Group Administration Module
- an examplary case of software development for ILIAS3

Nils Eiken, Matthias Maschke, Martin Rus @ ILIAS open source, Universität zu Köln
Topics

• Introduction
• Approach to software development
• Results
• Perspectives
Approach to software development

Requirements specification:
Requirements specification:
  • Use Cases to figure out the functional requirements
Approach to software development

Requirements specification:
• Use Cases to figure out the functional requirements
  → recognize interdependencies
Approach to software development

Requirements specification:
• Use Cases to figure out the functional requirements
  → recognize interdependencies
  → understand the application flow
Approach to software development

Requirements specification:
- Use Cases to figure out the functional requirements
  - recognize interdependencies
  - understand the application flow

Design:
Approach to software development

Requirements specification:
• Use Cases to figure out the functional requirements
  → recognize interdependencies
  → understand the application flow

Design:
• distinction into layers:
  • Graphical User Interface (GUI)
  • Application layer
  • Core layer
  • Foundation layer
• two sectors with different views: administrator and user
  → administrator sector: ilObjGroup, ilObjGroupGUI
  → user sector: ilGroup, ilGroupGUI
Approach to software development

Implementation:
Approach to software development

Implementation:
• kick-off as a group (collaborative programming) 
  → big savings later
Approach to software development

Implementation:

• kick-off as a group (collaborative programming)
  → big savings later
  → good way to share knowledge
Approach to software development

Implementation:
- kick-off as a group (collaborative programming)
  → big savings later
  → good way to share knowledge
  → increases quality!

- later split-up according to class responsibilities
  → weekly meetings about the development status
Approach to software development

Implementation:
• kick-off as a group (collaborative programming)
  → big savings later
  → good way to share knowledge
  → increases quality!

• later split-up according to class responsibilities
  → weekly meetings about the development status

Testing:
• continuous testing during coding
  → early discovering of mistakes
Results

Technical requirements:
• high demand for reusability of classes
Results

Technical requirements:

• high demand for reusability of classes
  → requires well designed interfaces
Results

Technical requirements:
• high demand for reusability of classes
  → requires well designed interfaces

Organizational requirements:
• ongoing development of core classes
Results

Technical requirements:
• high demand for reusability of classes
  → requires well designed interfaces

Organizational requirements:
• ongoing development of core classes
  → sustained adjustments according to changes
Results

Technical requirements:
• high demand for reusability of classes
  → requires well designed interfaces

Organizational requirements:
• ongoing development of core classes
  → sustained adjustments according to changes

• cooperative software development
Results

Technical requirements:
• high demand for reusability of classes
  → requires well designed interfaces

Organizational requirements:
• ongoing development of core classes
  → sustained adjustments according to changes

• cooperative software development
  → high level of coordination
Results

Technical requirements:
• high demand for reusability of classes
  → requires well designed interfaces

Organizational requirements:
• ongoing development of core classes
  → sustained adjustments according to changes

• cooperative software development
  → high level of coordination
  → strict compliance with coding conventions
Perspectives

• Requirements laid down have been realized
Perspectives

• Requirements laid down have been realized

• optimization of code with regard to performance and coding conventions
Perspectives

• Requirements laid down have been realized

• optimization of code with regard to performance and coding conventions

• appropriate way of developing for given situation
Contact:

ILIAS open source  -  ilias-info@uni-koeln.de

Prof. Dr. Wolfgang Leidhold
Matthias Kunkel
Boris Schürmann

Head of Project  wolfgang.leidhold@uni-koeln.de
Project Management  m.kunkel@uni-koeln.de
Software Development  boris.schuermann@uni-koeln.de