

TB141 – ICT System Engineering and Rapid Prototyping Exam

Jacopo De Stefani - Marijn Janssen

24th May 2023

Instructions

General remarks

- The exam lasts three hours and it is a closed book, computer exam in two parts :
 - **First part - UML Modeling**
 - * **Points : 10 pts**
 - * **Weight on the final grade : 30%**
 - **Second part - ICT System Engineering**
 - * **Points : 10 pts**
 - * **Weight on the final grade : 40%**
- The remaining 30% of final grade of the course is based on the (previously developed) summative assignment.
- Make sure to write your name, family name, student ID and faculty on **each** sheet you are going to hand in.

Disclaimer

All characters and other entities appearing in this work are fictitious. Any resemblance to real persons or other real-life entities is purely coincidental.

Part I - UML Modeling

Given your excellent performances in modeling the TUDelftToGo application, TUDelft would like to employ your expertise in modeling and designing the IT system supporting the master thesis supervision process in a university research center.

The research center is organized in research units. Each research unit is directed by a senior researcher and can employ up to ten junior researcher. For every researcher, the system should store their name, family name, ID, birth date, and date of the start of their employment. Both junior and senior researchers are allowed to be part of a supervision committee, but only senior researchers are allowed to chair the committees. Each research unit can participate in one or more research projects, but only one research unit is responsible of the coordination of a project. Each project is characterized by its name, its domain and its budget. Each master thesis is identified by its title, author list and list of keywords. A master thesis can be developed as part of at most one research project but could also be developed without any association to a specific project.

The master thesis supervision process is started by a student, which has a preliminary meeting with his thesis committee. After this meeting, the student defines an hypothesis and starts running the experiments to test his/her hypothesis. If the results are not satisfying, the student needs to consult again his/her principal investigator. Then, the PI proceeds in adapting the hypothesis and communicating to the student a new set of experiments, until a satisfying result is obtained. If the results of the experiment are satisfying, the student proceeds with writing the first draft of the thesis. Once she/he wrote the first draft, the thesis will be sent to all his committee for proofreading and feedback. The student will then proceed to integrate all the corrections, and submit the final version to the university administrator.

We ask you to model the aforementioned ICT system using the UML modeling language. For each of the following questions, provide :

- The required UML diagram
- A brief text motivating the modeling choices you made and the additional modeling hypotheses you might have made.

Make sure to employ the appropriate notation for the requested diagram.

Question 1 (UML Class Diagram - 5 points)

Represent the entities that the application described in the previous text needs to manipulate using a UML Class Diagram. Make sure to include all the relevant attributes and methods in the appropriate parts of the diagram.

Question 2 (UML Sequence Diagram - 5 points)

Represent the master thesis supervision process, including regular/exceptional flows. described in the previous text using a UML Sequence Diagram. Make sure to use the appropriate notation to distinguish between synchronous, asynchronous and response messages.

Part II - ICT System Engineering

Question 3 (Software engineering requirements - 3 points)

Given the following case :

You are tasked with the objective to develop an application to manage multi-modal travel reservations, that is, allowing to store reservations for different means of transportation (e.g. train, bus, car sharing, boat). The application should allow to :

- *Import existing tickets from the most common public service companies.*
- *Export a summary of monthly expenses divided by company.*
- *Seamlessly share tickets across multiple devices for the same users.*

The reservations should be stored both locally on the devices and in a cloud storage and should be password protected in order to ensure users' privacy. The maximum response time of the application should be below 2s. Moreover, the user interface application should be accessible also to people having visual impariments. Finally, the storage of sensitive in the application should be GDPR (General Data Protection Regulation - EU norm) compliant, and compliant with the regulations of the different supported countries.

- Provide a short definition for the different category of requirements (functional, non-functional, domain requirements).
- Provide a table, for each category of requirements, summarizing the classification of the different requirements in the text and the motivation behind your choices.
- For 3 requirements of your choice, describe which format (natural language, structured natural language, graphical notation, mathematical specification) you will be using for their specification.

Question 4 (Software development methodology - 2,5 points)

For an **incremental** software development methodology (Agile-like), provide :

- A list describing the sequence of the different software development activities for the proposed process
- A brief description, for each activity in your solution, of the details of the operations performed within that step
- Advantages and disadvantages of the considered methodology

Question 5 (Programming languages - 2,5 points)

1. What is a programming language?
2. Present the different categories of programming languages introduced during the course in a table including three columns :
 - The name of the category.
 - A description of the category.
 - Languages belonging to the category (include at least one language per category).

Question 6 (Computing Hardware and Architectures - 2 points)

Present the **pipe and filter** software architecture introduced in the course by providing :

- Its description and its components.
- A practical example (drawn from the course or your own experience) of implementation of the architecture

Licensing



Except where otherwise noted, this work is licensed by **Jacopo De Stefani** under a Creative Commons **CC-BY-NC-SA** license : <https://creativecommons.org/licenses/by-nc-sa/4.0/>

All images are all rights reserved, solely employed for educational use, and you must request permission from the copyright owner to use this material.