|  |
| --- |
| **Ксения Сидельникова** |
| **QА Аutomation Еngineer Senior**  **Опыт работы 5,5+ лет** | |
|  | |
|  | |

**QА Аutomation Еngineer**

**(Июнь 2024 – Настоящее время)** 2ГИС

**Описание**: ETL процессы, потом перешли на ELT, только Backend.  
Сбор и доставка данных в Экспорт.  
Данные, необходимые для конечных продуктов (гео и справочные данные), собираются из разных источников, преобразуются и сливаются в единый источник, с которым в последствии работают сервисы Экспорта.  
Собираются как из источников компании (команды добычи данных или команды, которая хранит и обрабатывает упорядоченные объекты карты).  
Сторонние источники, такие как OSM (OpenStreetMap) и Wikipedia – используется для наполнения справочными данными.

Итоговым результатом, с которым далее работают сервисы Экспорта данных/Поиска и другие, является:  
– база данных объединенных данных;   
 – API Service для удобной работы с геоданными, справочниками.   
Есть также разные сервисы, отвечающего за формирование потока данных  
  
Матчинг данных:  
Сервисы, позволяющие выполнять сравнение данных из различных источников, с настраиваемыми правилами сравнения. Обычно для сравнения используется информация из внешнего источника и информация компании.  
  
Результаты матчинга используются для актуализации и обогащения данных в базах 2ГИС.   
Источники для матчинга:  
  
Feed - источник данных, поставляемый на регулярной основе (например, feed от RocketData и Pointer, сетевых компаний)  
Результаты парсинга данных из открытых источников (Желтые, Оранжевые, Синие).  
Разовые файлы с данными.  
  
Поддерживаемые виды матчинга:  
Матчинг организаций;  
Матчинг зданий (гео-объектов);  
Ключевой сервис – Groot.

**Основные обязанности**:

1. Тестирование новых сервисов и доработки старых.
2. Написание junit тестов, интеграционных тестов, e2e.
3. Написание автотестов на Kotlin.
4. Проведение ревью разработчиков и тестировщиков.
5. Взаимодействие с командой разработки и менеджментом.
6. Обеспечение качества и отсутствие критичных багов, совместно с командой тестирования обеспечивали качество всего продукта.

**Технологический стек**: Kotlin, Kafka, Postgresql, API Сlickhouse, bash, Grafana, Кibana, Junit5, Allure, Gitlab, Gitlab ci, Test containers, http, Mock server, Mock, ETL, Airflow

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**QА Аutomation Еngineer**

**(Июль 2022 – Май 2024)** QIWI Ltd. (Цифровой рубль)

**Описание**: Общие микросервисы для контроля лимитов и правильности списания средств, цифровой рубль, СБП.   
Микросервисы.  
Для интеграции с цифровым рублем было два основных сервиса. Контроллер, который по сути дела выступал как прокси, и процессинг, там, где была основная логика. Процессинг также отправлял сообщения в цифровой рубль.   
Цифровой рубль отправлял сообщения в kafka, процессинг вычитывал их из kafka, складывал в базу данных, и потом можно было получить их из БД по ID.   
Основная сложность интеграции с цифровым рублем состояла в сертификатах, подписях и с запаковкой конверта.

Команда пользовалась своим сервисом. Эта система хранения сертификатов, там же были методы для подписей, расшифрования, запаковки в конверт.  
ЦБ присылал основные сценарии, которые нужно было пройти, и предоставить отчет.   
Сложность была в том, чтобы выпустить сертификаты, при этом там должны были быть реальные данные человека с паспортными данными. Команда автоматизировала тест-кейсы покупки, продажи цифрового рубля, получения баланса кошелька, переводов  
  
**Основные обязанности**:

1. Разработка автотестов для backend микросервисов (интеграционные и e2e) на Kotlin, Java, Groovy, а также рефакторинг и поддержка старых тестов.
2. Тестирование API (автотесты на Java).
3. Разработка нагрузочных тестов (gatling, scala).
4. Настройка ci/сd и встраивание тестов в пайплайн, а также сбор метрик тестового покрытия (teamcity, jacoco).
5. Организация процессов автоматизации тестирования.
6. Взаимодействие с командой разработки и менеджментом.

Достижения

1. Актуализировала нагрузочные тесты.
2. Добавила сбор покрытия интеграционными тестами, и следила, за тем, чтобы эта метрика была не ниже 80 %.
3. Совместно с Lead организовали процессы тестирования для нового проекта.
4. Автоматизировала Postman коллекцию и добавила эти тесты в уже готовый фреймворк, а также отрефакторила и улучшила старые тесты.
5. Взаимодействие с командой разработки и менеджментом

**Команда:** scrum команда 6 человек (Team Lead, разработчики и один QA)

**Технологический стек**: Kotlin, Java, Groovy, gatling, scala, teamcity, jacoco, kafka

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**QA Automation engineer (Kotlin)**

**(Апрель 2021 – Июнь 2022)** Банк Открытие

**Описание**: Мобильное банковское приложение. Основано на технологии мастер кард - decoupled.  
Команда, в которой работал специалист: карты, лояльность.  
Создание карты банка, возможность привязать карты других банков к ней, Создание общего кошелька. Что позволяло отслеживать аналитику по расходам с карт разных банков, Разные действия с картами: выпуск, блокировка и т.д.  
Интеграция с Госуслугами. Большое количество микросервисов

**Основные обязанности**:

1. Совместно с Lead выстроили процессы тестирования и автоматизации, разрабатывали тестовый фреймворк, далее вместе с растущей командой формировали и улучшали подходы.
2. Обеспечение качества и отсутствие критичных багов, совместно с командой тестирования обеспечивали качество всего продукта.
3. Разработка тест-кейсов, их автоматизация и поддержка (покрытие около 100% с успешным прохождением 95%).
4. Анализ спецификаций.
5. Оценка трудоемкости и сроков выполнения работ в части тестирования.
6. Тестирование backend (smoke, функциональное, регрессионное, повторное, интеграционное).
7. Тестирование API (автотесты на Kotlin).
8. Проведение код-ревью.
9. Взаимодействие с командой разработки и менеджментом

**Команда:** Отдел backend разработки (30 человек), отдел backend тестирования (12 человек), продуктовая команда (12 человек).

**Технологический стек**: Kotlin, PostgreSQL, Kafka, Kotest, Junit, bash, Кibana, API, Gradle, Docker, Redis, Wiremock, Swagger, HTTP, Rest, GitHub, Linux, Json, GraphQL, Scrum, Allure.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**QA Automation engineer (Java)**

**(Июнь 2019 – Апрель 2021)** Альфа Банк

**Описание**: MDM система (cбор данных из различных банковских систем, упорядочивание, предоставление этих данных через сервисы).  
Микросервисы  
  
**Основные обязанности**:

1. Совместно с Lead выстроили процессы тестирования и автоматизации, разработали тестовые фреймворки.
2. Вместе с командой обеспечивали качество продукта и отсутствие критичных багов.
3. Разработка тест-кейсов, их автоматизация и поддержка (покрыли около 70% с успешным прохождением более 95%).
4. Анализ требований к системной архитектуре компонентов ИТ-Продукта.
5. Написание автотестов на Java.
6. Оценки трудоемкости и сроков выполнения работ в части тестирования.
7. Проведение код- ревью.
8. Тестирование backend (smoke, функциональное, регрессионное, повторное, интеграционное тестирование)

**Команда:** 12 человек, backend разработка микросервисов (карты, аккаунты, платежи и переводы).

**Технологический стек**: Java 11, PostgreSQL, Gitlab, Docker, API, RabbitMQ, Swagger, bash, Kibana, Linux, Postman, Testrail, Rest, Junit, WireMock, JOOQ, Gson, SpringBoot, Scrum, Kanban, SoapUI, MS SQL, Bitbucket, XML, HTTP, Json, Selenium, GitLab.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Образование**

|  |  |
| --- | --- |
| **2016** | Санкт-Петербургский государственный институт психологии и социальной работы |

**Ключевые навыки**

Java, Kotlin, Groovy, PostgreSQL, bash, Кibana, Kafka, MS SQL, Soap UI, Postman, Swagger, Docker, Git, HTTP, XML, Grafana, Linux, Intellij IDEA, Teamcity, Jacoco, Gatling, ETL, Airflow, Scala, Atlassian Confluence, Scrum, Atlassian Jira, Kotest, Junit, API, Gradle, Redis, Wiremock, Rest, GitHub, Json, GraphQL, Allure, RabbitMQ, Testrail, JOOQ, Gson, SpringBoot, Kanban, Selenium, Bitbucket

