Software Test

General Information
v1.1

Uwe Gühl, Jittat Fakcharoenphol

Fall 2007/ 2008

Thank You

- Jittat Fakcharoenphol – for fun during jointly preparation
- Arnon Rungsawang – who made my stay at KU possible
- Students of the courses TIT05IN, TIT05NSA, TIT05NSB at Berufsakademie Stuttgart* – for their feedback
- Prof. Nitsche-Ruhland – who made the first lecture at Berufsakademie Stuttgart* possible and gave much feedback
- Mirja Meyer Friese – for her suggestions concerning quality, roles, integrated concept and her test case example
- Sigi Voigt – for details concerning test automation and tools
- Stefan Brähler – for feedback and suggestions
- Markus Schell – for defect management examples
- Susanne Girod and Klaus Knoll – for test process examples and exercises

*university of cooperative education

Contents

- Administrative
- Contents of the course
- Literature
- General Idea
- Grading
- Project Teams

Administrative

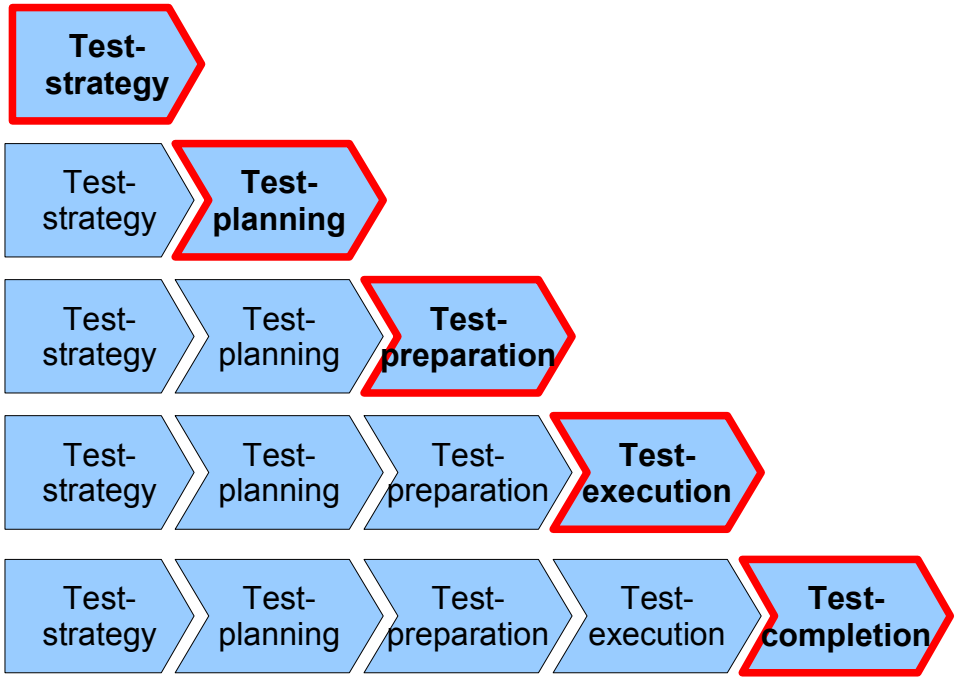
- 219343: Software Test
- Lecturer
  - Uwe Gühl, Email: uweguehl@hotmail.com
  - Jittat Fakcharoenphol, Email: jtf@ku.ac.th
  - Department of Computer Engineering
  - Faculty of Engineering
  - Kasetsart University, Bangkok, Thailand
- Lecture documentation:
  - http://mike.cpe.ku.ac.th/~uwe
- Office hours: on demand

Contents of the course (1/2)

- Overview / Introduction
- Test basics, terms, definitions
- Unit testing
- Inspection & review
- Automated testing & model-based testing
- Specification review techniques
- Black Box Testing

Contents of the course (2/2)

- A sample testing cycle
  - Test strategy
  - Test planning
  - Test preparation
  - Test execution
  - Test completion
  - Outlook, sources



Literature

- [KFN99] Cem Kaner, Jack Falk, Hung Quoc Nguyen: Testing Computer Software, Wiley Computer Publising, 1999
- [KBP02] Cem Kaner, James Bach, Bret Pettichord: Lessons Learned in Software Testing, Wiley Computer Publising, 2002

General Idea (1/5)

- Combining theory with practice
- Project
  - Development of a “KasetClock” for international Teams
  - Many watches could be shown at the same time

General Idea (2/5)

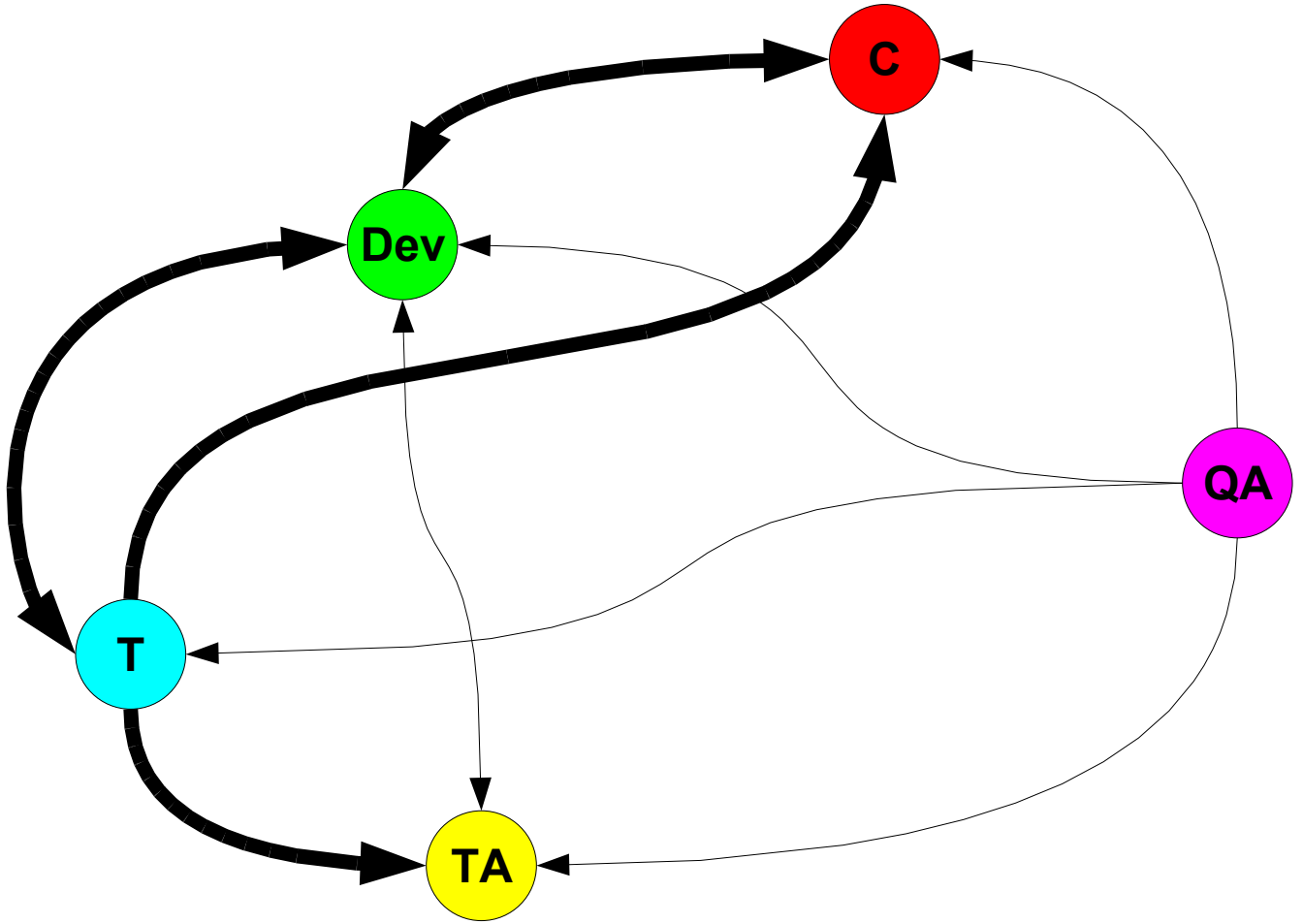
- Establishing of 5 (Testing) Teams
  - Focus in contents
    - White Box / Black Box / Test automation / Quality assurance
  - Cooperation of the teams
    - E.g. Automation Test Team uses Test Cases of Testing Team

General Idea (3/5)

- Teams
  - Consulting Team (for Customer) C
    - Specification, BUC, UC, Change Mgmt.
  - Developer Team Dev
    - Software-Development
  - Quality Assurance Team (for Developer / Process) QA
    - QA / Guide lines / ReleaseMgmt / CVS / Reviews
  - Testing Team T
    - Test prep: TC, TS, TD / Test exec / Defect mgmt.
  - Test Automation Team TA
    - Test automation prep / exec.

General Idea (4/5)

- Communication between Teams (Idea)



General Idea (5/5)

- Homework
  - 10 times (2 times each team) presentation to all!
  - 5 times: Individual homework
    - Technical Test Case Creation and Execution during programming
    - Test Case Creation
    - Test Execution

Grading (1/2)

- The score of the course will be calculated like this:
  - 50 % Group work
    One score will be given to each member of a group
    Base for the score for the group work are
    - 2 presentations given to the course
    - Final documentation covering the working results
  - 50 % Individual
    Base:
    - Final examination at the end of the course (45 %)
    - Homework (5 %)

Grading (2/2)

- That's why the **main goal** of each group is:
  - Presenting the status of the work to the group.
    Please keep in mind: We would like to learn!
    A typical presentation should content:
    - What did we do?
    - What were success factors to bring the project forward?
    - What could have be done better?
    - Next steps?
  - Creation of a final documentation
    Should contain a problem description and should describe the work which was done

Project Team
Consulting Team (1/2)

- Goal: Creation of specification and to ensure, that the customer could accept the delivered software
- Tasks
  - Creation / updating of a specification and requirements
  - Creation / updating of Business Use Cases to describe the Business Processes
  - Creation / updating of Use Cases
  - Set up a change management
  - All activities to make an acceptance possible

Project Team
Consulting Team (2/2)

- Competence
  - Specification Know How
  - Modelling: Use Cases, Business Process
  - Communication
- Responsibility
  - Satisfaction of the Customer
    - Easy downloadable and running software is available
    - Acceptance is possible
  - Representation of the Customer
  - Information about updated specification

Project Team
Developer Team (1/2)

- Goal: Creation of a running system
- Tasks
  - Software-Development
  - Enabling of revision control, version control, source control and of a release management
  - Creation of an installing procedure
  - Ensuring quality (e. g. with automated test cases)
  - Program description
  - Creation of a Handout (pdf) for user

Project Team
Developer Team (2/2)

- Competence
  - Programming Know How
  - Design Pattern, UML
- Responsibility
  - Development of good designed, extendable open source code
  - Use of methods to ensure high quality
  - Creation of an easy to download and use software

Project Team
Quality Assurance Team (1/3)

- Goal: Assuring high quality in the project, definition and ensuring of processes
- Tasks
  - Ensuring high quality - passive and active – general
    - Definition of standards for documents, deliveries
  - ...and specific in
    - Processes
    - Specification (Quality of Specification, Requirements)
    - Programming (Quality of Code)
    - Testing (Quality of Test Cases, Test Scenarios)

Project Team
Quality Assurance Team (2/3)

- Tasks
  - Planning, defining and controlling processes
    - Offering methods
    - Description of documentation policy (standards)
    - Organizing and perform reviews
    - Definition and control of time line / milestones

Project Team
Quality Assurance Team (3/3)

- Competence
  - Know how concerning QA-methods
  - Abilities to abstract
  - Basic Specification, Modelling, Programming Know How
  - Communication
- Responsibility
  - Defining and ensuring of quality standards in the project
  - Ensuring that processes work fine

Project Team
Testing Team (1/2)

- Goal: Documentation of defects, high test coverage, statements related to software quality
- Tasks
  - Test strategy and planning
  - Test preparation: Creation of Test Cases, Test Scenarios and Test Data
  - Test execution
  - Establishing of a Defect Management
  - Selection and use of a Defect Management Tool (bugzilla, trac)
  - Selection and use of a Test Tool

Project Team
Testing Team (2/2)

- Competence
  - Specification Know how
  - Testing Know how
  - Data Modelling (UML, Entity-Relationship-Models)
  - Communication
- Responsibility
  - Documentation of Test coverage and defects
  - Helping reducing defects in the final software with intelligent testing

Project Team
Test Automation Team (1/2)

- Goal: Efficient use of automation tools to document defects, to achieve high test coverage, and to document software quality
- Tasks
  - Test automation strategy and planning
  - Collaboration with Developers concerning Unit-Testing
  - Collaboration with Testing team to choose best Test Cases, and Test Scenarios
  - Test Data Management
  - Test preparation and Test execution
  - Selection and use of a Test Automation Tool

Project Team
Test Automation Team (2/2)

- Competence
  - Programming Skills
  - Testing Know how
  - Data Handling and Data Modelling
- Responsibility
  - To automate Testing to help reducing defects in the final software with intelligent automated testing concerning
    - Unit testing
    - Functional testing
    - Non-Functional testing