CIP – Project HS2024 V01

Introduction:

In this CIP course, you get the opportunity to apply the techniques and tools introduced in the PDS and CIP courses (parts 1 and 2) in a group project. As a team of 3 students, you will go through an (adapted) ETL-process customized for this course.

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 questions you want to answer with the scraped data. As the final work, you extract, transform and store your data in a way that you can answer the questions from the feasibility study. The project consists of group work as well as individual contributions.

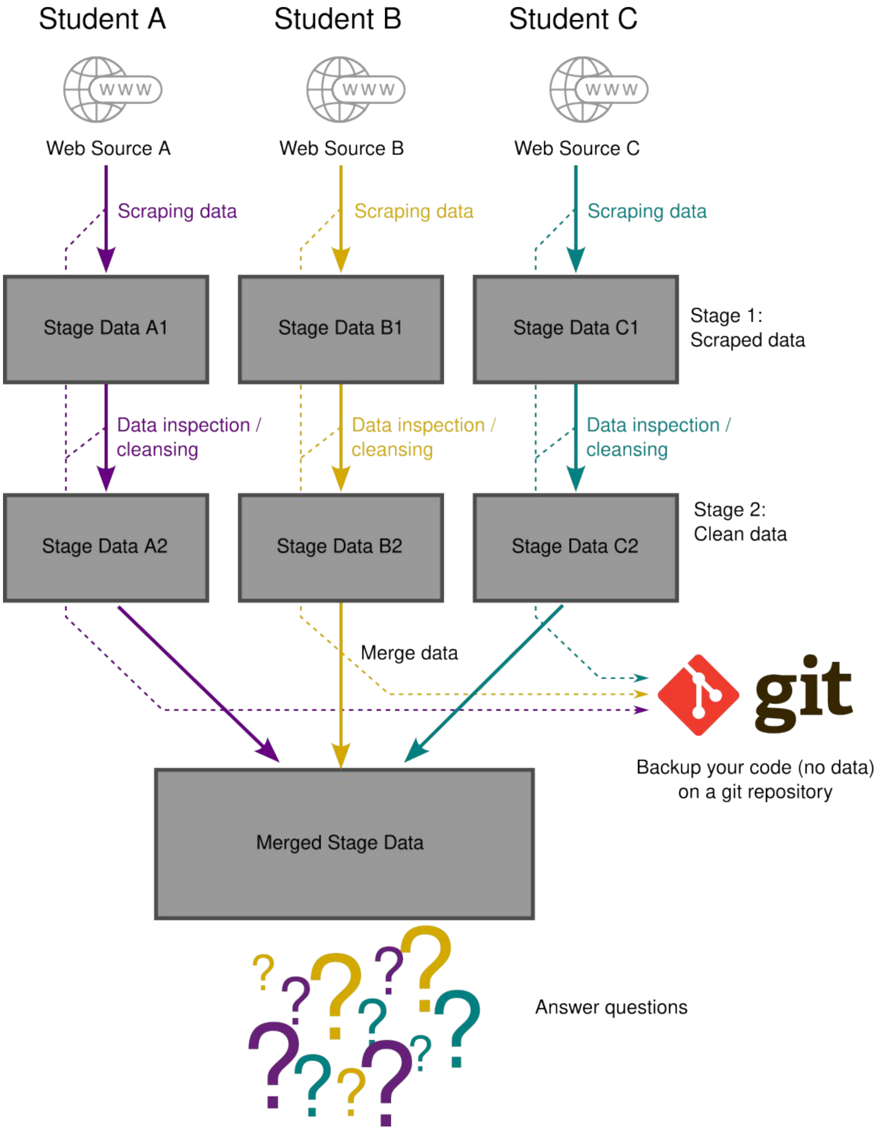

Figure 1: Project Overview

Detailed Task Description

Feasibility study

The first step is to write a feasibility study that provides a concise evaluation of the viability of your project. Begin with a brief introduction outlining the projects scope and purpose. Pose at least three questions that you plan to answer using Python and data scraped from the internet. 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. You can expect the feedback within 2 weeks.

Collaboration and Data Sharing

The focus of the project lies in the application of Python. That is also why the main contribution to your final grade stems from the review of your code. We expect your repository to be clean and necessary information to be easily accessible. Subfolders for each student are created and contain their respective work.

Extract phase:

Every student must extract data from a web source providing the relevant information for answering your questions. Ideally, data should have complementary character and provide new insights by combining the different data sets. Therefore, prove the data for merge feasibility in advance while elaborating the project narrative.

Scraping data implies gathering publicly available data from internet. In a simple form, such data are available as a list and extracting data gets reduced in complexity to a simple table download and extraction. Yet, in a real scenario, valuable data are usually hidden and not obvious at first glance. As a result, we would like to motivate you in this project to perform the data extraction on an advanced level including dynamic web pages - if applicable and useful. It lies in your responsibility respect the legal boundary conditions, please always consult "robots.txt" when using automating tools.

- Find and interact with at least one dynamic element (search bar, buttons, list scroll, filter, ...) on the website.
- Selenium and BeautifulSoup must be used in combination for scraping data from website.

Transform phase:

Scraped data are occasionally already "quite clean." In principle that's not a bad thing. However, "clean" and "very clean" data do not serve the purpose of our project. You should also be able to handle poor data quality after scraping. For this reason, it is mandatory to implement all of the following quality tests/transformations at least once on your scraped data set (every student!). Your project plans may as well require that you extend the list:

- 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

The cleaning and enrichment steps shall be as generic as possible. This means, avoiding single case operations by preferring functions that are applied on full data frames.

Group work phase: subfolders on Git for individual works

As a group, merge the different data sources provided from each student where reasonable and try to answer your project questions with the help of Python scripts. Data in the final format must be submitted through ILIAS as well. Create a git repository to collaborate and share all Python code through a git repository with a clear structure to separate individual work from group work.

Final Documentation

Prepare a comprehensive final documentation that effectively summarizes the project undertaken. The documentation should include an introduction that provides a clear overview of the project and present your motivation. The methods section should describe the data collection and transformation techniques used, and analysis methods employed during the project. In the results section, present the findings, summarizing key data, observations, and outcomes. While visualizations or tables are not mandatory, they can be included to enhance clarity. 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

Contributions

Individual contribution of each student:

- Scrape data from web sources with python
- Perform cleansing, transforming and preprocessing on the obtained data with python

Group contribution:

- Write a feasibility study
- Create a git repository to collect and share your code
- Answer your project questions with python
- Document your project

Deliverables (proportion contributing to final grade):

- Feasibility study (2 pages max.) **(10%)**
- Project documentation (ca. 4, max. 6 pages) **(20%)**
- Git repository containing: **(70%)**
  - Individual part for each student: Python scripts for scraping
  - Individual part for each student: Python scripts for cleansing and preprocessing
  - Group part: Git repository, Python scripts for merging data sources and transforming data to answer project questions (10% of total grade)
- Individual data as a zip-file on ilias (do not commit the data files! -> gitignore)

Remark: all Python code is expected to be well documented / commented

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