# Learning Ressources

Es gilt, sich durch zwei Themenbereiche durchzuarbeiten: UML und Objektorientiertes Programmieren. Beide Themenfelder dienen zur Vermittlung der Fähigkeit, eine Software-Architektur mit zeitgenössischen Mitteln planen und realisieren zu können.

1. UML
In unserem Kurskontext benötigen wir folgende Diagramme: USE-Case Diagram, Sequence Diagram und Class-Diagram.
Posting für USE-Case: <https://www.lucidchart.com/pages/uml-use-case-diagram?usecase=uml>
Posting für Sequence: <https://www.lucidchart.com/pages/uml-system-sequence-diagram?usecase=uml>
Posting für Class-Diagram: <https://www.lucidchart.com/pages/how-to-draw-a-class-diagram-in-UML>
Als Werkzeug nutzen wir draw.io.
2. Objektorientierte Programmierung (OOP) in Python:
Corey Schafer hat dazu vor einigen Jahren eine immer noch sehr gute Video-Serie gemacht: <https://www.youtube.com/playlist?list=PL-osiE80TeTsqhIuOqKhwlXsIBIdSeYtc>
Bitte alle sechs Videos durcharbeiten

[The Game that learns](https://www.youtube.com/watch?v=sw7UAZNgGg8) by Vsauce2

[Template for Shreksapawn](https://twitter.com/VsauceTwo/status/1107733737364770817) by VsauceTwo

# Technical Ressources

Visual Studio Code, Python, VENV or Docker

[Diagram.io](https://app.diagrams.net/)

# Your Project

Using the content covered in both courses and develop the Game “Shreksapawn”.

Expected outcome:

* UML diagram
* Verified and executable code

Put all the resulting files together in a project folder. The project folder should have a structure suitable for collaboration in a small data science team of 4-5 people. Submit your project folder in Moodle by 18.07.2023