QA Automation evolution.
The new SEiT role breaks borders and limitations.
Serhii Romaniuk

- 6+ years of professional experience in IT industry specializing in Quality Assurance;
- Thorough experience in manual and automated testing and setup of Quality Assurance processes;
- Current areas of professional interest:
  - Quality Assurance
  - Project Management
  - Team Leadership
- Company: DataArt Poland
Projects parameters and QA setup

Factors that affect QA setup, efficiency and cost of quality:

- Project size and pace
- Team size and level of maturity
- Technological complexity
- Etc. etc.
Extensive QA team setup
Switch from Black-box to White-box testing
Team with SEiT

Software Engineer in Testing coordinates all QA activities
Key principles and concepts

1. Software Engineer in testing instead of Manual testers/QA automation engineers
2. "T-Shaped" developer
3. Shift-left testing approach
4. Three Amigos
5. Testing Pyramid
6. Definition of done
Role description:

As a Software Engineer in Test, you have the opportunity to accelerate the delivery and to improve the quality of product. You will be responsible for designing and implementing development and test infrastructure. You will be a part of an Engineering team, which means that you may participate in all discussions and decisions. Your focus is organisation of the testing process on all stages of development, developing code of the testing framework and tools rather than finding bugs. Your job is to accelerate product development by helping developers help themselves.
SEiT activities in SDLC

- **UX Research User Testing**: Objective: Are we building the right thing? How?: Understand customer point of view.
- **Inception Epic**: Objective: Are we building the right thing? How?: Need to understand the dependencies and the scope of the project to assess the risk.
- **Definition User Stories**: Objective: Are we building the right thing? How?: Help on having clear requirements for a user story.
- **Support**: Assist team in understanding current functionality of the app.
- **Backlog Grooming**: Objective: Are we building the right thing? How?: Agree with team on acceptance criteria.
- **Release**: Objective: Helping dev team in testing activities. How?: Preparing the test environment needed for testing. We DON'T decide what and when to release.
- **Task in Next**: Objective: Are we building it right? How?: Agree with dev on how it will be tested (non functional/UI automated test, Regression test).
- **Task in Validation**: Objective: Helping dev team in testing activities. How?: Helping the team on executing functional or UI tests when needed.
- **Task in Code Review**: Objective: Are we building it right? How?: Creating Cases in TestRail. Creating API tests if applicable.
- **Task In Progress**: Objective: Are we building it right? How?: Having our API tests reviewed by devs and vice versa. (Nice to have: Reviewing UI tests).
- **Monitoring Test Environment**: Objective: Helping dev team in testing activities. How?: Communicate when suite is failing because of a change in backend or environment.
- **Reports from Regression suite**: Objective: Are we building it right? How?: Maintaining the tests and suites for testing backend. Communicate when suite is failing because of a change in backend or environment.

Legend:
- Blue: Business / Backlog
- Orange: Development Board
- Green: Continuous work
Developers learned how to write documentation in the form of a test case, and how to learn about app. features from test cases.

Developers write unit / Integration / E-E test for the piece of code they develop.
Shift-left testing approach

QA activities are performed on all stages of SDLC
Three amigos as a concept refers to the primary perspectives to examine an increment of work before, during, and after development. Those perspectives are:

**Business** – What problem are we trying to solve?

**Development** – How might we build a solution to solve that problem?

**Testing** – What about this, what could possibly happen?

*Three amigos help to focus QA activities on the most valuable business areas. They decide how a new feature will be tested before it is even developed. That means building a TESTABLE application.*
Testing pyramid

https://testing.googleblog.com/2015/04/just-say-no-to-more-end-to-end-tests.html

https://martinfowler.com/bliki/TestPyramid.html
Definition of done

Feature DoD Examples:
• Acceptance criteria met
• Testing done:
  • Test cases(documentation) created
  • Feature level automated tests(unit/integration/e2e) created
  • Automated regression pass
• Non-Functional requirements met
Which part of my presentation do you most disagree with, and why?