

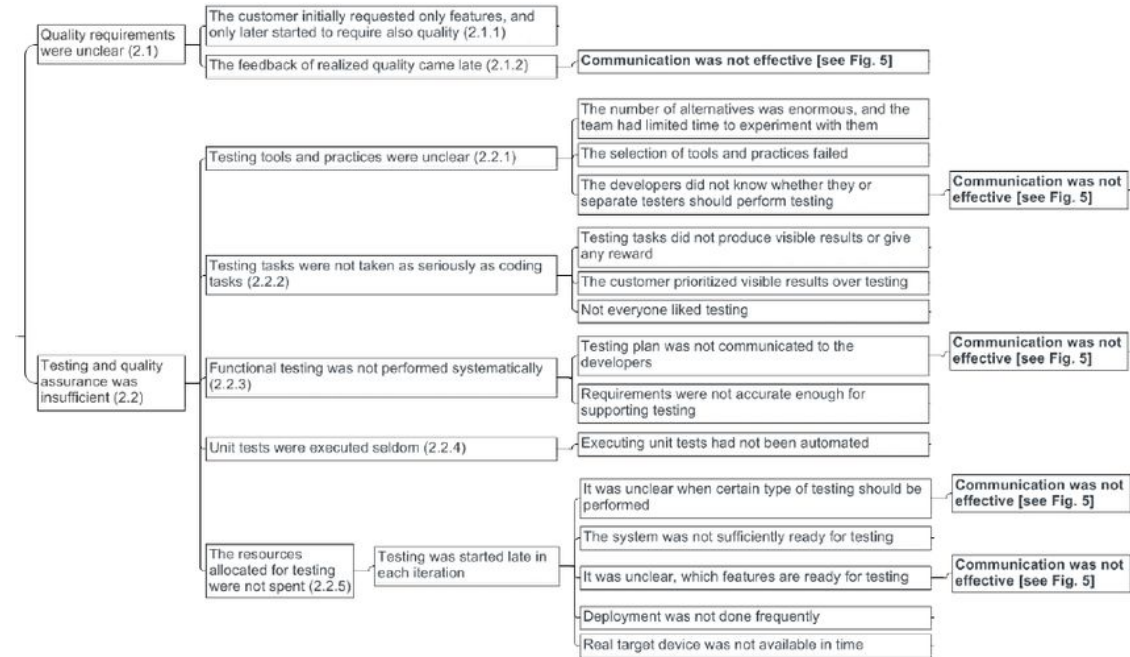
EES2: Quality Assurance & Testing

**Antti Ahonen - Tech
Excellence Oy**

Modern quality assurance

- Why?
 - to provide customer value for the end users
 - less errors (external quality)
 - more speed to iterate on customer needs (internal quality)
- What?
 - coding practices
 - testing
 - code reviews
 - static code analysis
 - continuous integration & quality gates
- How?
 - constantly → automation
 - as fast as possible → automation
 - all phases by developers, some parts by dedicated qa personnel
 - → creates feedback loop

Quality assurance problems on this course?



Vanhanen, Jari & Lehtinen, Timo. (2014). Software Engineering Problems Encountered by Capstone Project Teams. International Journal of Engineering Education. 30. 1461-1475.

Testing

- Automated
 - unit
 - integration
 - system
 - acceptance
- Manual
 - exploratory
 - confirmatory

Practical testing strategies for course?

- *How extensive should the testing be this early in the project?*
- *Which tests should be automated? Balance between automatic/manual testing?*

Testing

Discussion topics:

- How to document manual testing, frameworks? Has your team used any methods, pros/cons with them?
- Accessibility testing
- Testing techniques for frontend-development?
- Testing and machine learning?
- How to implement tests in each sprint?
- How much time should be given to testing vs coding?

Code reviews

- Synchronous
 - Pair or mob programming
- Asynchronous
 - Pull/merge requests
- **Why to do it?**

Code reviews

Discussion topics:

- Reviewing other things than code, i.e., cloud (AWS, Azure...) setups
- **Code reviews strictness vs efficiency in the early stage?**

Continuous integration

- Doing incremental development
 - small batches
 - verifying that integration between the new batch and the old main works as expected
- Pipelines
 - Phases
 - Quality gates
 - **Describe a good continuous pipeline setup?**
- Different tools
 - Gitlab CI
 - Github Actions
 - CircleCI
 - TravisCI
 - Jenkins etc..

Continuous integration

Discussion topics:

- Gitlab runner and pipeline setup
- How to create test pipelines (eg - in azure devops)
- How to automate testing for continuous integration?
- **What kind of quality checks in your code are necessary to avoid unreliable code being integrated?**

Static Code checks

- Analyze the code without running it
- What is it good for?
 - Programming errors
 - Coding standard violations
 - [Syntax](#) violations
 - Security vulnerabilities
 - Structure correctness
- Tools
 - eslint
 - prettier
 - sonarqube
 - snyk
 - react-axe
 - dependabot

Static Code checks & other QA

Discussion topics:

- **Measuring/testing non-functional requirements**
 - performance, security, usability, maintainability?
- Accessibility testing
- Facilitating Dailies in a natural and organic way?