M3 Assignment Prototype

1 Introduction

This document describes the results that must be submitted at milestone M3. It also contains the criteria according to which the results will be assessed.

Upload the artifacts to the corresponding submission folder on Moodle before the presentation. The exact submission modalities are described on Moodle at the respective milestone.

2 Submissions

The following results must be submitted at Milestone 3.

2.1 Source Code

The final version of the source code for your project must be submitted.

To submit the source code, place a single ZIP file with the following contents in the designated Moodle submission folder: complete source code including all configuration files required to run the program; generated API documentation as HTML (Javadoc or similar).

2.2 Technical Report II

Technical Report II expands on Technical Report I in terms of content, primarily with further design decisions. In addition, the feedback from M2 must be taken into account appropriately. It must also be uploaded to Moodle as a PDF document.

3 Structure of the Technical Report II

A technical project report (approx. 30 pages) with the most important results of the project must be prepared.

3.1 Content Structure

- Title page
- Abstract (in English)
- Project idea
- analysis
- design
- Implementation
- Results
- Directories and appendix

3.2 Abstract

Please consult the «Guideline formales und wissenschaftliches Schreiben» chapter 6 for instructions on how to write a correct abstract.

3.3 Project Idea

The most important points of the corrected and updated project outline (for everything except sections 2.2.6-2.2.9, see the document “M1 Project sketch”) are summarized again here.

3.4 Analysis

The most important results of the problem analysis are to be documented here:

- Use case model
  - UML use case diagram for an overview
  - Most important use cases fully dressed, rest casual or brief
- Additional requirements
- Domain model

3.5 Design

The most important aspects of the chosen solution must be documented:

- Software architecture
- Design class diagram (DCD)
- Selected interaction diagrams
- Documentation of important architecture aspects, design decisions and applied design patterns

Document the existing design (primarily domain layer, without UI) with a de-sign class diagram (DCD) and suitable interaction diagrams. Also show a corresponding interaction diagram for at least 4 system operations. The UML class diagram can also be logically divided into several pages (e.g. according to packages). UI classes do not belong in this diagram.

The principles and design patterns applied should be appropriately documented.

3.6 Implementation

The most important information on the packaging of the solution and the tests carried out must be documented here:

- Delivery results (deliverables), executable files (jar, exe etc.)
- Test concept: Summary of the test strategy - which tests were carried out at which test levels with a justification - and an overview of all tests (unit, integration and system tests)
- The detailed test results are an optional part of the appendix.

3.7 Results

The project results should be briefly summarized and placed in relation to the original project idea:

- Summary of the objectives achieved
- Open points
- Outlook for possible further developments

3.8 Indexes and Appendix

Bibliography according to IEEE standard

List of figures and tables
Glossary (optional)
Appendix (optional), e.g:
- Installation instructions (optional)
- Operating instructions (optional)
- Test results (optional)
- Project management
Description of the procedure/methodology (milestone/iteration plan).
Summary of the actual effort with the planned effort per iteration and in total.
Justification of significant deviations.

4 Evaluation

The assessment aspects for the prototype results are briefly explained in the following sections.

4.1 Source Code (30%)

The following aspects of the source code are evaluated:

- Conformity with the design
- Is the code executable?
- Code quality (clean code, naming, compliance with coding conventions)
- Code documentation (Javadoc)
  - Commented responsibilities in the class comment
  - Commented public methods
- Testing
  - Test concept (described in the technical report)
  - Coverage and quality of unit, integration and system tests
- Achieved goals (compared to the project outline or the agreed features)
- Scope of the implemented functionality
- Degree of difficulty
- Effort invested

These 9 aspects are evaluated separately and are included in equal parts in the evaluation of the prototype.

4.2 Technical aspects of the technical report II (10%)

All chapters of Technical Report I are evaluated separately according to the following criteria:

- Completeness and comprehensibility according to the above description
- Formal correctness according to the theory lessons in SWEN1 (as far as reasonable)
- Comprehensibility and comprehensiveness
- Feedback from M2 taken into account
- Appropriateness to your project

The 7 chapters of the Technical Report II are weighted equally and result in the technical grade of the Technical Report II.

4.3 Linguistic Requirements

The Technical Report II must be written in a language that is understandable to those addressed, yet precise and formal. The text should be structured according to the template, with meaningful subtitles, in an appealing layout, with a title page, table of contents and references. The various text elements (statements, sections) are formulated coherently and make explicit reference to graphics or tables. The essential facts are documented (references), the assumptions made are clearly stated and the conclusions drawn from them are easy to understand. The language used must be correct in terms of form, grammar and content and adapted in style to the intended audience.

4.3.1 Linguistic assessment (10%)

Specifically, Technical Report II is assessed linguistically based on the following criteria:

- The comments on the project outline must be fully incorporated into TB2
- The structure (incl. abstract) corresponds to the report structure and requirements specified above
- The content is relevant and supported by high-quality sources.
- The style is scientific, factual and consistent.
- Sentences and paragraphs are logically linked
- Text-image references are explicit and explicative
- The text is formally correct (grammar, spelling, semantics)
- Figures, references and the bibliography are consistent according to IEEE.

4.3.2 Fundamental documents – linguistic standards of the SoE

- «Guideline formales und wissenschaftliches Schreiben»

4.3.3 Further material

- «Useful websites for your English»