Praktikum Wirtschaftsinformatik
14. January 2026
Prof. Dr. Mirjam Minor
M.Sc. Tolga Tel

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

**In this exercise, we will put everything together by first generating a descriptive text, then extracting information and at last searching for mistakes to improve the generated text. We use multiple Pipelines to decrease the complexity of the problem.**
**If you decide to skip exercise 3, please prepare a presentation, showing how you solved the other exercises.**
**You can find all the documentation on https://haystack.deepset.ai/**

**Deadline: 26.01.2026, 10am**

## Exercise 1 (Pipelines) — 8 Points

a. Create a Pipeline (Generator), that is given a .bpmn file and generates a descriptive text for the given BPMN-model. The prompt should also contain a BPMN-text pair as an example (if you want to use annotations for the evaluation, you can include them in the generation). **(2 Points)**
   - Input: BPMN-file
   - Output: Descriptive Text

b. Create a Pipeline (Extractor), that evaluates a given text using a method from exercise sheet 4 by extracting all tasks (, events and gateways) from the text. **(2 Points)**
   - Input: Descriptive Text
   - Output: List of element names

c. Write a function that receives the list of element names from the Extractor-Pipeline and the element names extracted from the BPMN-file used in exercise a) and returns mistakes. **(2 Points)**

d. Create a Pipeline (Corrector), that is given the mistakes that were found in the text and instructs an LLM to correct them (from exercise sheet 5). **(2 Points)**
   - Input: Mistakes
   - Output: Corrected Text

## Exercise 2 (Connecting Pipelines) — 5 Points

Write Python Code to connect the Pipelines by using the Output of a Pipeline as the Input of the next Pipeline.

a. The output of the Generator-Pipeline should be given to the Extractor-Pipeline. **(1 Point)**

b. The output of the Extractor-Pipeline should be evaluated to identify mistakes. If no mistakes are found, this process stops and returns the descriptive text. If there are mistakes in the text, the mistakes are given to the Corrector-Pipeline. **(2 Points)**

c. The output of the Corrector-Pipeline is given back to the Extractor-Pipeline to create a loop. The results are reevaluated until either 5 iterations of the Extractor we executed or the text does not contain any mistakes. **(2 Points)**

**Exercise 3 (Testing)** **7 Points**

Run the code for 3 different BPMN models (airport models from exercise sheet 3) and document the different results.
Documentation should include:

- Initially Generated Text
- Mistakes Found
- Generated Text after each Corrector Pipeline run

Present your results.