Advanced HCI - Project 2024-2025

Crafting the digital education of the future

The recent advance of novel technologies such as artificial intelligence, robotics, virtual reality, cloud computing, and so on has accelerated the digital transformation of the world. Such a transformation has impacted several spheres of society by opening new opportunities in people's lives as well as shaping new challenges to be faced. Everyday exposure to digital data, indeed, could amplify biases as well as increase vulnerability or exclusion.
Education and training systems are not an exception: they are the incubators to create the world of the future *"providing young people with forward-looking knowledge, skills and competences they need to innovate and prosper"*. The impact of digitalisation in the education landscape is resulting in a reshaping of teaching and learning methods in a variety of ways.
Recently, the European Commission defined an action plan having as its focus *"the implementation and the need to stimulate, support and scale up purposeful use of digital and innovative education practices"*. Three pillars were identified:

1- Making better use of digital technology for teaching and learning;
2- Developing relevant digital competences and skills for the digital transformation;
3- Improving education through better data analysis and foresight.

In such a framework, your team is asked to imagine a technological solution that could be the game changer for the next generation of education and training tools.
Stemming from the three pillar mentioned above, your solution should intervene in one of the following areas:

a - Enriching the teaching / learning experience: possible topics within this area cover peer learning, peer assessment, personalized learning, active learning, and so on;
b - Fostering individual competences: possible topics within this area cover critical thinking, creative use of technology, addressing disinformation and online safety, computational thinking, programming, and so on;
c - Promoting well-being in digital education: possible topics within this area cover inclusive learning experiences, mental health and self-care, and so on;
d - Data-driven education: possible topics within this area cover monitoring progress, interactive analysis of data;
e - Other: everything is not covered by the other areas but you think that is relevant!

Your work

Before starting your work, please read carefully the instructions provided in the Instructions Section at the end of this document.

Alone or in a small team

You can choose to work alone or in a small team (max 3 students). In this last case, during the presentations of your project, take care that each student of the team has adequate time to explain their contribution. **The contribution of each student has to be clear.**

Part I: Problem and Solution

Choose **one** of the previously mentioned 5 fields. Identify a specific problem that can occur in this field and that you wish to address. Propose **two different** technological solutions to solve the problem specifying at least 3 pros and 3 cons of each one.

Make a draft list (high level, no technical details are required at this stage!) of the functionalities offered by the solutions explaining how they concur to solve the problem.

Each solution **must** exploit 3 modalities: modalities can either be used, for example, one after the other or in parallel; at least one functionality of your technological solution must be supported by two or more modalities to take into account user preferences; a modality can be used for example to control error …what you want! About the functionality that is supported by two or more modalities, explain how one or more of the strategies of the ARCADE model could be used to help user choices.

Please, notice that when you conceive your ideas, you do not need to exactly know all the technical details of the implementation.

If you prefer, you can also choose two different fields and propose 1 solution for each field.

Complete a small-scale literature review on how researchers already dealt with the problem you chose. The review should be a descriptive and critical summary of previous research in the field and it should include **at least 3 scientific references (at least 4 for the teams composed by 3 people)** that have to be properly cited in your presentation. This review is aimed at informing readers of the knowledge that has been established and technological solutions that have been adopted on that problem. Highlight the novelty of your solution with respect to this previous work.

Part II: Go deeper in your idea!

Following the discussion you had after the presentation of Part I, choose the idea you think is the most promising, and refine your work including the feedback you received, and possible further material you found.

Identify needs and goals of people expected to use the technological solution you are proposing. To do that, ground your analysis on existing data, that is sources already available such as statistics, surveys or market analysis reports and so on. From such material, derive a prototypical user profile (i.e. a *persona*), and a scenario of how your solution works. List the possible stakeholders for your solution.

Identify hardware and software tools that you hypothesize to use to develop your solution, not necessarily limited to what we studied during the lectures. More specifically, make a list of the hardware components needed to develop your solution by also identifying the technical specifications more suitable in your scenario (e.g. spatial and temporal resolution and so on). Design a *UML component diagram* of your solution illustrating how components are wired together to build your technological solution.

Refine the draft list of functionalities adding more details about they concretely work and how they could be implemented (no implementation is required).

Part III:

Design a *UML state chart diagram* (i.e. a diagram of the finite-state-machine governing the game) of your solution to model its behavior and its states. Do not forget to find ways to manage errors.

Now, choose at least one functionality of your solution and implement it as a fake real-time solution using Python or another programming language, as you wish. Fake real-time means that you will work with pre-record inputs (that is files) and you will read them as they were acquired on the fly.

Compute the features used by the chosen functionality on time windows. These windows should have an adequate size and can be consecutive or overlapped according to the goals you want to achieve. To select size and overlapping, take care of the output sampling frequency you need. Concretely, time windows can be implemented as arrays or matrices.

Analyze the features computed on the pre-recorded data to implement the required functionalities. You can use e.g. thresholds, rule-based approaches, simple machine learning, pre-trained models and so on. Every student has to work using his/her own data (e.g. gesture trajectories, facial / body landmarks, AUs, speech features and so on), that is he/she must not record/collect data of other people. Describe the results of your approach (e.g. plots, tables, stats, and so on).

Instructions

Deadlines

15 November 2024 - Presentation of your work on Part I to the other students and to the teacher. The presentation has to be structured as follows: a PowerPoint (or similar) presentation in which **1** slide will briefly describe the field you decided to address, **1** slide will present the problem you want to solve, **6** slides will report the literature review (2 for each reference), **5** slides describing your solutions, their functionalities and the ARCADE strategies you think to adopt, **1** slide with issues to be faced for developing the idea, **1** slide with references. The goal of Part I is to present your thoughts and solutions and to discuss them for possible improvements. The duration of your presentation should last **15 minutes**.

13 December 2024 - Presentation of your work on Part II to the other students and to the teacher. The presentation has to be structured as follows: a PowerPoint (or similar) presentation in which **1** slide will present your final idea, **1** slide will show the improvements with respect to your previous presentation, **1** slide on the methodology employed to identify needs and goals, **1** slide reporting the persona, **1-2** slides describing the scenario as a short text or as a storyboard, **2-4** slides with the refined list of functionalities, **2-3** slides about the hardware and software tools, **1** slide will depict the UML component diagram, **1** slide with references. The duration of your presentation should last **20 minutes**.

At the exam call (January, February in the first semester; June, July, September in the second semester) - Final Presentation of your whole work. The presentation has to be structured as follows: a PowerPoint (or similar) presentation in which **1** slide will present the field you address and the problem you want solve, **2** slides about literature review, **1** slide will describe your solution, **3-4** slides about the functionalities of your solution, **1** slide will report the persona and how you identified its needs and goals, **1-2** slides will describe the scenario, **1** slide will provide the list of the hardware components (and their technical specifications) you need to develop your solution, from **1** to **3** slides on technical tools, **1** slide will depict the UML component diagram, **1** slide will depict the UML state chart diagram, **3-4** slides about the functionality implementation and results, **1** slide will be about conclusions and lessons learnt, **1** slide with references. After the presentation, you will have to show a demo of the functionality you implemented. The duration of your presentation should last at maximum **30 minutes**. The duration of the demo should be around **10 minutes**.

Two days before the exam call, upload to the Project section of Moodle a .zip file with your final work. The name of the file must be: **Lastname_student(s).zip**.
The folder has to contain:

1) your final presentation (pdf format);
2) a folder with your code (no data!)

Your final grade will be assigned based on your actual contribution to the project, your skills at mastering the presented topic, and your answers to questions on the course topics.