Software Engineering 2  
SUPD REPORT

Status Update Report

|  |  |
| --- | --- |
| **Team number:** |  |

|  |  |
| --- | --- |
| **Team member 1** | |
| **Name:** | TODO |
| **Student ID:** | TODO |
| **E-mail address:** | TODO |

|  |  |
| --- | --- |
| **Team member 2** | |
| **Name:** | TODO |
| **Student ID:** | TODO |
| **E-mail address:** | TODO |

|  |  |
| --- | --- |
| **Team member 3** | |
| **Name:** | TODO |
| **Student ID:** | TODO |
| **E-mail address:** | TODO |

|  |  |
| --- | --- |
| **Team member 4** | |
| **Name:** | TODO |
| **Student ID:** | TODO |
| **E-mail address:** | TODO |

|  |  |
| --- | --- |
| **Team member 5** | |
| **Name:** | TODO |
| **Student ID:** | TODO |
| **E-mail address:** | TODO |

Dear students,

This template document suggests an outline for the required contents of the SUPD report. The included descriptions and examples are supposed to help you write a clear report that documents and presents your actual solution well. Please remove this additional text (similarily written as this text) and exemplary material before you submit your report!

The actual SUPD report document can be based on this template, or can be written using an arbitrary text editing program such as Latex, LibreOffice and the like, as long as the required material (described in the assignment document) is contained.

In general, write the report in such a manner as to provide all information to a third party who is not involved in the design and development and unfamiliar with the exact tasks of the semester project assignment. Think of this third party as a company that roughly wants to offer a software product as described in the assignment document and commissioned your team to start this software engineering project and produce a viable project which follows best practices, is of high quality, and is worth to be funded further after the SUPD deadline.

The SE2 Team.

# Design Draft

## Design Approach and Overview - Chris

Describe your design approach and how you arrived at your solution. Typically you will start with an initial solution that you refine in an iterative way by means of re-factorization. Use at least class diagrams and the used technology stack for documenting:

* major design decisions (e.g., design alternatives). Discuss and explain your design decisions also regarding the overall class layout. Possibly include design descriptions/class diagrams showing (a part of) the design before a refactorization step and/or alternatively considered (parts of) designs.
* a design overview of the current state of your solution at SUPD. Note that used design patterns should be well visible in this design overview (e.g., in the class diagrams). Regarding design patterns you may cross-reference to Section 1.2 for the details.

Carefully check all your UML diagrams for syntactical and semantical correctness!

### Class Diagrams - Leyla

Class diagrams (for example, see Figure 1) describe the structure of a design. Try to omitt unnecessary detail (e.g., getter and setter operations), and focus on giving a good structural overview. You may also use multiple levels of details (zoom levels), or present multiple class diagrams showing parts of the overall solution that are (mostly) independent from each other.

Ein Bild, das Text, Karte enthält.

Automatisch generierte Beschreibung  
Figure 1: Sample Class Diagram from <https://medium.com/@smagid_allThings/uml-class-diagrams-tutorial-step-by-step-520fd83b300b>. TODO: remove in final submission!

### Technology Stack - Felix

Briefly describe the frameworks and/or libraries your solution is using or going to use (name, website link, version, 2-3 sentences describing the purpose of the framework/library in your solution). Also document considered alternatives and how you arrived at the decision to select one of the alternatives and disregarded the other(s). For your decision process you might want to pick a set of criteria (e.g., existing team experience, available documentation, convenience features, …).

## Design Patterns - Leyla

For each different design pattern make a subsection where you first briefly summarize the used pattern in general (in your own words!), then discuss how you applied the pattern in your solution. For each instance of the pattern in your solution

* give a (textual) argument how the pattern instance relates to one of the functional requirements,
* explain the particular problem that the pattern should solve,
* provide class diagrams and possibly other UML diagrams to give a high-level structural and possibly also behavioral description, and
* relevant code snippets from your implementation showing the implemented pattern

Focus on a detailed description for the first two pattern occurrences, but at least mention the remaining pattern instances along with references to your source code.

### Design Pattern 1: Factory Pattern

### Sample Design Pattern 2 TODO: replace with an actual design pattern

* Alle: Nach Pattern suchen, das ggf passen könnte

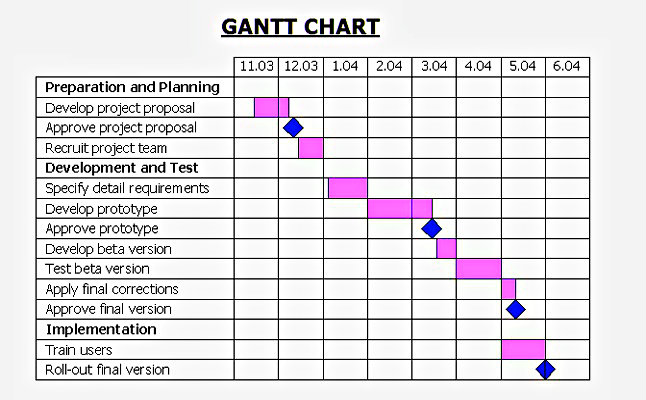
# Code Metrics - Alle

Provide information on at least the number of packages, lines of code, comment lines of code, number of classes and code bugs with regard to your current state of the implementation. It is recommendable to use a static code analysis tool (e.g. SpotBugs). Discuss your findings.

# Team Contribution

## Project Tasks and Schedule - Felix

Try to break down the overall semester project in several tasks and schedule them appropriately and in accordance with the SUPD and DEAD milestones. Gantt charts like shown in Figure 2 may help to guide and visualize this process. However, feel free to use any visualizations, methods, and tools that support you in meeting the project deadline, and report this here.

  
Figure 2: Sample Gantt Chart from <http://diagramscharts.com/gantt-chart/>. TODO: remove in final submission!

## Distribution of Work and Efforts – Chris & dann alle

Report in a table how you distributed the overall work among team members and how much time was spent by each team member on the tasks.