

Assessment guidelines



In this first part of Technological Basics you will concentrate on **conceptualizing, drafting out and prototyping a possible application** based on the knowledge of Python that you acquired in this course.

This application *could* be:

- a text based game
- a website (with the Flask framework)
- a desktop application operating a specific database (your music / book / movie collection)
- an artistic project experimenting with text

At this stage your final project for TB1 consists of a **written presentation** of your idea (approx. 5 pages). You will already have **code fragments** integrated in your project pitch which represent some core mechanisms of your application. This should have only prototypical character. (If you have for example a dictionary from which you want to retrieve data entries representing your, let's say, record collection, you will only have a dummy dictionary, containing some few entries. The basic principle of your functions should be visible though.)

Your presentation will be accompanied by a **collection of source code of comparable applications**. You should try to analyze and describe the source code and its parts as differentiated as possible and explain which parts represent similar functionalities in relation to what you want to build.

This 5 - xx pages long pitch should contain the following parts:

- presentation of the main idea
- description of functionalities
- description of the main structure (you could use a flow chart or some other visualization for this part, describing f.i. system-user-interaction, the way the data flows in your app or any other kind of visualization of structural aspects of your app)
- description of used modules / libraries / frameworks
- possible extensions of the currently feasible; isolation of currently unfeasible parts, inference of tools, skills, knowledge necessary for accomplishment, estimation of time required to acquire those skills

To resume: the main objectives of this part of „Technological Basis“ upon whose successful accomplishment you will be graded are the following:

1. Show the widest possible spectrum of **programming skills** that you acquired during this class.
2. Develop an **imagination** regarding possible applications for your **programming abilities**. You should train your creative abilities in regard to a realistic project which you can possibly build in the time given. You should be able to isolate parts that aren't feasible yet for you at the very moment and develop realistic strategies for future implementation. This implies to infer particular functionalities from a more general **concept**, identify concrete mechanisms and functions and research available frameworks, libraries or modules for its implementation.
3. Learn how to research comparable projects, train your **code literacy** in reading and analyzing other programmer's source code, be able to get an overview about existing applications in order to better **qualify the innovation factor / originality of your own work** and push further the existing state-of-the-art.

NOTE: The hereby developed project should be ideally completed by you in the following semester. As expressed in the above guidelines, your draft not only can, but should contain ideas for further extensions, which are not yet feasible given your current state of knowledge. The ideal proportion here should be around 50:50. Nevertheless you will be graded upon the overall accomplished effective effort. This means: if your application is of extensive dimension and you succeeded to only build 20 percent, this is still acceptable given the case that the difficulty and effort of the delivered work is reasonable in view of the available time.

Another tip: Bear in mind that next semester we will learn new things which will give you the hands-on expertise to extend your project in various ways: we will have a look at how to build GUIs, work with images and generate graphic content with Python, visualize data and will finally start working with hardware.

Try to think in these directions, f.i. imagine how you could potentially extend your project with hardware. This does not have to make sense in a strictly pragmatic real-world application but can have an artistic and playful character, without any pragmatic value. For this course it is mostly important that you show the widest spectrum of technical skill and good versatility in applying it. To help develop a possible idea how to interface your application with hardware, think about in- and output devices that could be connected to your app and control processes of your application or vice-versa use some data your app is outputting. This could imply audio / sound, video / light, mechanical triggers etc.

These websites can help you getting inspired regarding the possibilities of using technology in a creative way:

www.instructables.com

www.makezine.com

www.hackster.io

Last but not least: Try to build something that is fun for you or has a good practical purpose. This will guarantee that you will work on it with joy and solid motivation! ;)