Name of the Report

Network Programming, ID1212

Your Name and Email Address

Date

Tips for Report Writing

REMOVE THIS SECTION BEFORE SUBMITTING THE REPORT.

The target audience has exactly the same skills as the author, except they do not know anything at all about the specific program described in the report.

Consider the following:

- **The report must be *centered around the requirements*. Which are they (Introduction), how did you work to meet them (Method), what is the solution that meets them (Result), and how can you be sure they are met (Discussion). This is the IMRaD method.**
- **The report must show that you have done the work yourself and that you have understood what you have done. Both of these goals are met by carefully explaining the source code.**
- Is spelling and grammar correct? Is spoken language avoided?
- Does the report have a good structure with sections, subsections and paragraphs?
- Is the solution clearly explained? Will the reader understand the program? What would you yourself want to know if you read about the program, is that included in the report?
- Is the solution analyzed and evaluated? Are important properties of the program explained? Should there have been more extensive evaluation?
- Is the text clarified with images and/or other figures, and with links to the code in your Git repository? Remember that all figures (images, tables, graphs, code listings, etc) shall be numbered and have a short explaining text.

1 Introduction

This section tells *what* are you going to do.

Explain the task and the requirements on the solution. It is very important to *clearly state the requirements*. If you wrote the program together with other students, list their names and email addresses here.

2 Literature Study

This section must prove that you collected sufficient knowledge before starting development, instead of just hacking away without knowing how to complete the task. State what you have read and briefly summarize what you have learned.

3 Method

This section tells *how* you solved the task.

Explain how you worked when solving the tasks and how you evaluated that your solution met the requirements. Mention development tools and IDEs you used. *Do not explain your solution and do not refer to code*, that belongs to the *Result* section.

4 Result

This section explains *the result* of what you did.

Present your solution and prove that it meets the requirements. It is very important to *state each requirement that is met* and explain *how you met it*. You do this by including user interface screenshots, like the example in Figure 1, remember that figures must be explained and referenced in the text. Also include links to the code in your Git repository. If you want the extra bonus point for a layered architecture, you must prove that also this requirement is met. You do this by explaining and motivating the layers and important design decisions.

Also, prove that you participated in writing the program, and that you understand it in detail. You prove this by explaining essential parts of the program.

5 Discussion

This section *analysis* the result presented in the previous section.

Summarize the requirements and *clearly state which of them you have met*. What lessons have you learned and what problems did you face? How were the problems solved? Should you have done something differently?

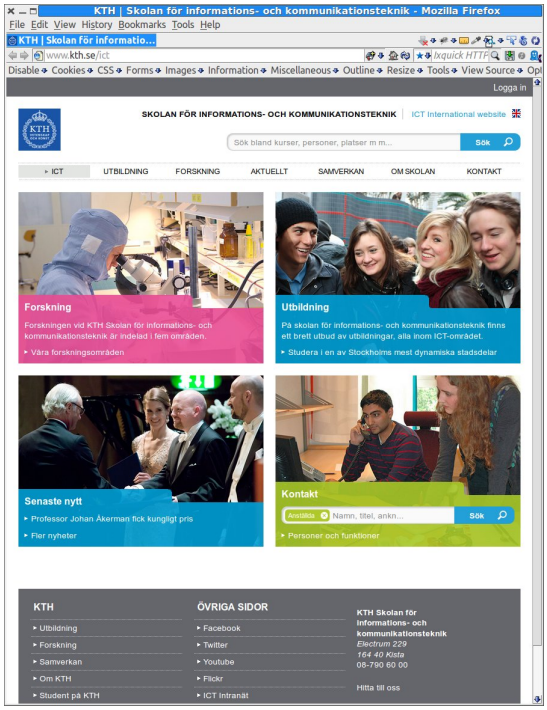

Figure 1: A sample user interface screenshot to illustrate caption (this text), numbering and reference in text.

6 Comments About the Course

Any comment(s) related to this course offering or to coming offerings is much appreciated. *Please also tell approximately how much time you spent on the assignment*, including lectures and exercises. This is of great help for course evaluation.