Antescofo: Project Title

Martin Aigner

Computational Systems Group
University of Salzburg
firstname.lastname@cs.uni-salzburg.at

1. Foreword

Writing this report is part of the requirements of [CLASSNAME] intended to make students think about combining different topics of education through a single common property: the use of computers.

Say here, that this is important because...

The goal is to propose a hypothetical project. (to learn this and that)

The cool thing here: this project has actually happened. The drawback: it is no longer a hypothetical project and therefore, kind of, contradicts the goal of thinking about: "How would I plan and lead that project?" Nevertheless, we did plan the project in advance as much as possible and clearly state any changes we have applied to *the plan* during the project. Still, we have improvised a lot, e.g., making up on-demand mini lectures on the fly. The interested reader might wonder if improvisation further contradicts the idea of planning and describing a hypothetical project. We don't think so! A teachers ability to adapt to individual student's knowledge, needs, and interests is, in our opinion, a key quality to have in the educational business.

Our goal with this report is simple. We want to enable others to repeat the project under similar circumstances. The students achieved a great result which can be viewed at `https://youtu.be/a_AVsBpvBVo`

2. Abstract

(Context) The Computational Systems Group Salzburg is involved in a research project on Antescofo, a real-time multimedia system, developed by IRCAM, Paris. Antescofo is a complex piece of software used to accompany musicians and orchestras on the stage. It is used at various concert halls throughout the world, including the Festspielhaus in Salzburg. We have recently submitted a research proposal with IRCAM on advancing the real-time aspects of Antescofo for embedded devices.

(Internship) The task of the students within this internship is to setup, use, (and so performance analysis of Antescofo. Some of the challenges of Antescofo (are scalability, as well as proper modelling of time, topics that our (research group has expertise on. The students are expected to get Antescofo (running in a lab environment, demonstrate simple use with an actual (instrument, and isolate performance issues that motivate our research). This (internship project will be a valuable kick-off for our research on enhancing (the real-time aspects of Antescofo.

(technical bla-bla) Assets for the students: experience working on a highly (sophisticated software system; get acquainted with technical issues of (setting up a system; experience with performance analysis and with research (on real-time aspects of computing; fun with music and complex software.

3. Project Goals

Combine computer science, art, and music.

Making music with computers.

Present your work! Since *video killed the radio star* we will do a music video that wraps up our project.

4. Tools

This section gives a high level introduction to the software products used in this projects.

4.1 Pure Data

“Pure Data (aka Pd) is an open source visual programming language. Pd enables musicians, visual artists, performers, researchers, and developers to create software graphically, without writing lines of code. Pd is used to process and generate sound, video, 2D/3D graphics, and interface sensors, input devices, and MIDI. Pd can easily work over local and remote networks to integrate wearable technology, motor systems, lighting rigs, and other equipment. Pd is suitable for learning basic multimedia processing and visual programming methods as well as for realizing complex systems for large-scale projects.” [1]

4.2 Antescofo

“Antescofo is a modular polyphonic Score Following system as well as a Synchronous Programming language for musical composition. The module allows for automatic recognition of music score position and tempo from a realtime audio Stream coming from performer(s), making it possible to synchronize an instrumental performance with computer realized elements. The synchronous language within Antescofo allows flexible writing of time and interaction in computer music.” [2]

4.3 Logic

4.4 iMovie

5. Tasks

This section describes the tasks we assign to the students.

5.1 Getting started with Pure Data

Antescofo is implemented in - and controlled through - Pure Data. Installing Pure Data is as simple as installing any OSX application. Details can be found on the Pure Data website and are not repeated here[1].

A Pure Data application is called a patch. A patch has a graphical Pure Data window

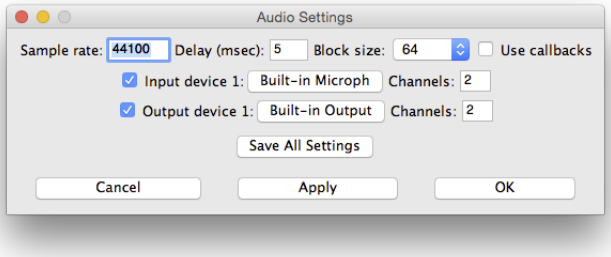

Figure 1: Pure Data Audio Settings

5.2 Implement your first Pure Data Patch

Do something nice with Pure Data

5.3 Getting started with Antescofo

Hand out the manual.

Understand events and actions.

[1] https://puredata.info/downloads

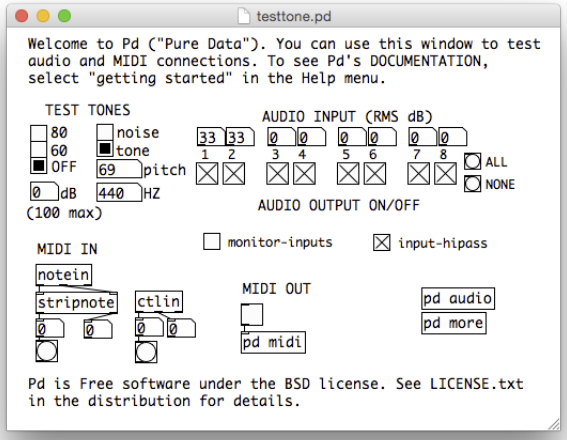

Figure 2: Pure Data Audio Test Patch

6. Time Table

The time frame for the project is approximately 2 weeks, 6 hours per day, or 60 hours. Note that the time table is heavily affected by the students' prior knowledge in programming. Table REFERENCE gives a brief overview of the suggested time required for each individual project task.

4 weeks, preparation classes, prerequisites

7. Acknowledgments

The project was supported by the Austrian Research Promotion Agency (FFG) TODO: add grant number

References

[1] "Pure data," September 2015. [Online]. Available: https://puredata.info

[2] "Antescofo," September 2015. [Online]. Available: http://repmus.ircam.fr/antescofo

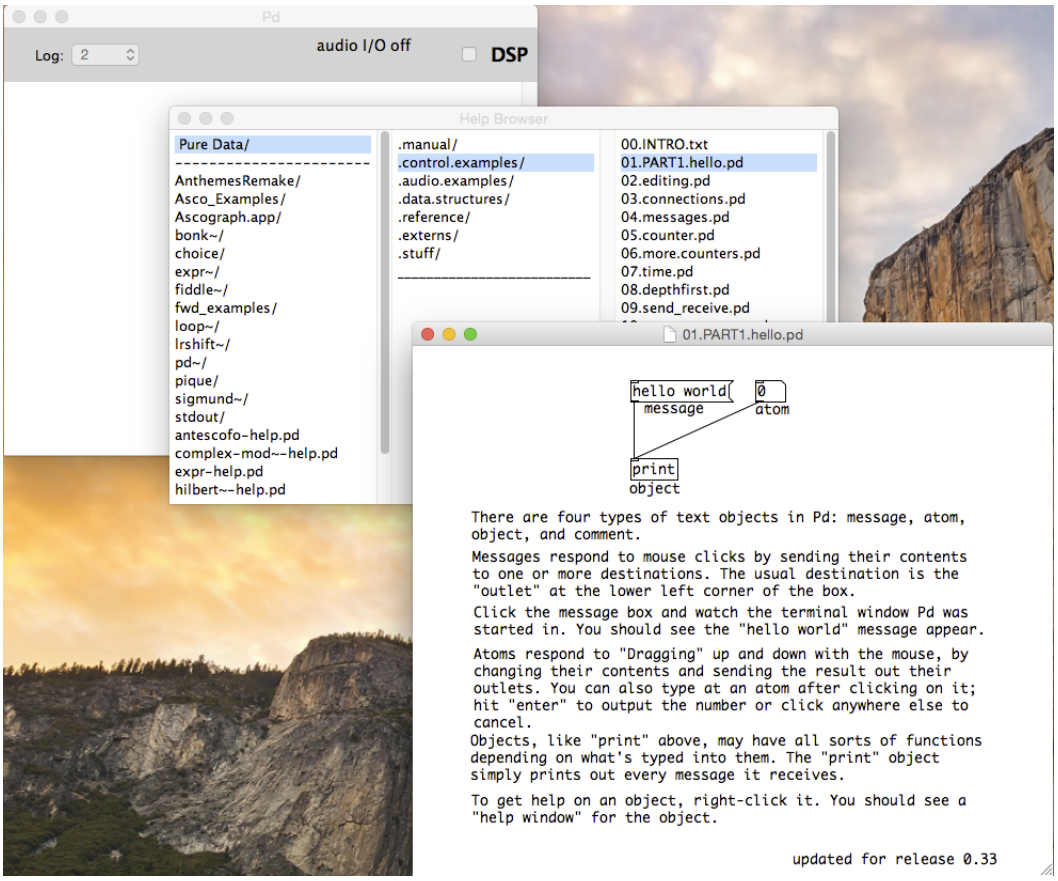

Figure 3: Pure Data Help Browser