

LOG8371E: Software Quality Engineering (Fall 2023)

TP1 (15% weight)

Functional Suitability, Reliability, and Maintainability

Notes:

Practical work is an important part of the course and is intended to motivate you to understand software quality assurance practices, design software quality assurance plans, develop verification strategies, and use the various tools available to evaluate the quality characteristics of software. It is recommended that you take this work seriously and use your creativity and critical thinking to make it a success. Collaboration with your colleagues is allowed during and outside the laboratory sessions. However, plagiarism regulations still apply at all times.

I) Objectives:

The objectives of this first TP are to get familiar with:

- The definition of software quality plans.
- The definition of software quality assurance activities (e.g., tests, reviews).
- The verification methods for different quality characteristics.

II) Specifications:

Q1) Prepare a quality plan for the PetClinic system (30 points)

The quality plan will include a list of quality goals and a plan for quality assurance strategies.

Quality goals: The plan will apply on a reduced version of the PetClinic system with one major feature (you choose your own feature by imagining your team is responsible for a major component of the system). The plan will focus on **two from the three quality characteristics** (ISO 25010): Functional Suitability, Reliability, and Maintainability. For each quality characteristic, define at least **two quality goals**, each corresponding to a different sub-characteristic (ISO 25010). In other words, you should have at least four quality goals in total. For each quality goal, indicate the quality characteristic, sub-characteristic, the quality measure, and the specific objective (you can use a table). (20 points)

Quality assurance strategies: Describe the planned activities (e.g., reviews, audits, tests, static analyses) for the assurance of the quality goals. If applicable, describe sub-types of activities (e.g., unit/integration/system testing). For each planned activity, describe its scope (e.g., for verifying which quality (sub-)characteristics or which goals), stage (e.g., development stage,

system testing stage), and the roles involved (e.g., developers, independent testing teams, clients). A table or other structured presentation is recommended. (10 points)

Q2) Verify the quality characteristics of PetClinic (50 points)

Measure the quality characteristics of the subject software using the quality measures from Q1 and verify whether each quality objective has been met. Describe your verification methods and results.

Verification methods: For each quality goal from Q1, investigate the existing verification methods and choose/design one as your solution. You can use either an automated tool or manual effort. Describe your verification methods in a clear, structured manner. For example, you can first use a table to summarize your verification method (used tool, approach) for each quality goal, then describe the detailed verification steps for each quality goal. The described verification methods should be unambiguous and reproducible. (30 points)

Verification results: Report the verification result for each quality goal. For each quality goal, provide the obtained value of the corresponding quality measure and indicate whether the quality objective is met. When a tool is used, you can leverage screenshots of the tool output to demonstrate the result. (20 points)

Q3) Improve the system and compare the quality characteristics between the two versions (20 points)

The goal of this question is to investigate the methods of improving software quality. Focus on one quality characteristic (e.g., maintainability) and the two related quality goals. Investigate and perform approaches (e.g., refactoring) to improve the quality in terms of the focused quality characteristic. Describe your improvements. Verify the quality characteristic of the improved version (same as Q2, but focus on one quality characteristic) and compare the obtained quality measures with the results from Q2 (using a table). Discuss the differences and explain the causes of the differences (i.e., the specific improvements that lead to the differences).

III) Presentation:

Each team should submit a report no longer than 15 pages. The report must contain: 1) a table of contents; 2) an abstract that provides a clear overview; 3) an introduction that describes the subject software, the importance of its quality, the chosen feature, the considered quality characteristics, and an overview of the approaches; 4) the quality plan (including the quality goals and the assurance strategies); 5) detailed verification of the quality characteristics (including the methods and results); 6) description of the approaches for quality improvement; 7) comparison of quality characteristics between two versions; and 8) a conclusion that

summarizes the key points of the report. Diagrams and tables should be clear and have captions; diagrams and tables are self-explanatory or explained in the text. References should be used appropriately when necessary. The report shall be of high quality and be treated professionally (suppose you will submit it to the managers of a company).

IV) Evaluation:

Your report will be evaluated by both the quality of work and the quality of presentation:

Team Score = Quality of work * sqrt (Quality of presentation) * 15 (Weight of TP1)

For example, if you get 90% for the quality of work and 90% for the quality of presentation, your score will be: $0.9 * \sqrt{0.9} * 15 = 12.8$

Individual Score = Team Score * Contribution Factor

Contribution Factor (CF): CF = 1 for satisfactory contribution; CF = 0 for no contribution; $0 < CF < 1$ for unsatisfactory contribution. CF is determined by optional peer reviews.

Optional peer reviews: Each team member may optionally email a confidential statement of work to the lab instructor within 24 hours after the due date of the assignment. A statement of work first lists in point form the parts of the assignment to which each team member contributed. In addition, the statement of work also describes whether the work load was distributed fairly evenly among the team members (you may describe the percentages of contribution by each team member). A statement of work will be used to determine the contribution factor and the score of a team member who is not contributing sufficiently to the assignment, subject to further investigations. It is not necessary to send a statement of work, if a team distributed the work for the assignment fairly evenly and each team member had satisfactory contribution.

We strongly encourage each team to use a **private GitHub repository** for the collaboration. The repository can not only help the team manage and track the work, but also provide evidence for each individual's contribution in case an investigation is needed.

V) Submission:

The work must be done in teams of 3-4 people and must be submitted via Moodle no later than:

➤ **September 28th before 23:59** (one submission is needed for each team).

Please submit a PDF report that includes the names and student numbers of the team members. The file will have the name:

➤ **log8371E_TP1_[TeamName].pdf**

Late work will be penalized by 10% per day of delay. No work will be accepted after 4 days of delay.