



Факультет
Компьютерных наук

Программная инженерия
3 курс

Москва
2024

Факультет
компьютерных
наук

Образовательная платформа

Командный
программный проект

Выполнил студент 3 курса:
Мохов Сергей Александрович БПИ217

Сокомандники:

Киселев Иван БПИ217

Федоров Артем БПИ217

Научный руководитель:

Приглашенный преподаватель ДПИ
ФКН

Сосновский Григорий Михайлович



Цель работы:

Проектирование архитектуры взаимодействия платформы и разработка инфраструктурного решения с возможностью быстрой интеграции новых сервисов.

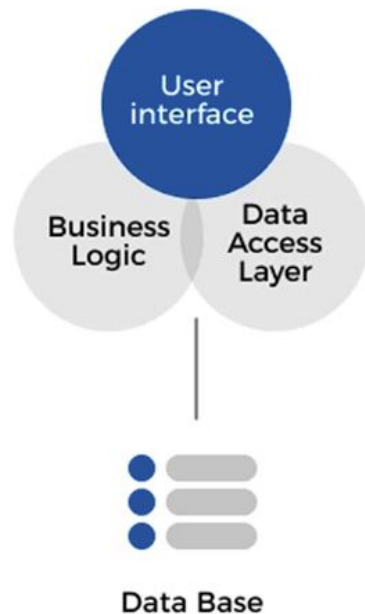
Основные задачи:

- Стандартизация контрактов взаимодействия элементов платформы
- Предоставление единого решения для контроля доступов внутри платформы
- Разработка продуктовых решений для коммуникации с пользователями платформы
- Разработка сервиса отзывов на услуги платформы
- Формирование удобной инфраструктуры для разработчиков платформы

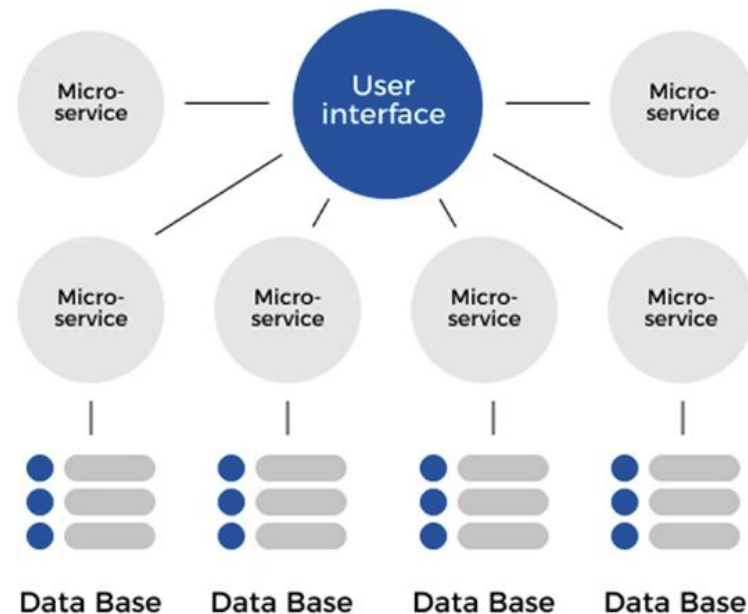


Проектирование платформы или почему мы выбрали микросервисы

MONOLITH ARCHITECTURE



MICROSERVICES ARCHITECTURE

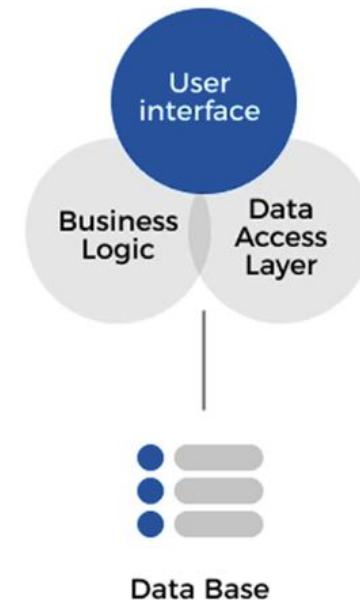




Плюсы монолитной архитектуры

- Простота разработки
- Удобство тестирования
- Мониторинг
- Транзакции
- Легковесная инфраструктура

MONOLITH ARCHITECTURE

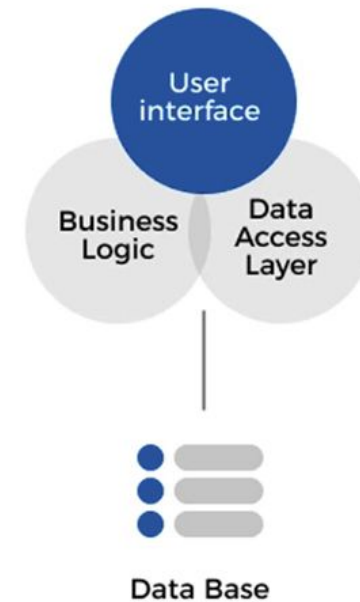




Минусы монолитной архитектуры

- Отсутствие гибкости
- Сложность поддержки
- Однородность масштабирования
- Пониженная отказоустойчивость

MONOLITH ARCHITECTURE

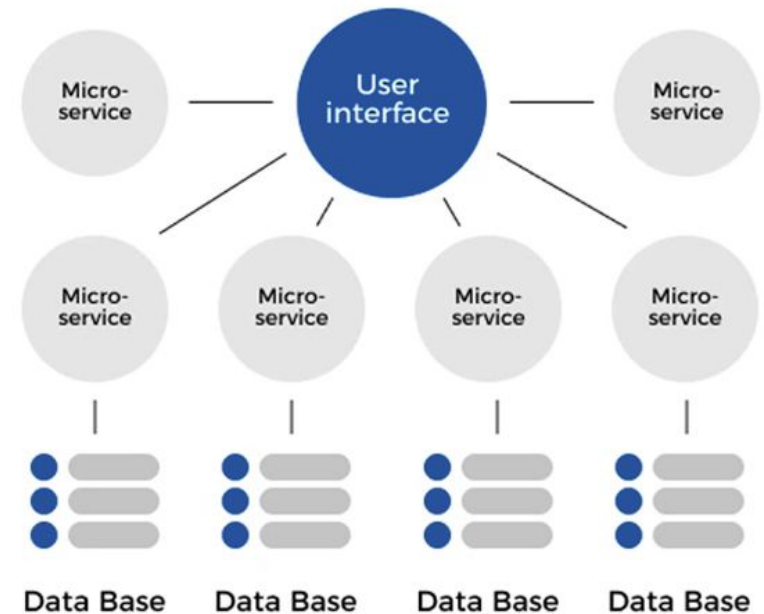




Плюсы микросервисной архитектуры

- Гибкость в выборе технологий
- Независимая разработка
- Повышенная отказоустойчивость
- Производительность
- Масштабируемость

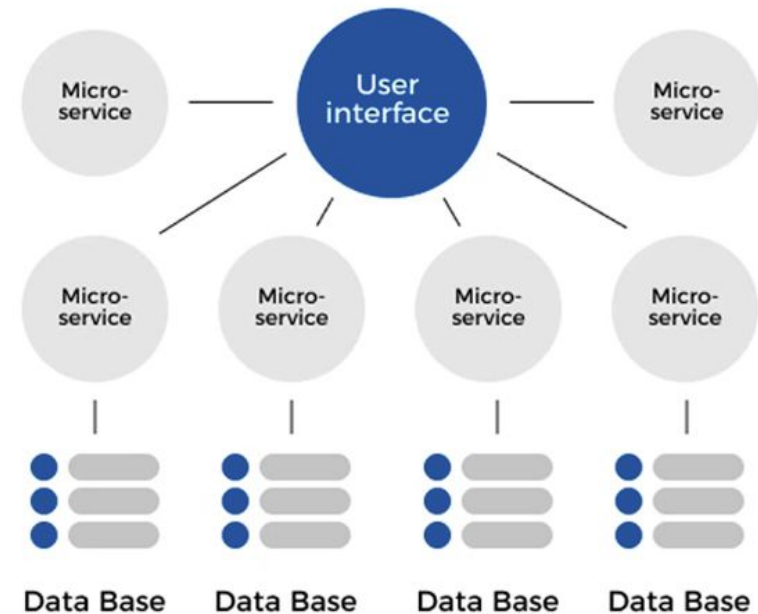
MICROSERVICES ARCHITECTURE



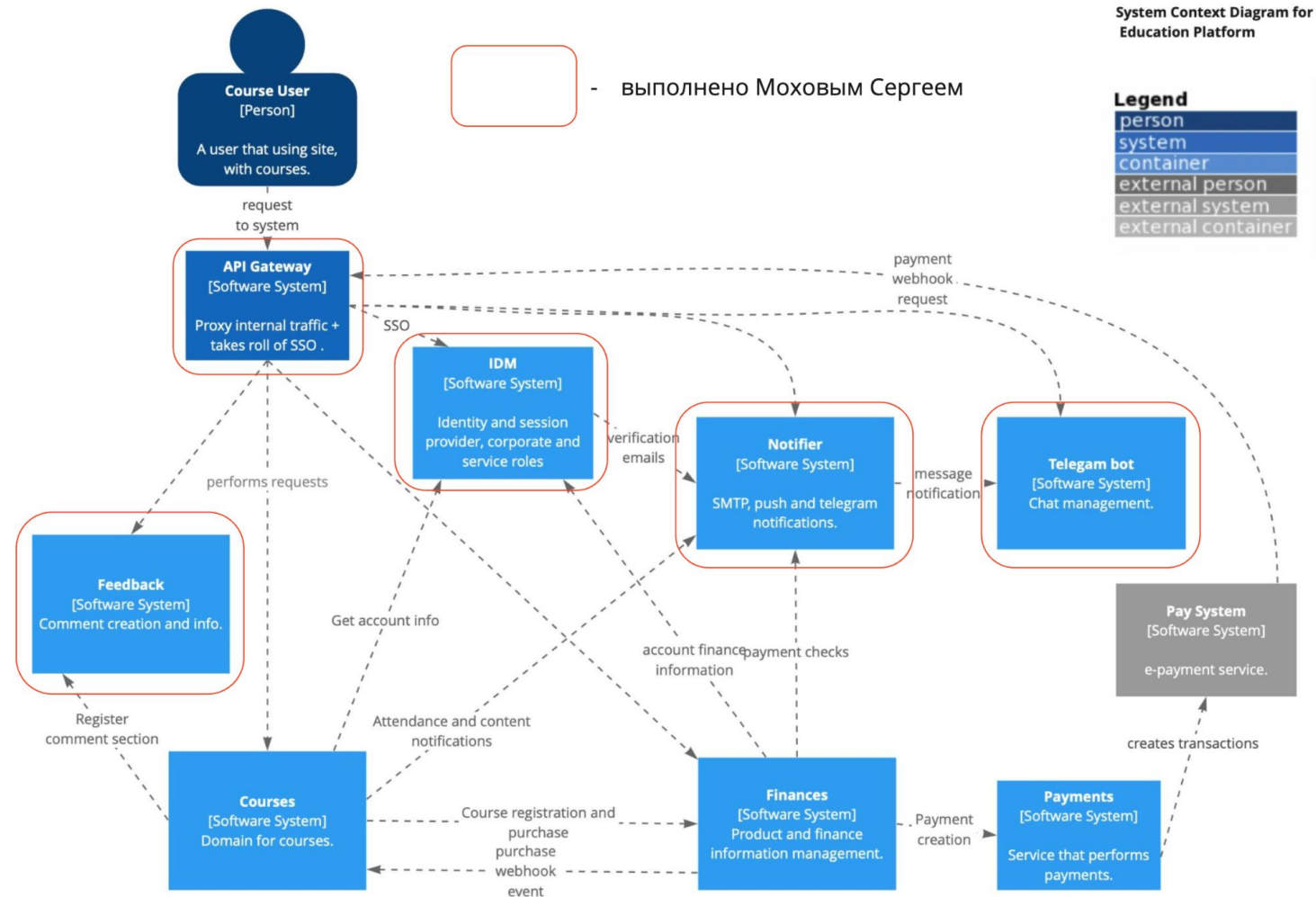
Минусы микросервисной архитектуры

- Сложность синхронизации
- Сложность тестирования
- Повышенные затраты на железо
- Дополнительные затраты на инфраструктуру

MICROSERVICES ARCHITECTURE

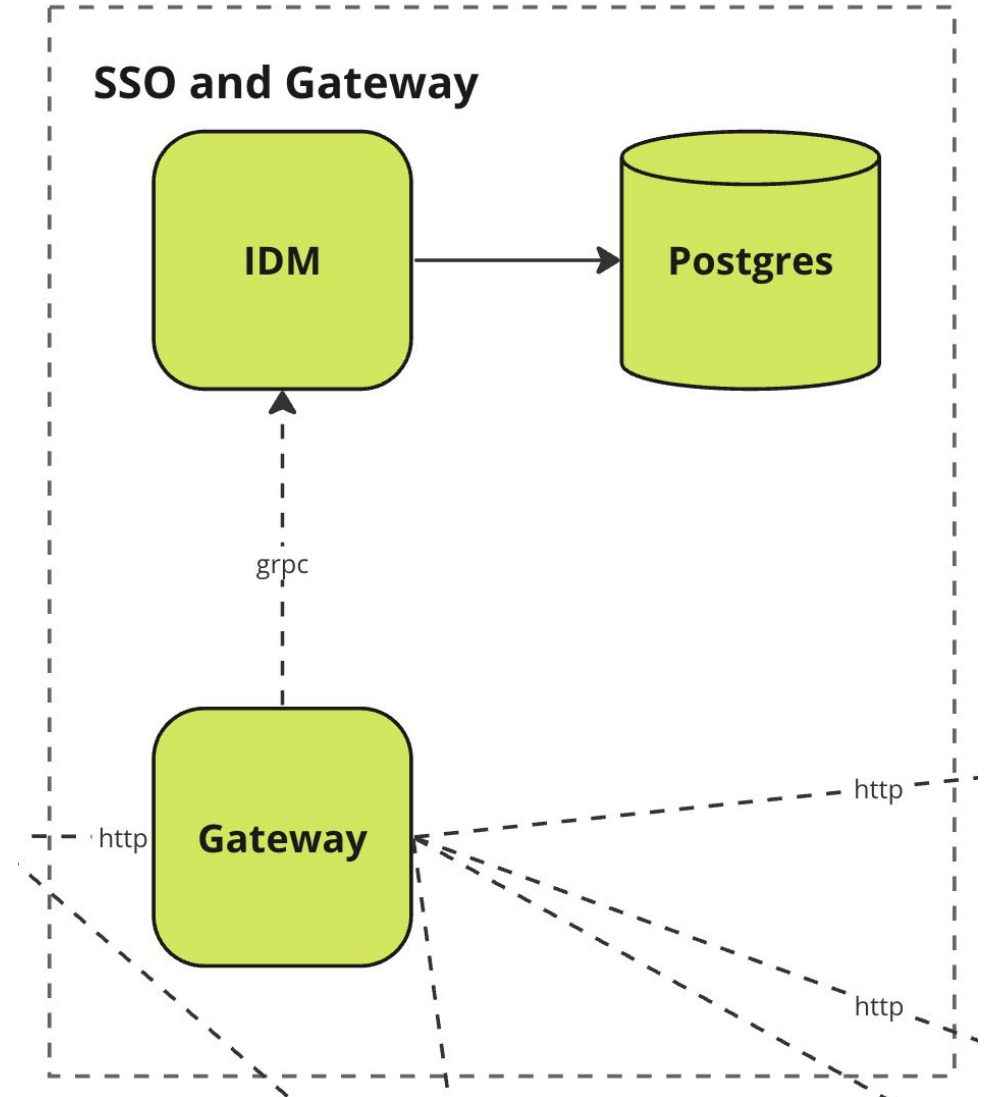


Полученная архитектура платформы



Концепция взаимодействия Gateway и IDM

- Gateway - легковесное решение для валидации и обогащения входящих запросов. Единая точка входа в экосистему платформы
- Gateway гарантирует передачу инфраструктурных данных всем участникам платформы
- Gateway обеспечивает быструю интеграцию новых сервисов
- IDM - главный источник информации о пользователях и их доступах внутри платформы
- IDM предоставляет интерфейс аутентификации и делегирует процесс авторизации Gateway





Конфигурация доступов к сервисам в Gateway

Gateway позволяет с легкостью добавить новый сервис в инфраструктуру платформы. Для изменения доступов к вашему ресурсу достаточно изменить файл конфигурации

```
1 resources:
2   - name: "idm"
3     host: "http://localhost:5758"
4     routes:
5       - pattern: "^/auth/.*"
6         required-roles: [ ]
7       - pattern: "^/account/.*"
8         required-roles:
9           - "PLATFORM_USER"
10      - pattern: "^/admin/account/.*"
11        required-roles:
12          - "IDM_ADMIN"
13   - name: "notifications"
14     host: "http://localhost:8478"
15     routes:
16       - pattern: "^/platform-notification/.*"
17         required-roles:
18           - "PLATFORM_USER"
```




Устройство работы Gateway. Secure запросы

HTTP sage / gateway / pass secure



POST {{BASE_URL}}/api/gateway/secure/pass

Params Authorization Headers (9) Body Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL ☒ JSON

```
1 {
2   ... "access_token": "eyJhbGciOiJIUzI1NiJ9.
      eyJzdWIiOiJmb3hsZXJlbnk5ZW5kZXgucnUiLCJpYXQiOiJlE3MTI4Nzc3NjYsImV4cCI6MTcxMjk2NDE2Nn0.
      wcal0mYu0ewXIOANJEo2gFpIMCCE6hdS0cttpkwmoyY",
3   ... "resource": "notifications",
4   ... "route": "/platform-notification/list?recieverUuid=149f9551-c7b3-4716-bad1-9affdaf5e7b0",
5   ... "method": "GET",
6   ... "body": {}
7 }
```

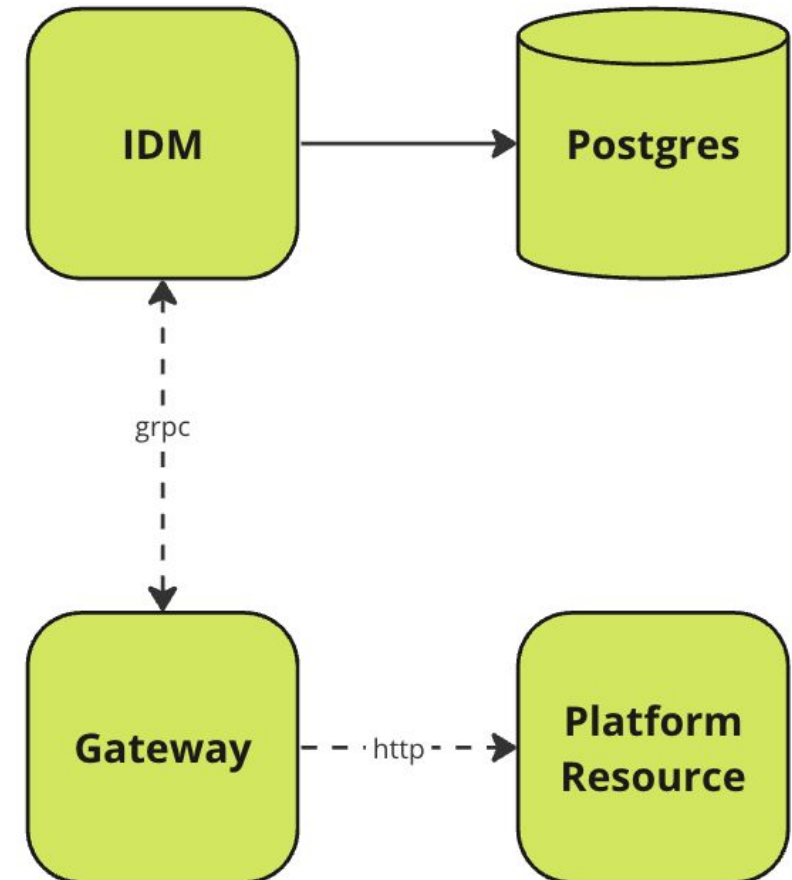
Body Cookies (1) Headers (3) Test Results

Status: 200 OK Time: 7 ms Size: 1.5 KB

Pretty Raw Preview Visualize JSON

```
1 {
2   "status": 200,
3   "body": [
4     {
5       "checked": true,
6       "checkedAt": "2024-03-25T01:20:49.935619",
7       "createdAt": "2024-03-25T01:03:12.462974",
8       "message": "Это тестовое уведомление",
9       "receiverUuid": "149f9551-c7b3-4716-bad1-9affdaf5e7b0",
10      "senderUuid": "149f9551-c7b3-4716-bad1-9affdaf5e7b0",
11      "uuid": "35dfbc77-abab-4c42-ab29-28648b29f7cf"
12    },
13  ]
14 }
```

Secure Request





Единый формат запросов на платформе

Gateway выполняет внутренние запросы к ресурсам предварительно, проставив системные заголовки. Любой сервис внутри инфраструктуры гарантированно получит сведения об авторе запроса

Системные headers:

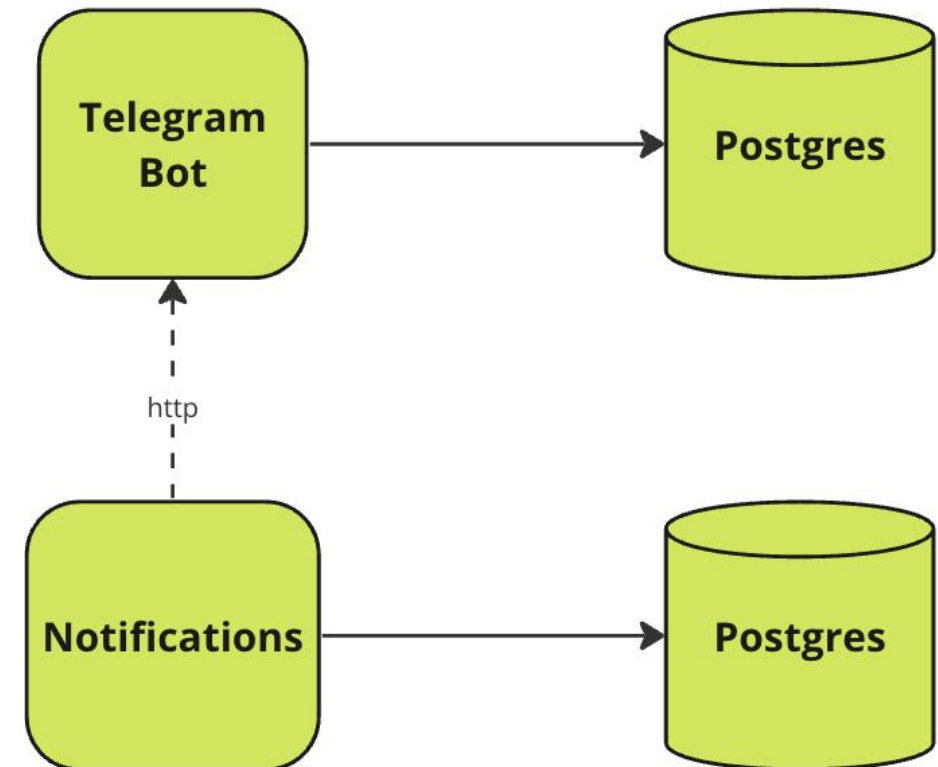
- Xor-Authorization содержит статус авторизации входящего запроса
- Xor-Account-Uuid содержит uuid аккаунта платформы
- Xor-Account-Email содержит email аккаунта платформы
- Xor-Account-Roles содержит роли, выданные на аккаунт



Сервис Notifications

Предоставляет функционал отправки уведомлений пользователям платформы различными способами. Также реализует в себе сервис пуш-уведомлений на платформе

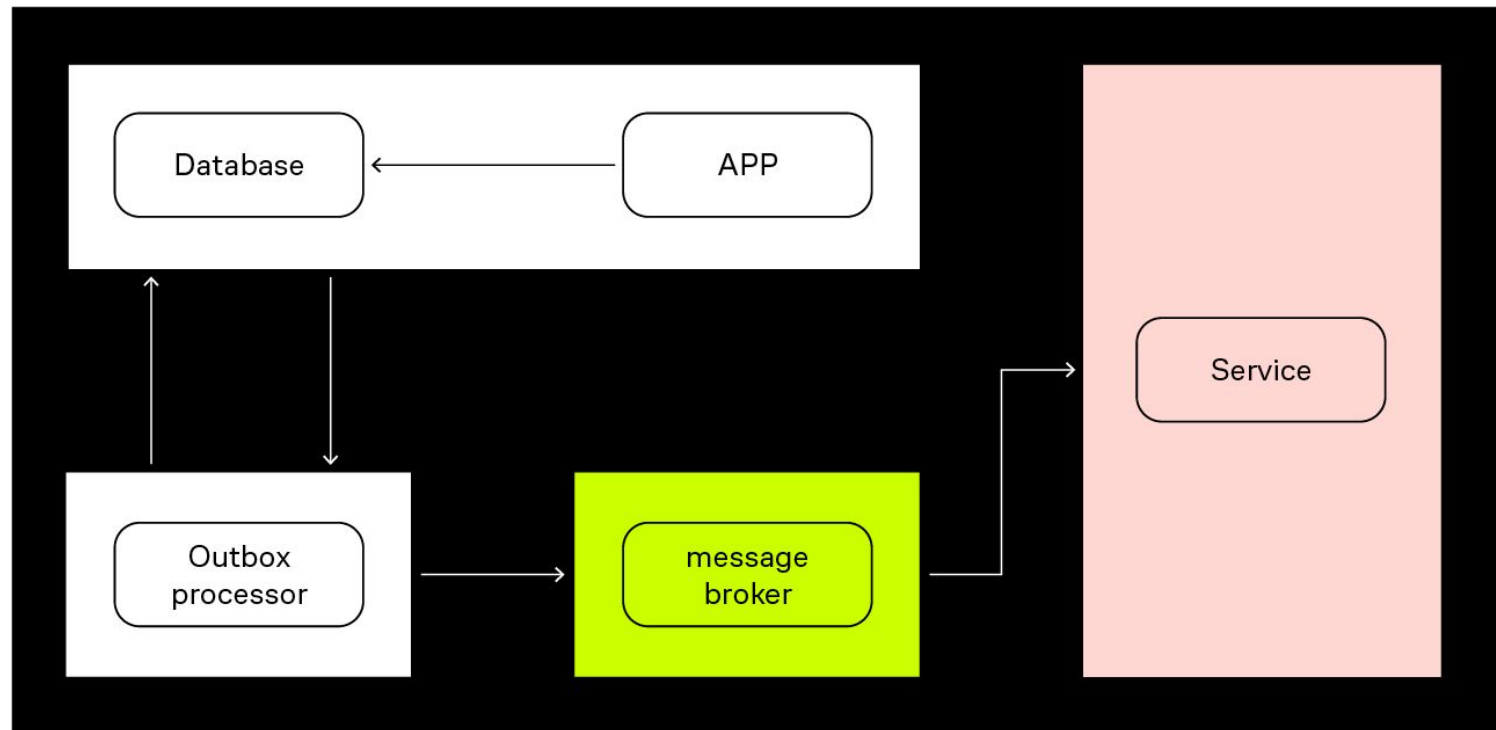
- Отправка письма на почтовый адрес
- Отправка пуш-уведомления на платформу
- Отправка сообщения в Telegram





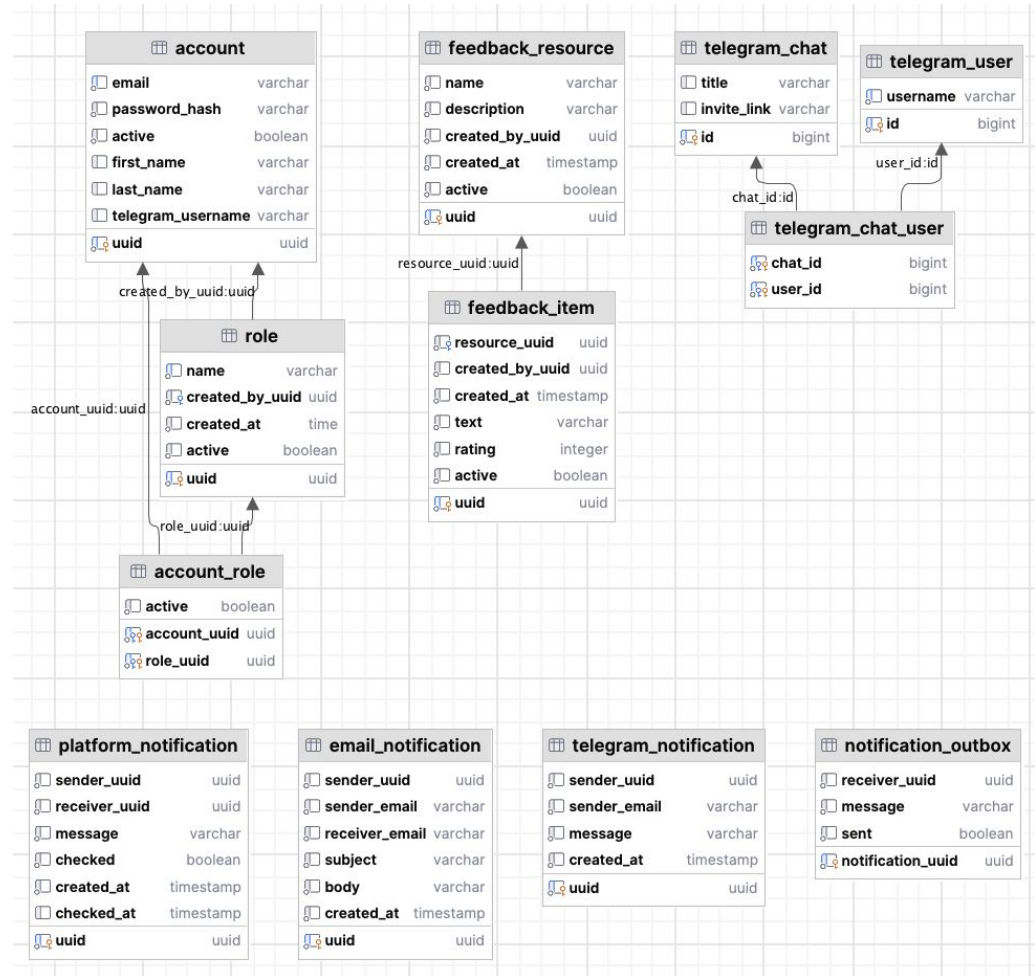
Реализация Outbox паттерна в Notifications

Сервис Notifications выполняет резервное копирование всех полученных запросов на уведомление и регулярно пытается отправить уведомления в случае недоступности сервисов-партнеров





Представление данных в БД





Доступное Аpi

- ▼ bot
 - > POST send-message
 - > POST block
- ▼ feedback
 - ▼ resource
 - > GET list
 - GET stats
 - > POST create
 - > PUT update-info
 - > PUT set-active
 - ▼ item
 - > GET list
 - > POST create
 - > PUT update-info
 - > PUT set-active

- ▼ idm
 - ▼ auth
 - > POST register
 - > POST log-in
 - ▼ account
 - > GET list
 - > PUT reset-password
 - > PUT update-info
 - ▼ admin/role
 - > GET list
 - > POST create
 - > PUT set-active
 - > POST assign role
 - > POST revoke
 - > PUT set-active
 - ▼ admin/account
 - > PUT update-password
 - GET get roles

- ▼ notifications
 - ▼ platform-notification
 - > GET list
 - > POST create
 - > PUT check
 - ▼ email-notification
 - > GET list
 - > POST create
 - ▼ telegram-notification
 - > POST create
- ▼ sage
 - ▼ gateway
 - > POST pass secure
 - > POST pass insecure



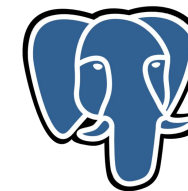
Мониторинг





Используемые технологии

- Gateway: Golang. Взаимодействие по http и grpc
- IDM: Java. Взаимодействие по http и grpc
- Notifications, Feedback: Java
- Bot: Python



PostgreSQL



Prometheus



Дополнительные замечания

- Проектирование сервисов выполнено в парадигме “Чистой архитектуры”
- Код содержит защиту от sql инъекций
- В IDM добавлена поддержка jwt токенов
- В монорепозитории настроен github ci
- Написаны скрипты для сборки .proto моделей под разные языки



Результаты

- Реализована инфраструктура для контроля доступов в сервисы платформы
- Реализован единый стандарт передачи данных сервисам платформы
- Процесс интеграции нового сервиса занимает минимальное время
- Реализованы продуктовые сервисы для уведомлений пользователей и оставления отзывов на услуги
- Реализована инфраструктура в репозитории для сборок проектов и их образов



Планы на будущее

июнь

Перейти на использование k8s. Добавить поддержку Vault

июль

Расширить функционал Gateway и IDM. Покрыть тестами и метриками критические участки кода

август

Написать инфраструктуру для деплоя на сторонние серверы

Демонстрация



Спасибо за внимание!



С радостью отвечу на ваши вопросы!