

ECE4220/7220 – Guidelines for Final Project Report and Submission

The final project reports should contain the main sections required for the lab reports, plus some additional requirements. The following organization is recommended:

1) Abstract

Brief overall summary of your project.

2) Introduction

Including the description of the project, the objectives/goals, the motivation behind the project, etc.

3) Background

Include any method that you are replicating (e.g. from a paper), common/relevant approaches used when addressing your particular problem or application, examples of where/why your application is used, etc.

4) Proposed method / System description / Implementation

Explain what you did and why you did it that way. This section should contain flowcharts/diagrams of your program(s) and of your hardware components. It is very important to explain how the multiple processes/threads interact, as well as how the devices/components interface.

5) Experiments and Results

Here you describe the tests that you did and present the results that you obtained. How did you make sure that your program was working properly? What cases did you try? How many times did you run your program(s)? How did you divide the work? In addition to describing your results, you should include tables, charts, snapshots, etc. Be sure to include statistics, when appropriate.

6) Discussion and Conclusions

Discussion/analysis of the results. Did your observations/results make sense? Were they what you were expecting? Why or why not? Discuss any problems that you may have encountered and how you fixed them. Talk about what you learned, interesting/unexpected things that you observed, problems/limitations that the current approach may have, alternative solutions/approaches, etc. Overall conclusions of the project.

7) Appendices

Include your well-commented code, and any other thing that you consider necessary.

In addition to the report itself -- described above -- you are also required to submit:

- a) Your programs -- i.e. source code, Makefiles, and binary files -- everything needed to create your binary files.
- b) A short ReadMe file explaining how to interconnect your system, and how to compile and run your program(s).
- c) A video demonstration of your project. Your video file should not be larger than 500 MB (read note 1 below).
- d) Your code should also be included in the appendix of the report, with comments, caption, and other details that will tell us what we are looking at.

Please, DO NOT submit "hundreds" of files!!! I will NOT accept such submissions. Everything must be in ONE SINGLE .zip/.tgz/.tar file!!

Name the file like this: ECE4220-LastName1_LastName2.zip (or .tgz or .tar). The last names should be in alphabetical order. You should upload the file to **Canvas**.

Note 1: If your video file is too large, Canvas may not accept your submission. If that is the case, create a .zip/.tgz/.tar file with everything except your video, and upload that to Canvas. Then, upload your video (named ECE4220-LastName1_LastName2-video) using the following link:

<http://vigir.ee.missouri.edu/~gdesouza/UploadDir>

Note 2: Presenting the code both in the Appendix and as an attachment (with Makefiles, README etc...) is threefold: 1) it allows us to judge the complexity of your project; 2) it allows us to measure your programming skills (the software designing skills are measured by the diagrams that were requested, and the explanations on how the multiple processes/threads interact); and 3) to make sure that your program compiles and runs (the latter, when possible). In the past, there have been cases where students submitted programs that wouldn't even compile due to syntax errors (C/C++ language). Once confronted, they replied that they didn't know that they were supposed to test their program/project. Do not be like that...