *новых* | *аннулиро-ванных* |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |