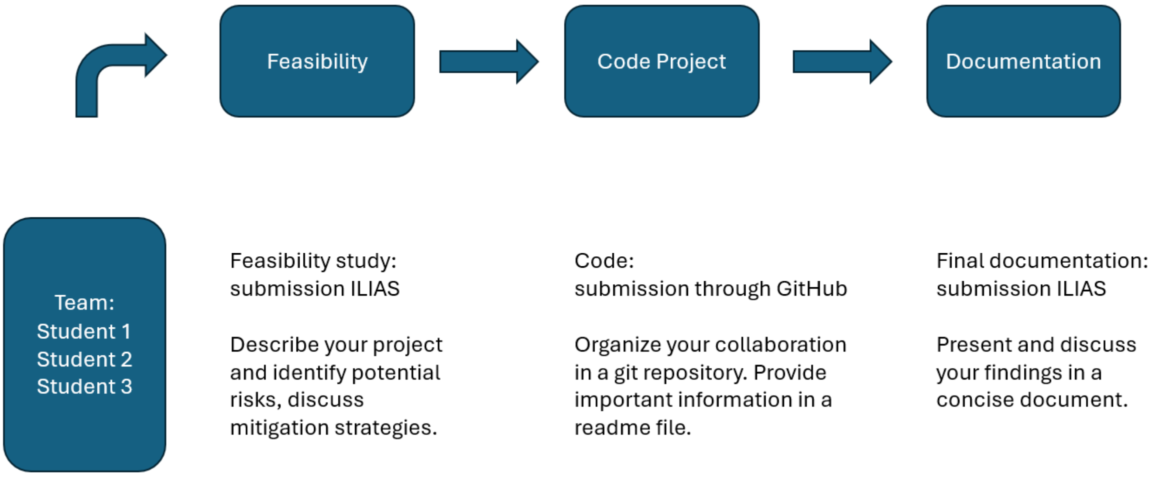
CIP – Project Description

Introduction:

In this CIP course, you get the opportunity to apply the techniques and tools introduced in the PDS and CIP courses in a group project. As a team of 3 students, you will go through a typical data science project journey.

First, you will choose a topic with public data that you are interested in. Find an attractive topic with plenty of data available that might expose new information in new combinations and might be of interest for data science purposes. In a second step, you elaborate a feasibility study that summarizes your project including three research questions you want to answer with the gathered data. Eventually, the actual project should allow you to apply and extend your python skills. It contains the acquisition of the data, its preparation and cleaning as well as the answering of your project questions. The final documentation where you discuss and present your findings will wrap up your work.

Keep in mind: the focus of the project should lie in the application of PYTHON!



Feasibility study:
submission ILIAS

Describe your project and identify potential risks, discuss mitigation strategies.

Code:
submission through GitHub

Organize your collaboration in a git repository. Provide important information in a readme file.

Final documentation:
submission ILIAS

Present and discuss your findings in a concise document.

Figure 1: Project overview

Detailed Task Description

Feasibility Study

Your project begins with writing a feasibility study that provides a concise evaluation of the viability of your project. Start with a brief introduction outlining the projects scope and purpose. Pose at least three research questions that you plan to answer using Python and collected data. Identify and describe the key data sources you plan to use. Discuss potential risks related to data collection and quality, that could impact the project's success and suggest backup plans or alternative strategies to mitigate these risks if they arise. The feasibility study should not exceed **two pages**, ensuring that content is concise and focused.

The submission of a feasibility study allows the lecturers to give you feedback on how suitable your idea already is and gives the opportunity to offer advice on how it may be improved.

Content Phase

Data Acquisition:

You must organize the data providing the necessary information to answer your research questions yourself. Whether you use web scraping, API's or even simulations as your main data source is up to you. Ideally, you find data from various locations that have complementary character and provide the potential to create new insights through combining the different subsets.

Scraping data implies gathering publicly available data from the web. It lies in your responsibility respect the legal boundary conditions, please always consult "robots.txt" when using automating tools. We recommend considering the following remarks:

- Find and interact with at least one dynamic element (search bar, buttons, list scroll, filter, ...) on the website.
- Use Selenium and BeautifulSoup

If you are using APIs as your main data source, make sure to consult their documentation and try different requests. Since the effort to acquire data via APIs is much lower compared to scraping, generally bigger / more complex datasets are expected.

In case you are interested in creating/simulating your own data set, make sure to adequately describe this process in your final documentation as well.

Data Transformation / Cleansing:

API-, scraped- or simulated-data are often already "quite clean". Since in most real-world scenarios we are far from that situation, we would like to highlight the importance of this part. You need to be able to handle poor data quality. For this reason, we expect you to address the following list of mandatory transformation / cleansing steps, also if some are not directly meaningful in your context:

- Check for gaps / missing data
- Check if columns show appropriate datatypes, change if needed
- Check if values lie in the expected range
- Identify outliers, treat them reasonably
- Format your dataset suitable for your task (combine, merge, resample, …)
- Enrich your dataset with at least one column of helpful additional information

Analysis and Visualization:

After transforming your original data sets into usable formats, use python to answer your research questions and visualize the results with tables or figures.
Try out different visualization frameworks, choose whatever framework you are interested in the most. A remark for unexperienced programmers: it can be beneficial to stick to one framework, reducing the initial efforts required for getting familiar with multiple frameworks. Keep in mind though, no matter which frameworks you use, there are objective evaluation criteria for adequate visual representation, as e.g. axis labels (units), readability, captions, legends, etc.

Final Documentation

Prepare a comprehensive final documentation that effectively summarizes your project. The documentation should include an introduction that provides a short overview of the project and present your motivation. The methods section should describe your data sources / creation, the transformation/cleaning steps and the analysis methods applied. The third section is the discussion of your results, where the focus should be. Present here your findings and discuss your results. At least two different types of visualizations are requested. Finally, the conclusion should give a short summary about the learnings, limitations and give an outlook on potential future steps to improve. The final documentation should not exceed **six pages**, ensuring that content is concise and focused.

Task Summary, Assessment

Description

After the feasibility-study, where you define your project goals/questions, the project is divided in three tasks:

1) data acquisition, crawler/API/simulation: requests, beautiful soup, selenium, etc.
2) data cleaning/transformation: pandas, numpy, etc.
3) presentation of results: matplotlib, seaborn, plotly etc.

Each of these three points needs to be addressed and will be evaluated. Nevertheless, the priority and degree of detail dedicated to the tasks do not have to be equal, set your focus on where you profit the most!

Deliverables:

ILIAS:

- Feasibility study (2 pages max.)
- Project documentation (max. 6 pages)
- Raw data as a zip-file

GitHub:

- git repository:
  - code
  - (short) README.md, where each student refers to their specific code contribution
  - repository naming conventions: CIP_[HS/FS][YEAR]_yourgroupnumber

Remark: relevant due dates are found in the course agenda.

Remark: Each group must submit the documents only once! (-> group number must be part of the file names)

Evaluation Criteria:

- Accepted feasibility required for final evaluation
- Completeness
- Documentation
- Complexity of (sub-) tasks
- Transformation/Cleansing approaches
- Quality of figures
- Code structure, readability
- Organization of git repository