

## Master Thesis offered

Within the framework of the FFG Bridge project "*QuASE – Quality Aware Software Engineering*", the Application Engineering Group offers the following subject:

## Ontology Support for Quality-Aware Software Engineering

### Research question:

Analysis of state-of-the-art software engineering ontologies in comparison with the QuASE Quality Ontology (QuOntology).

### Description of practical work:

The student will be a member of the QuASE (prototype) development team. She/he will realize QuOntology in OWL and implement the QuOntology support module including the related QuASE tool core components.

### Responsibilities:

- Communicating with our industry partners on project-related topics
- Taking part in developing the QuASE architecture
- Performing the detailed design, coding and testing

### Required skills:

- Requirements and Software engineering experience, knowledge of the software process
- Basic knowledge of ontology-based techniques
- Proficiency in Java (and the knowledge of the Java and Eclipse platforms)
- Knowledge of the Eclipse development environment
- Database design and implementation skills
- Ability to work in a team, good communication skills

### Skills acquired in the course of the project:

- Working with modern issue management systems (JIRA, Mantis, XEOX, SAP, @enterprise) and their development support (e.g. plug-in APIs)
- Knowledge and experience management skills
- Extended knowledge of ontology-based techniques

### General Conditions:

- Start date: as soon as possible
- Salary: Employment as student assistant (KV §30 "Studentische MitarbeiterIn")





## The QuASE Project

The elicitation of and the agreement on quality-related requirements is still an open challenge in the software development process: Software developers and business stakeholders have different backgrounds, use different terminologies, and they have different views on the software to be developed as well as on the environment the software has to be embedded into. As a consequence, quality defects often are detected and complained by the users only when the software is developed and made available for acceptance testing. However, improvement measures at that stage are expensive and often lead to unpleasant disputes and even to the failure of the whole project.

These consequences may be softened by including the business stakeholders, i.e. the future users, into the whole development process in order to allow them to validate intermediate results. This presumes an effective communication between all parties concerned. With QuASE we want to provide the scientific and practical foundations to support this communication: We aim at a software tool that supports the co-operative elicitation and analysis of quality-related requirements while allowing the concerned parties for using their individual terminologies, and that can be integrated with existing and widely used Issue-Management-Systems. Core of that solution is a knowledge base (i.e. a semantic repository), into which quality-related knowledge is integrated along a view-independent and continuously established quality-ontology. For the analysis and validation by the different parties, the knowledge base contents are retransformed and presented according to their individual views (ontologies).

Intelligent support shall be provided for the following development process stages:

1. *Foundation stage*: establish the knowledge structures that are relevant for the intelligent support of dealing with quality-related issues;
2. *Elicitation stage*: acquire raw information about quality-related issues from the different parties in a software process, and convert this information into a set of semantic knowledge structures reflecting the views of these parties;
3. *Integration stage*: convert and integrate (including conflict resolution) the results of stage 2 into a "global view";
4. *Analysis stage*: based on the global view perform analytical tasks such as facilitating knowledge reuse or predicting the handling of future issues;
5. *Dissemination stage*: convert and externalize the global view back into the views of the different parties;
6. *Completion stage*: final integration and acceptance of the tool solution.



QuASE aims at enabling software developers (SME's but also development departments of larger enterprises) to incorporate quality-related issues and requirements into their design and development decisions throughout the whole development process and in close and loss-free communication with the future users. This will reduce the risk of late and costly rework or even complete project failure. Thus, QuASE Tool will contribute to increase the quality of software and thus strengthen the competitiveness of its developers. QuASE Tool is intended to be introduced into the market and can be used stand-alone or associated with widely used Issue-Management-Systems like JIRA or Mantis. The market includes development companies and departments as well as management consultants and IT-service providers. For the long term, the tool also may be introduced into other areas where quality management is of relevance.