Student supervision guide*

Jaap-Henk Hoepman[1]

iHub
and
Institute for Computing and Information Sciences (ICIS),
Radboud University Nijmegen, the Netherlands,
jhh@cs.ru.nl

Abstract. This guide is meant to clarify, both for you (my student) and myself, how I supervise and judge master assignments, internships ("stages") and design assignments.

1 Introduction

All assignments consist of a certain period of *work* in which a certain problem is being studied or a certain product is being made. The results of the assignment are documented in a *report* and are summarised in a *presentation*. Sometimes, a separate *product* (e.g. a piece of software) is delivered. Sometimes, assignments are done externally (at a company). Sometimes, assignments are worked on in a group.

I usually have some predetermined assignments available. If you want to take up one of these, that's great. If you have your own topic for an assignment, that's even better. Feel free to contact me and discuss your ideas.

2 Assignments and supervision

All three (or four) of the highlighted topics above are important. Here are some of the main points to think about.

2.1 (Formal) procedures

The university has a set of formal requirements and procedures every bachelor or master project has to adhere to.

See https://ru.osiris-student.nl/#/onderwijscatalogus/extern/cursus/?collegejaar=huidig&taal=en&cursuscode=NWI-IBC033 (for bachelor projects) and https://www.ru.nl/en/institute-for-computing-and-inform education/master-thesis-lab (for master projects) for details.

* Thu Aug 20 15:19:28 2026 +0200 / bc5aa26 / student-guide.tex

(Note: this information was last updated 2026-03-19. The university has a habit of updating this information and/or changing the links where this information can be found.).

2.2 Work

In the end, you are responsible for finishing the assignment. I expect you to make a realistic project plan as soon as the assignments starts. When I foresee serious problems, I will tell you. Likewise, I expect you to inform me when problems occur or may occur. Otherwise, I expect you to work independently. I am available for questions, by mail, phone, video-call or face-to-face. Feel free to walk into my office for a chat, or a discussion.

Typically we will have one or two meetings a month to discuss progress and ask questions. You are responsible to schedule these meetings, make an agenda and take notes. If there is anything in particular you want to discuss, let me know beforehand, so I can prepare. If this requires some reading or reviewing, email me the documents at least a full working day before the meeting.

If you want me to review something, first carefully and critically read it yourself first.

2.3 Research plan

A good research plan is

relevant it solves a genuine theoretical or practical problem, that you are motivated and well equipped to answer, and that has not been answered before.

well founded it has a clear scope and is based on clearly expressed assumptions and a properly described state-of-the-art.

precise it clearly states the main research question.

methodological it clearly states the approach needed to answer the research question, the scientific method to be used, and the sub-questions to be answered.

functional it contains a detailed project plan that makes explicit which resources are needed, what external resources (information, support, etc.) it depends on, the steps to be taken and the amount of time they will take.

consistent the proposal as a whole is consistent and complete.

A good research plan provides convincing arguments for each of these aspects.

2.4 Report

It helps to make a skeleton of the final report early on in the assignment. It should contain chapter and section titles, each with a brief description of their final contents.

Discuss this skeleton report with me. For master assignments I expect the report to be in English. For internships and design assignments I leave it up to you.

Use the web. It's a great, albeit sometimes unreliable, source of information. But don't just cut-and-paste: always use your own words. Clearly indicate quotes. Cite your sources. Don't overdo it: it is better to cite a few good papers and read them, than to cite a lot you didn't really read.

Use generative AI sparingly, if at all. Clearly indicate if and how you have used it during your research and while writing the thesis.

Make sure you deliver the draft final report in time, to allow me to read it thoroughly. Allow me one week to do so. Then write the final report[1].

The university maintains a repository of bachelor and master theses, see https://www.cs.ru.nl/bachelors-theses/ (bachelor theses) and https://www.cs.ru.nl/masters-theses/ (master theses). Browse those to get a feel for what a proper report looks like. There are also default style files available at XXXX. Use those.

At the start of your project, read one of these two books with writing advice

- Pinker, S. (2015). The sense of style: The thinking personś guide to writing in the 21st century. Penguin Books.
- Leith, S. (2017). Write to the Point: How to be Clear, Correct and Persuasive on the Page. Profile Books.

There is additional information available at http://www.cs.ru.nl/~jhh/guides.html.

2.5 Presentation

Assignments end with a presentation of about 30-60 minutes. This includes the time for questions (either during the presentation or at the end). This means you will need 15-30 slides. You can use Powerpoint, Adobe acrobat (PDF), or the white board. Don't put too many words on one slide, and remember: pictures speak louder than words. Tell more than what's on the slide. Give examples of what you mean. Be prepared for questions. Look the audience in the eye.

Invite other people (friends, colleagues, etc.) to be present. This makes the presentation 'real', and ensures you feel you actually tell somebody something new. Aim your presentation at this — wide — audience.

There is additional information available at http://www.cs.ru.nl/~jhh/guides.html.

[1] Make sure there is time to do so.

2.6 Products

When writing software, remember that it may be used by others in the future (both as an application or as a basis for further development). Hence, documentation is of utmost importance. Stick to the following rules.

- Write a design document, that describes the structure of the software and documents the design choices.
- Write end-user documentation, that allows a tech-savvy person to use the software.
- Document each variable and function declaration in the source — at the very least the publicly accessible ones. For a function, briefly describe what it does, list the purpose and contents of the parameters, and document the returned result. Don't forget error conditions. For Java, use `javadoc`.
- Catch errors, and give meaningful error messages. Don't abort the program immediately. Try to repair the situation.
- Pick a coding style and stick to it. Use sensible variable and function names.
- Check the validity of the actual parameter values passed to non-private functions (e.g. range checks, `null`-pointers, etc.).
- Use assertions and logging. Offer the user the option to switch them on or off.

See also [Sta01].

2.7 Confidentiality

Assignments are not confidential.

3 Judgement

At the end of the project you will be graded for your work. The final grade can be one of the following.

Grade	Significance
5	Fail ("onvoldoende")
6	With satisfaction ("voldoende")
7	Average ("ruim voldoende")
8	Good ("goed")
9	Very good ("zeer goed")

Table 1. Grades, and what they mean.

On average, most students will obtain a 7 as a final grade. If you're heading for a 5, I will have told you much earlier...

3.1 Scoring

For external assignments, the opinion of the external supervisor is more important than my own. The final grade depends on your score on the following items.

Content and results Expertise, Scientific attitude, Abstraction, and Scope.
Thesis Elaboration of problem statement, Subject area, Organisation, Argumentation and substantiation, Accountability and research integrity, Writing style, Design and presentation.
Other products Quality.
Oral presentation Substantiation, Coherence, Completeness, Structure, Attractiveness, Interaction, Presentation technique.
Effort Preparation, Progress, Communication, Method of working.

References

Sta01. STALLMAN, R. Gnu coding standards. Tech. rep., Free Software Foundation, 2001. http://www.gnu.org/prep/standards_toc.html.