PORTFOLIO

Assignments for the course: Project: Computer Science Project (DLMCSPCSP01)

TABLE OF CONTENTS

1. TOPICS AND TASKS

A significant part of computer science is the development of machines that efficiently map an input (the problem) to an output (the solution). The machine is a computational device that uses one or many methods to transform input to outputs. Computer scientists are in the business of developing appropriate and efficient methods (processes, software, algorithms, hardware) that solve problems.

How do we apply computational methods in the real-world? In reality, we are using computational methods for almost all aspects of life, especially in those areas where automation can make our lives' easier. We build machines to increase efficiency of otherwise tedious, manual problems. Call centers use voice recognition algorithm to securely identify a customer over the phone using voice biometrics. Voice recognition in this use case automates several minutes of conversation between a customer service agent and the customer to verify the customers identity. Face ID simplifies the authentication time for a user logging into a smartphone.

To do so successfully, computer scientists invent tools, such as programming languages or compilers, that tend to have sophisticated algorithmic foundations. These tools evolve as people start using them. For example, object-oriented programming started with a simple question: instead of using "types" as prescribed by the programming language, can a programmer implement their own types (i.e. classes). Do we need compilers, or can we build a program once that will run on any other system (interpreters)? Can we build a method that can process sound signals such that voice identification system will take a few seconds to assign a unique identifier to a human voice rather than 30?

The real-world is complex and computer science strives to find elegance, repeatable solutions to solve complex problems. The aforementioned topics are merely examples. This project allows you to think of a real-world problem worth solving. The choice of project is yours. You can think of investigating problems that have already been solved and dig deep into the computational machines and methods used to solve a particular real-world problem. Or you can investigate a problem you care about deeply and attempt to solve it using the tools of computer sciences.

1.1. Task

Task: Apply computer science techniques to a real-world problem of your own choice. Select and work on a problem that inspires you. The goal is to present an in-depth research report on how you addressed the solution to the problem. You are expected to implement some part of your solution by either developing code or providing a mathematical analysis of your solution or both.

To decide on the exact scope of your project you first need to investigate the context of the area you want to work in. Subsequently, it is advisable to breakdown the problem into a smaller number of appropriately difficult problems and use the project as a vehicle to explore solutions to them.

We expect you to competently use advanced concepts and methods as you have learned them in your 1st semester masters' program. For example, a project that contains an exposition of a recent development in data science should contain an original presentation of the relevant theory in your own words, e.g. with an interesting, new example or helpful notation.

There are no constraints to your creativity. The main deliverable is a research report.

Work on your problem in the following 3 phases:

1.1.1. Conception phase

This phase is for you to settle on a project idea. Think about an interesting problem that you might have encountered in real life and ask yourself, if there is a clever technology solution you could explore. Or think about an interesting solution that you experience in your daily life and always wondered how it actually works.

Your deliverable for Phase 1 is an **abstract**, not more than ½ of a Din A4 page (approximately 200 words). This abstract illustrates the problem you want to address, how you want to address it and what you expect the outcome to be.

For your submission, please submit on PebblePad as a **PDF** no longer than ½ of a Din A4 page. The text field in the PebblePad template remains empty.

Throughout the process, online tutorials are offered, and they provide an opportunity to talk, share ideas and/or drafts, and obtain feedback. In the online tutorials, exemplary work can be discussed with the tutor. Here, everyone has the opportunity to get involved and learn from each other's feedback. **It is recommended to make use of these channels to avoid errors and to make improvements.** You should only submit work after making use of the above-mentioned tutorial and informative media. This will be followed by a feedback from the tutor and the work on the second phase can begin.

1.1.2. Development phase/reflection phase

This phase is for you to get started on addressing the selected problem. You should decide on the appropriate technology (i.e. programming language, libraries to use etc.) to develop your solution idea. Appropriateness is important. For example, if your idea requires you to build a compiler, doing so on top of Python or a programming language you are very familiar with. If you want to explore Machine Learning, you might want to use Python and scikit-learn to keep it simple.

Your deliverable for Phase 2 is a **draft of your research report**. The project report is your main deliverable after phase 3, hence, you should get started here. The project report should be considered a technical document readable by a computer scientist who is not a specialist in this topic. The structure could be as follows:

- Abstract (e.g. revision of concept phase deliverable)
- Introduction (explain the problem or the context of the problem)
- Related work (who else has looked at the problem and what did they find)
- Technical background (what are the technical underpinnings of your problem)
- Method (what is the method you are using to solve the problem e.g. algorithmically, software design, etc.)
- Implementation (how did you implement the solution)
- Testing (how did you verify that your solution works)
- Conclusion (summarize and write about what else you could do if time permits)
- Bibliography and list of references

Should you be developing code in the process, please create a public GitHub repository. Any code assets prepared in this phase should be posted on GitHub, either now, or latest at the end of your project. We recommend you posting your status-quo in terms of assets on GitHub in this phase, but this is not mandatory.

Should you be developing a mathematical analysis (e.g. a complexity analysis of an algorithm you are investigating) you should mention this in your Method section.

Please note that the goal of this project is to go deep into using tools of computer science to investigate a problem. Make sure you understand which tools of computer science you want to use and clearly articulate in the draft.

For your submission, please submit your **draft report on PebblePad as a PDF**. Submit the **link to your GitHub project** into the PebblePad text field.

You merely need to prepare a draft of this report. This will help the tutor understand where you stand, what you might be struggling with and how to help you.

Throughout the process, online tutorials and other channels provide the opportunity to profoundly discuss ideas and/or drafts and to get sufficient feedback, tips, and hints. **It is recommended to use these channels to avoid errors and to improve your work.** Once this is done, you can hand in your second phase for evaluation. Following a feedback from the tutor, your work on the final draft will continue in the third phase.

1.1.3. Finalization phase

In the finalization phase, Phase 3, you will complete your research report after having received feedback from the tutor. Certain elements may have to be improved or changed again to finalize the task and complete the portfolio course. It is important that you consider the feedback that you received on your submission at the end of the previous phase.

You will submit an **abstract**, in which you can reflect on the process of developing your project, insights and potential aspects you would do differently, if you had to do it again.

The **main deliverable is the research report**.

Should you have code assets as a deliverable and you have not posted the code assets in GitHub yet, please do so by creating a public GitHub account, which will include:

1. Your code folders, including all code + libraries necessary to deploy and run the web app
2. Provide installation and run instructions in the README.md file in your GitHub repository

Please include the link to your GitHub repository in the research report.

Your deliverable for Phase 3 is the **final version of your research report and the abstract**. The research report should not exceed 30 pages including references.

For your submission:

1. Upload your **abstract as a PDF file** in PebblePad.
2. Attach the **research report as a PDF file with a link to your public GitHub repository**
3. Export your GitHub repository as a ZIP file and put it into **a folder**. You must zip this folder and upload it to your submission in PebblePad.

In the “Finalization phase”, the online tutorials and other channels also provide the opportunity to obtain sufficient feedback, tips, and hints before the finished product is finally handed in. **It is recommended to use these channels to avoid errors and to make improvements**. The finished product is submitted with the results from Phase 1 and Phase 2 and together with the materials mentioned above. Following the submission of the third portfolio page, the tutor submits the final feedback which includes evaluation and scoring within six weeks.

2. TUTORIAL SUPPORT

In principle, several channels are open to attain feedback for the portfolios. The respective use is the sole responsibility of the user. The independent development of a product and the work on the respective portfolio parts is part of the examination performance and is included in the overall assessment.

On the one hand, the tutorial support provides feedback loops on the portfolio parts to be submitted in the context of the conception phase as well as the development and reflection phase. The feedback takes place within the framework of a submission of the respective part of the portfolio. In addition, regular online tutorials are offered. These provide you with an opportunity to ask any questions regarding the processing of the portfolio and to discuss other issues with the tutor. The tutor is also available for technical consultations as well as for formal and general questions regarding the procedure for portfolio management.

Technical questions regarding the use of “PebblePad” should be directed to the exam office via mail.

3. EVALUATION

The following criteria are used to evaluate the portfolio with the percentage indicated in each case:

Evaluation criteria	Explanation	Weighting
Problem Solving Techniques	*Capturing the problem *Clear problem definition/objective *Understandable concept	10%
Methodology/Ideas/Procedure	*Appropriate transfer of theories/models *Clear information about the chosen Methodology/Idea/Procedure	20%
Quality of implementation	*Quality of implementation and documentation	40%
Creativity/Correctness	*Creativity of the solution approach *Solution implemented fulfils intended objective	20%
Formal requirements	* Compliance with formal requirements	10%

The design and construction of the portfolio should take into account the above evaluation criteria, including the following explanations:

Problem Solving Techniques: Clear scope as well as comprehensibility of the project. Use of appropriate computer science methods to solve a problem.

Methodology/Idea/Procedure: Appropriate application of theories or models and clear articulation about the choices made to analyze and solve the project problem.

Quality of implementation: Visible concepts of computer science, clear articulation of the solution pathway. Clear position of work in the context of related work. Sufficient depth in approach and analysis.

Creativity/Correctness: Originality of solution. Correctness of solution as verified by testing.

Formal requirements: The submission follows the acceptance criteria from Chapter 3 and the formal guidelines following in the next chapter. It is particularly important to respect the formal submission requirements outlined in Chapter 4.

4. FORMAL GUIDELINES AND SPECIFICATIONS FOR SUBMISSION

4.1. Components of the examination performance

The following is an overview of the examination performance portfolio with its individual phases, individual performances to be submitted, and feedback stages at one glance. A template in “PebblePad” is provided for the development of the portfolio parts within the scope of the examination performance. The presentation is part of this examination.

Stage	Intermediate result	Performance to be submitted
Conception phase	Portfolio part 1	• Abstract (½ Din A4 page, PDF) including: ○ Problem statement ○ How the problem will be solved ○ What is the expected outcome
		Feedback
Development phase/ reflection phase	Portfolio part 2	• Draft of the project report as PDF including ○ Abstract (e.g. revision of concept phase deliverable) ○ Introduction (explain the problem or the context of the problem) ○ Related work (who else has looked at the problem and what did they find) ○ Technical background (what are the technical underpinnings of your problem) ○ Method (what is the method you are using to solve the problem e.g. algorithmically, software design, etc.) ○ Implementation (how did you implement the solution) ○ Testing (how did you verify that your solution works) ○ Conclusion (summarize and write about what else you could do if time permits) ○ Bibliography and list of references • Link to GitHub repository in the PebblePad text field
		Feedback
Finalization phase	Portfolio part 3	• Abstract maximum of 2 pages in PDF • Research report maximum of 30 pages in pdf (including a link to your public GitHub repository) • Upload the zip folder (including the export of GitHub repository) • Result from phase 1 • Result from phase 2

Feedback + Grade

4.2. Format for Digital File Submission

Conception phase

Recommended tools/software for processing	Word, LaTex
Permitted file formats	PDF
File size	as small as possible
Further formalities and parameters	Files must always be named according to the following pattern: **For the performance-relevant submissions on "PebblePad":** Name-FirstName_MatrNo_Course_P(hase)-1_S(ubmission) Example: Mustermann-Max_12345678_PCS_P1_S

Development/reflection phase

Recommended tools/software for processing	Word, LaTex Coding platform or other technical tools as necessary
Permitted file formats	PDF
File size	as small as possible
Further formalities and parameters	Files must always be named according to the following pattern: **For the performance-relevant submissions on "PebblePad":** Name-FirstName_MatrNo_Course_P(hase)-2_S(ubmission) Example: Mustermann-Max_12345678_ PCS _P2_S In case you have a GitHub repo at this stage, please add the link into the PebblePad text-field

Finalization phase

Recommended tools/software for processing	Word, LaTex Coding platform or other technical tools as necessary
Permitted file formats	PDF
File size	as small as possible
Further formalities and parameters	**IMPORTANT** is the upload of the zip folder that has been created especially for the submission (please follow the instructions on myCampus). This folder contains all the files you used to complete the task. To ensure a better overview, please create subdirectories for this purpose. **The folder structure then looks like this:** ○ Main directory (name of the zip folder) -> *Name: Name-First_Name_Matriculation_Course* • Subdirectory (your research report)-> *Name: 01-research_report* • Subdirectory (code assets as a zip export of GitHub repo) -> *name: 02-code* Please make sure that you either embed the images (and fonts, if any) linked in your document or to place them in the respective directory. Otherwise your documents cannot be opened completely and therefore cannot be assessed! Files must always be named according to the following pattern: **For the performance-relevant submissions on "PebblePad":** Name-FirstName_MatrNo_ Course _P(hase)-3_S(ubmission) Example: Mustermann-Max_12345678_ PCS_P3_S

4.3. Format of Abstract

Length	At most 2 pages
Paper size	DIN A4
Margins	Top and bottom 2cm; left 2cm; right 2cm
Font	General Text - Arial 11 pt.; Headings - 12 pt., Justify
Line Spacing	1,5
Sentences	Justified; hyphenation
Footnotes	Arial 10 pt., Justify
Paragraphs	According to mental structure - 6 pt. after line break
Affidavit	The affidavit shall be made in electronic form via "myCampus". No submission of the examination performance is possible before it. Please follow the instructions for submitting a portfolio on "myCampus".

If you have any questions regarding the submission of the portfolio, please contact the exam office via mail.

Please also note the instructions for using PebblePad & Atlas!

Good luck creating your portfolio!