

INNOPOLIS
UNIVERSITY



AQA: кто это и чем занимается

Цели:


1. Узнать, кто такой AQA
2. Познакомиться со сферой автоматизации тестирования
3. Обозначить карьерное развитие AQA

Знакомство

Напишите в чат или расскажите голосом 3 пункта о себе:

1. Ваш опыт в IT.
 - a. Кем и сколько работали?
 - b. Что знаете?
 - c. Что умеете?
 - d. Опыт в тестировании
 - e. Опыт в программировании? На чем писали? Какие задачи решали?
2. Зачем пришли на курс?
 - a. Карьерный рост
 - b. Смена отрасли
 - c. Смена направления
 - d. Пришел сюда, потому что нетфликс больше не работает
 - e. Была возможность сходить за счет работодателя
 - f. Еще вариант
3. На курсе я бы хотел(а) узнать больше про...
 - a. Инструмент
 - b. Технологию
 - c. Подход
 - d. Не пишите, пожалуйста, “всего и по-больше” :) если не знаете, что ответить, придумайте самый абсурдный вариант :)

Правила игры

- 
1. **Вопросы – наше все.** Чем больше вы задаете вопросов, тем лучше. Всем лучше. Вопросы можно задавать в чат урока и чат Тг. Тупых вопросов не бывает, а вот упущенные возможности – да.
 1. После каждого урока будут **задания “для любопытствующих”**. Много. Необязательные домашки, которые можно/нужно делать, чтобы закрепить пройденное.
 2. **Практикуйтесь.** Иначе будет очередная книжка по карате.
 3. **Обсуждайте.** Учебный чат в Тг создан не для меня :) В чате сидят ваши коллеги, которые решают те же задачи, что и вы и сталкиваются с похожими проблемами. Вместе находить решение – одна из приятных частей нашей работы.
 4. **Будет больно.** Это признак роста.
 5. Иногда мы будем отклоняться от программы. При этом, мы сохраним все темы. И мы сохраним весь объем знаний, который должны были получить. Просто пойдем немного в другом порядке.

QA engineer vs AQA engineer

Quality Assurance – обеспечение качества
Quality Control (QC) – контроль качества

QC engineer контролирует качество готового продукта.

QA engineer обеспечивает качество продукта на всех этапах его ЖЦ.

AQA/QAA engineer автоматизирует процессы обеспечения качества продукта.

ЖЦ продукта

Жизненный цикл программного обеспечения (Software Life Cycle Model) — это период времени, который начинается с момента принятия решения о создании программного продукта и заканчивается в момент его полного изъятия из эксплуатации.

Основные этапы ЖЦ ПО:

- анализ требований,
- проектирование,
- кодирование (программирование),
- тестирование и отладка,
- эксплуатация и сопровождение.

AQA – это не только авто-тесты

- Автоматизация подсчета тестового покрытия;
- Автоматизация выгрузки отчетов;
- Написание фреймворков для тестирования;
- Написания вспомогательных программ для ручных и авто-тестирующих;
- Настройка инфраструктуры для проведения авто-тестирования.

Написание авто-тестов



Пирамида тестирования:

- Выделяют 3 уровня тестов. На каждом уровне свои задачи
- Нужно ли писать тесты всех уровней?

Инструменты автоматизации



- **JUnit** — фреймворк автоматизации тестирования для Java
- **IDE** — это интегрированная среда разработки, в которой можно писать, компилировать и отлаживать код.
- **REST Assured**. Java-библиотека для тестирования REST API.
- **Selenium WebDriver** — предоставляет автотестам доступ к браузеру

Только ли AQA пишут тесты?



Автоматизацией тестирования занимается вся команда, но AQA отвечает за качество продукта в целом и должен быть в курсе всего, что автоматизируется.

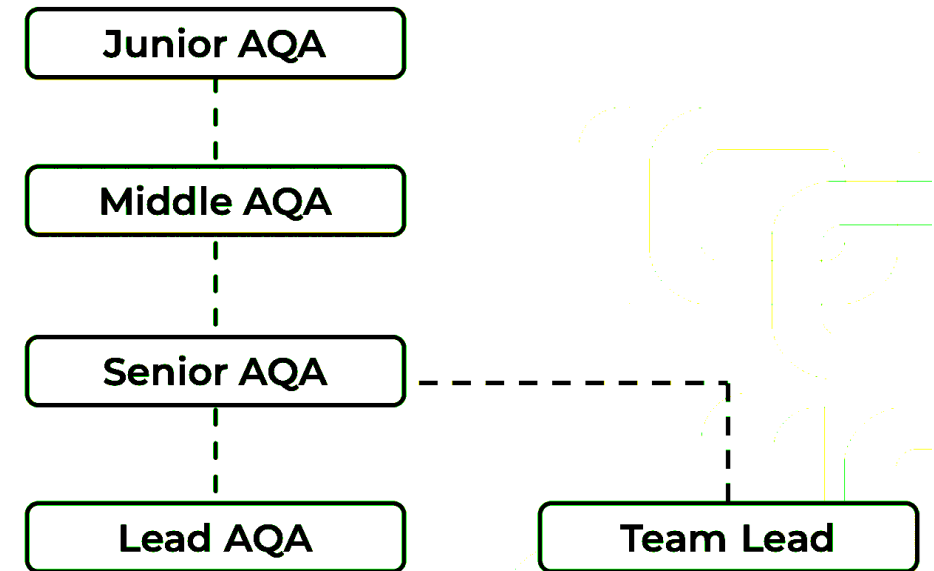
- Юнит-тесты пишут разработчики
- API- тесты пишут разработчики и тестировщики
- E2E-тесты пишут тестировщики

Как перейти в AQA из QA

1. Знать основы разработки и владеть хотя бы одним ЯП на начальном уровне.
Топ-3: Java, Python, JS/TS
2. Знать git на начальном уровне и основы работы в IDE.
3. Понимать, как работают клиент-серверные приложения.
4. Хорошо владеть ручным тестированием.

Будет плюсом:

1. Иметь учебный проект с авто-тестами в портфолио.
2. Знать английский язык на уровне чтения документации.



Примеры вакансий AQA:

Обязанности:

- Реализация UI автотестов на существующем проекте автотестирования;
- Анализ требований и составление тест кейсов, принятие активного участия в обсуждении задач и внесении изменений в работающий функционал;
- Ручное и автоматическое прохождение тестовых сценариев;
- Документирование и отслеживание ошибок в баг-трекере работы с Jira, Confluence;
- Разработка тестовых систем/заглушек (эмуляторы внешних сервисов);
- Работа в кросс-функциональной команде, налаживание эффективных коммуникаций;
- Участие в разборе инцидентов и багов с production;
- Помощь второй линии, а также участие в 3-й линии поддержки;
- Автоматическое тестирование API с помощью Postman и Rest Assured;
- Тестирование API с помощью специализированных инструментов SoapUI, Postman.

Примеры вакансий AQA:

Должностные обязанности:

- Разработка и тестирование программных продуктов в системе IBM "System i" (старые названия AS/400, iSeries), **языки RPG IV, C, Sql**(СУБД DB2)
- Разбор и анализ отчетов о тестировании вместе с командой разработки
- Поддержание работоспособности автотестов с учетом изменений в тестируемых программах
- Расчет метрик покрытия кода автотестами

Требования:

- Высшее, неполное высшее техническое образование
- Умение писать несложные запросы на языке SQL
- Опыт разработки с использованием языков программирования высокого уровня (C (предпочтительно), C++, Pascal, **Java** и др.)
- **Английский – на уровне чтения технической документации** (вся документация по проекту на английском языке)

Примеры вакансий AQA:

Обязанности:

- Подготовка тестовых сценариев;
- Автоматизация тестирования с использованием Cucumber-Java, Selenide, Selenoid, Akita, RestAssured;
- Доработка фреймворка для тестирования;
- Проведение регрессионного, интеграционного тестирования;
- Сопровождение UAT;

Требования:

- Опыт в тестировании систем на микросервисной архитектуре от 3х лет;
- Опыт автоматизации тестирования от 3 лет;
- Практический опыт применения практик тест-дизайна;
- Умение проектировать тест-кейсы и подготавливать тестовые данные;
- Знание инструментов автоматизации тестирования Web приложений - Java, Selenium/Selenide, Cucumber, RestAssured;
- Опыт тестирования REST сервисов;
- Опыт работы с системой версионного контроля git (бранчи, пулл-реквесты и тд.);
- Опыт работы с системами CI;

Задание на дом:

1. Поставить IDEA (Community!)
2. Установить JDK на ваш компьютер
3. Создать простой проект и проверить, что он работает (пример показываю)
4. Разобраться, что такое системная переменная Path.
 - a. Зачем нужна?
 - b. Как ее посмотреть на вашем компьютере?
 - c. Как ее настроить в вашем компьютере?

INNOVATION
UNIVERSITY



Вопросы?