

Тема: «Сервис автоматического код-ревью»

Группа: ПИБД-41

1. Реализация всех задумок СРО в виде продуманной архитектуры проекта.

В качестве архитектуры приложения мною была выбрана микросервисная архитектура, потому что она имеет следующие преимущества:

- Гибкий график релизов (каждый сервис имеет свой график релизов)
- Легкий порог вхождения в разработку отдельно сервиса (не придется изучать структуру всего приложения в целом, достаточно погрузиться в предметную область одного сервиса, сервисы между собой изолированы и взаимодействуют через четко определенные контракты)
- Возможность горизонтального масштабирования
- Сервисы могут быть написаны на разных языках

Микросервисная архитектура имеет свои издержки, связанные с DevOps, но в перспективе они окупятся.

На диаграмме ниже указаны следующие компоненты:

- PostgreSQL – реляционная СУБД
- Сервис пользователя – микросервис, отвечающий за процессы аутентификации, авторизации, хранения настроек пользователя.
- Сервис обработки событий – микросервис, принимающий по WebHook события от git-хостингов (GitHub, GitLab, BitBucket, etc), который фильтрует и преобразовывает их к единому формату, а затем отправляет их в шину данных
- Сервисы обработчики кода – набор типовых микросервисов, принимающих из шины данных файлы исходного кода с метainформацией и отдающих результат анализа обратно

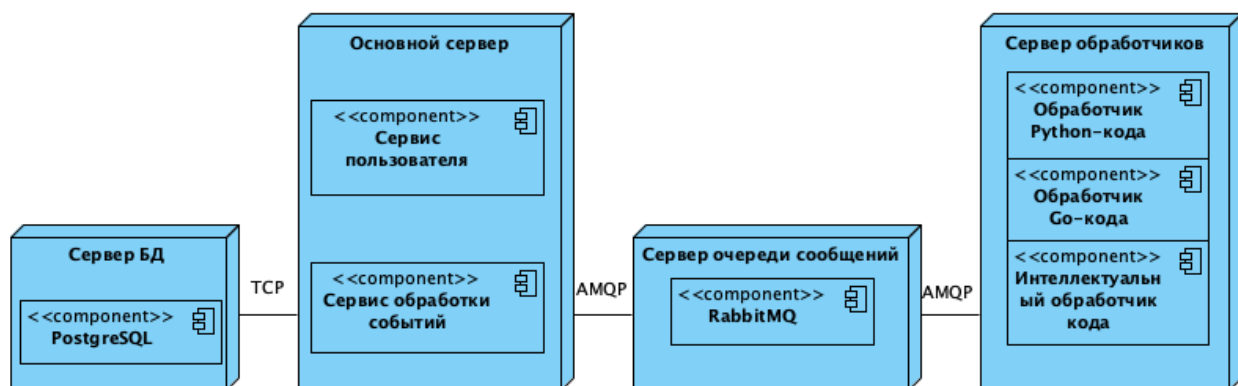


Рисунок 1. Диаграмма развертывания

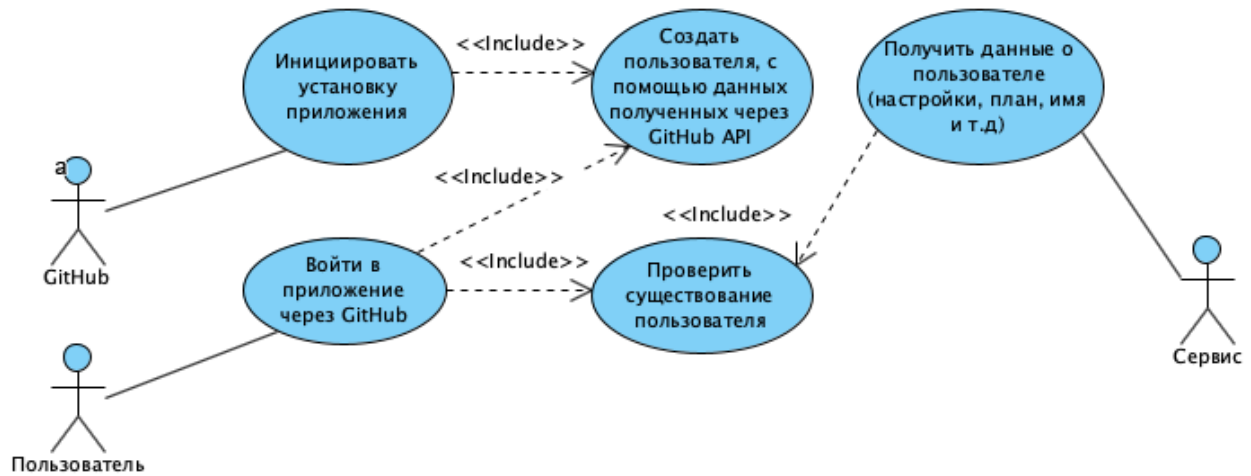


Рисунок 2. Use-Case диаграмма сервиса пользователя

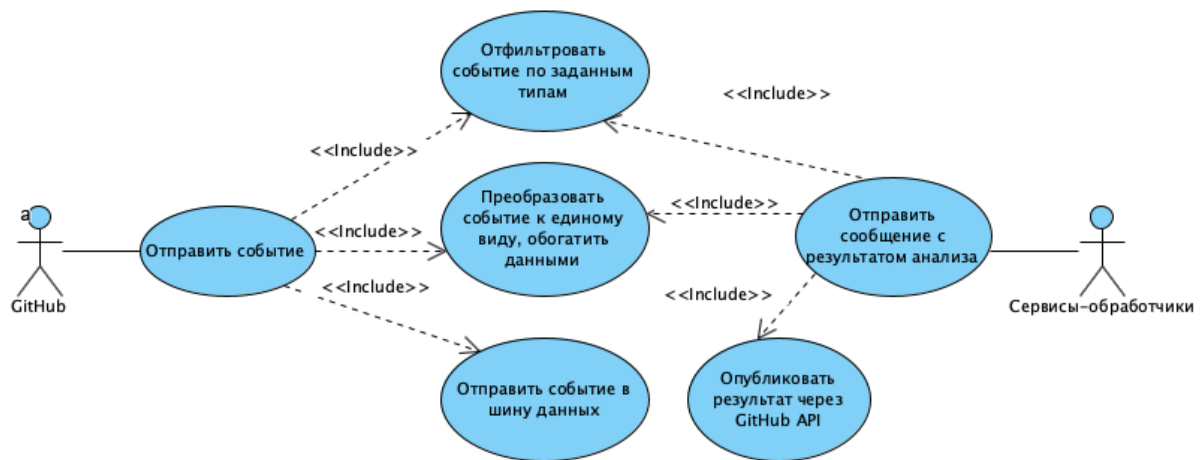


Рисунок 3. Use-Case диаграмма сервиса обработки событий

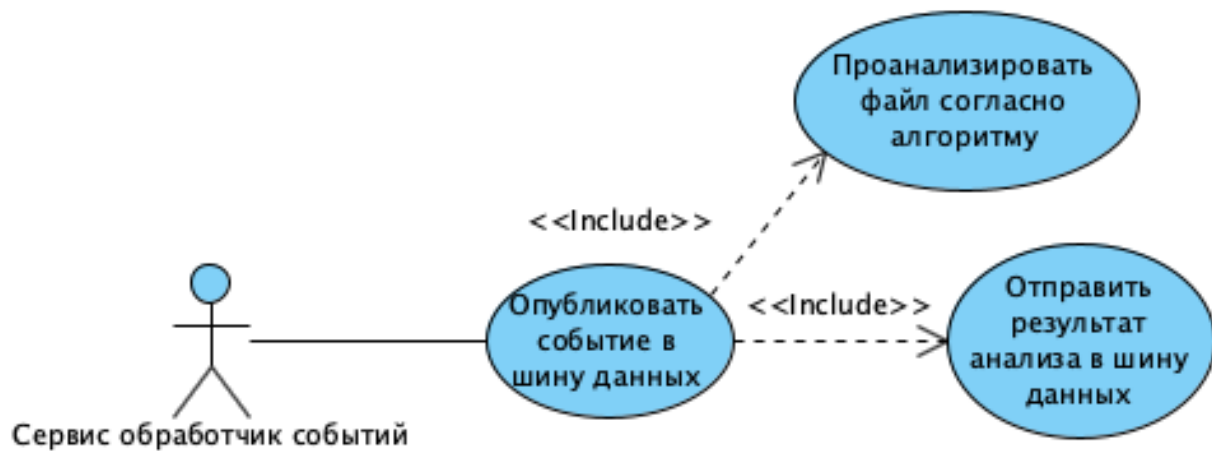


Рисунок 4. Use-case диаграмма сервиса-обработчика кода