

[ИТ] Росреестр - Интерчек

- Архитектура
 - Общие сведения
 - Диаграмма
 - Описание
 - Пользователь(User)
 - Системы партнеров(Partner systems)
 - Обратный прокси(Reverse proxy)
 - Прямой прокси(Forward proxy)
 - web-report-service
 - web-sso-service
 - api-gateway-service
 - auth-service
 - application-service
 - temporal
 - debezium
 - Apache Kafka
 - PostgreSQL
 - Opensearch
 - S3/Minio
 - Valkey
- Технический стек
 - Backend
 - Frontend
 - База данных
 - База данных для поиска по тексту
 - Кэш
 - База данных для бинарных данных
 - Шина данных
 - Система выгрузки событий из PostgreSQL
 - Система управления бизнес процессами
 - Система оркестрации сервисов
 - Прокси сервер
- Минимальные технические требования
 - Серверная часть
 - Клиентская часть
 - Браузер
 - Настройки браузера
- Инструкция по запуску системы
 - Минимальные требования
 - Запуск системы
 - Перезапуск системы с удалением данных

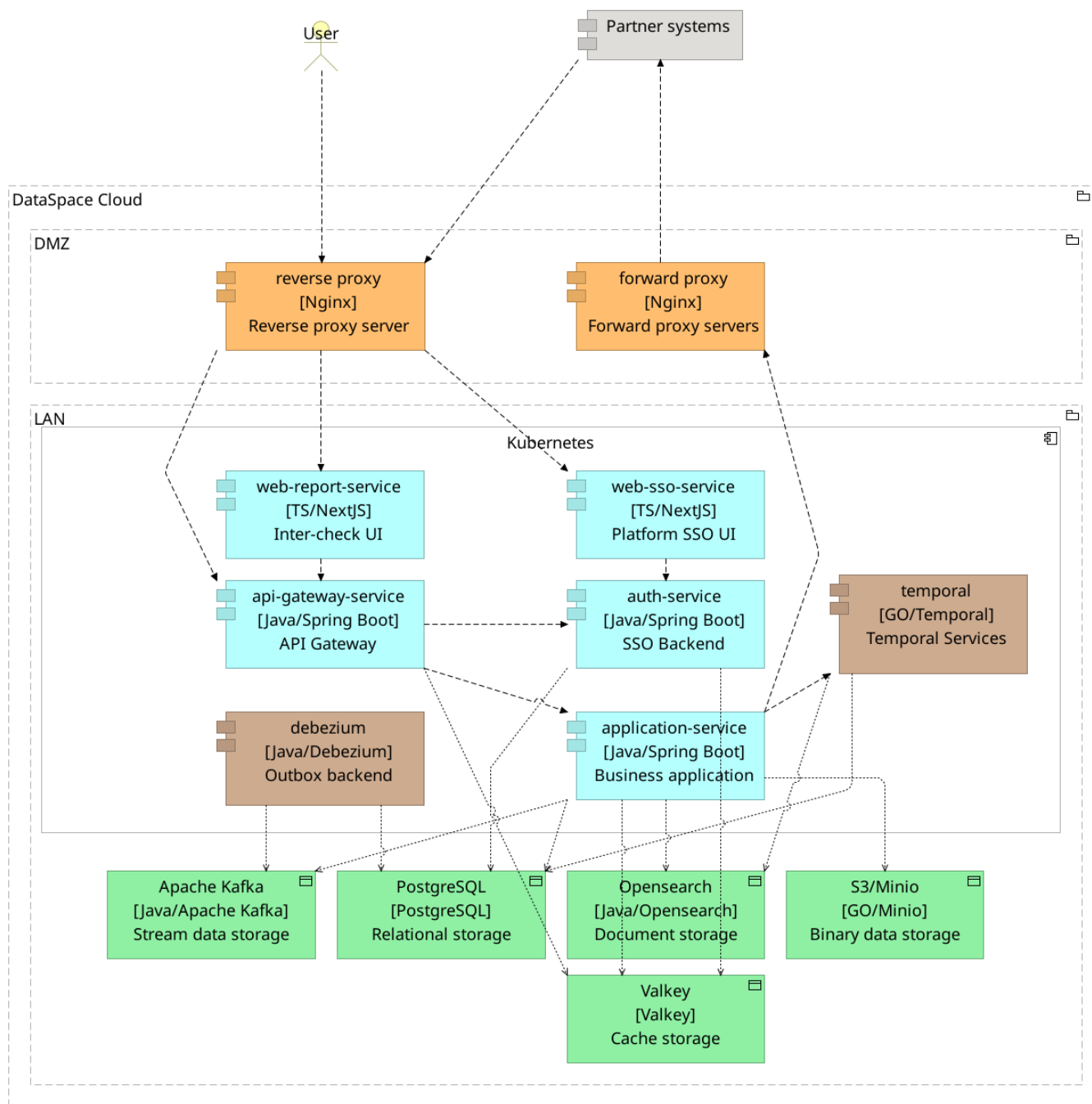
Архитектура

Общие сведения

Интерчек - SaaS, cloud native система получения справок об организациях. Основные функции:

1. Добавление заказов на создание справки.
2. Интеграция с системами партнеров в online и offline режимах для заведения и исполнения заказов.
3. Хранение справок, отображение справок пользователям.
4. Уведомления пользователей, аутентификации/авторизации пользователей.

Диаграмма



Описание

Пользователь(User)

Актор, использующий систему тем или иным образом, заинтересованный в наличии необходимых для него функций системы, например выдачи справок.

Системы партнеров(Partner systems)

Системы, с которыми интегрируется наша система, или которые интегрируются с нашей системой. Например для продукта "Справки" это системы выдачи справок.

Обратный прокси(Reverse proxy)

Отвечает за маршрутизацию входящих запросов из сети Internet во внутреннюю сеть. Используется Nginx.

Прямой прокси(Forward proxy)

Отвечает за маршрутизацию исходящих запросов из внутренней сети в сеть Internet. Используется Nginx.

web-report-service

Отвечает за интерфейс получения справок, просмотра заказов.

web-sso-service

Отвечает за интерфейс системы аутентификации пользователей.

api-gateway-service

Отвечает за маршрутизацию входящих запросов. Помимо маршрутизации проверяет сессию пользователя, обменивает сессию на JWT токены и проверяет авторизацию пользователя.

auth-service

Отвечает за серверную часть системы аутентификации пользователей.

application-service

Отвечает за основную логику приложения. Позволяет управлять данными пользователя, например заказывать и хранить справки.

temporal

Отвечает за управление бизнес процессами, например за процесс восстановления пароля пользователя.

debezium

Отвечает за выгрузку событий из PostgreSQL в Apache Kafka.

Apache Kafka

Отвечает за хранение и асинхронную доставку событий в системы.

PostgreSQL

Основное хранилище данных системы.

Opensearch

Хранилище данных, предназначенное для поиска по тексту. Необходимо для работы Temporal.

S3/Minio

Хранилище данных, предназначенное для хранения бинарных данных, например файлов. Используется для хранения файлов справок.

Valkey

Временное хранилище данных, предназначенное для оптимизации доступа к основной БД PostgreSQL. Снижается нагрузка путем кэширования наиболее часто используемых данных.

Технический стэк

Backend

[Java 21](#)

[Spring Boot 3.3.6](#)

[Jooq](#)

Frontend

[NextJS](#)

[Nx](#)

База данных

[PostgreSQL 16](#)

База данных для поиска по тексту

[Opensearch](#)

Кэш

[Valkey](#)

База данных для бинарных данных

[S3/Minio](#)

Шина данных

[Apache Kafka](#)

Система выгрузки событий из PostgreSQL

[Debezium](#)

Система управления бизнес процессами

[Temporal](#)

Система оркестрации сервисов

[Kubernetes](#)

Прокси сервер

[Nginx](#)

Минимальные технические требования

Серверная часть

1. Компьютер на базе ядра Linux с версией ядра 6.6.+
2. 16 CPU
3. 32 RAM
4. 100 SSD
5. [Docker](#)
6. Сетевая карта

Клиентская часть

Браузер

- Google Chrome - 132.0 и выше;
- Microsoft Edge - 132.0 и выше;
- Яндекс.Браузер - 24 и выше;
- Mozilla Firefox - 135.0 и выше;
- Opera - 76.0 и выше;
- Safari - 18 и выше

Настройки браузера

- В браузере должен быть включен Javascript
- В браузере должны быть включены Cookie

Инструкция по запуску системы

Минимальные требования

Минимальные системные требования описаны в разделе "Минимальные технические требования".

Запуск системы

- Необходимо работать в системе Linux
- Убедиться, что локально запущен docker desktop или совместимые приложения
- Убедиться, что текущий пользователь может работать с docker без sudo
- Все действия ниже производятся в командной строке
- Разархивировать архив Интерчек.zip и перейти в директорию с файлами архива
- Загрузить образ api-gateway-service в docker

```
docker load -i bdp-shared-api-gateway-service.tar
```

- Загрузить образ auth-service в docker

```
docker load -i bdp-shared-auth-service.tar
```

- Загрузить образ application-service в docker

```
docker load -i bdp-cloud-application-service.tar
```

- Загрузить образ web-sso-service в docker

```
docker load -i bdp-web-sso.tar
```

- Загрузить образ web-report-service в docker

```
docker load -i bdp-web-company-report.tar
```

- Запустить сервисы

```
docker compose up -d
```

- Подождать 5-10 минут, до тех пор, пока все сервисы запустятся
- В браузере перейти в <http://localhost>
- Система запущена!

Перезапуск системы с удалением данных

- В командной строке выполнить(удалит контейнеры и данные)

```
docker compose down -v
```

- Запустить систему заново

```
docker compose up -d
```

- Подождать 5-10 минут, до тех пор, пока все сервисы запустятся
- В браузере перейти в <http://localhost>
- Система запущена!