Software Engineering

Refactoring

3 Ba INF
2023 – 2024

Kasper Engelen
kasper.engelen@uantwerpen.be

1 Practical

- Deadline: **Sunday, December 10, 2023, 22u00**
- Project files can be found on Blackboard (`e_commerce_refactoring.zip`)

2 Context

Refactoring is a crucial step in order to keep the code base clean, easily maintainable and adaptable. In this project we will take a look at 4 possible refactoring techniques: *Extract Method*, *Move Behaviour Close to the Data*, *Eliminate Navigation Code* and *Transform Self Type Check*.

3 Assignment

1. Read the following articles about the 4 different techniques we will consider (beware: most techniques have different names by different authors):

   - **Extract Method**
     `https://refactoring.guru/extract-method`
   - **Move Behaviour Close to the Data**
     `https://refactoring.guru/move-method`
   - **Encapsulate Field**
     `https://refactoring.guru/encapsulate-field`
   - **Transform Self Type Check**
     `https://refactoring.guru/replace-conditional-with-polymorphism`

2. Explain the 4 different techniques using the following questions:

   - What do they do, what is the goal of applying the technique?
   - When do we use them, what are the symptoms in the code?
   - What are the benefits of applying this technique?
   - How do you apply the technique?

   At the evaluation you will be asked to explain these four techniques.

3. An updated version of the source code is available on Blackboard (`e_commerce_refactoring.zip`). Analyze the new `Payment` package. Use every refactoring technique you just learned at least once to improve the quality of the code. Make a detailed list of the refactorings you apply and explain why they improve the quality of the code.

4. Search the rest of the code base for code that you can refactor, maintain a detailed list of all applied techniques and add it to your report. Find at least one extra refactoring.