

**KNΔPP**

**OOP**

**Textadventure Game**

## Inhaltsverzeichnis

<b>Inhaltsverzeichnis</b>	<b>1</b>
<b>1 Aufgabenstellung</b>	<b>2</b>
1.1 Ziel des Spiels	2
1.2 Vorgaben und technische Anforderungen	2
1.3 Zeitrahmen und Abgabe	2
<b>2 Lösung</b>	<b>3</b>
2.1 UML	3
2.2 Beschreibung des OOP-Konzepts	3
2.3 Relevante Codeausschnitte (mit Beschreibung)	3

## 1 Aufgabenstellung

Entwickeln Sie im Rahmen dieses Projekts ein textbasiertes Adventure-Game, das vollständig objekt-orientiert (OOP) umgesetzt ist. In diesem Spiel übernimmt die Spieler\*in die Rolle einer abenteuerlustigen Person, die in einer geheimnisvollen, verzweigten Welt erwacht und nur mit Geschick und Verstand entkommen kann.

### 1.1 Ziel des Spiels

Das Hauptziel besteht darin, einen verborgenen Schatz zu finden, der in einem schwer zugänglichen Teil der Spielwelt versteckt ist. Um dieses Ziel zu erreichen, muss die Spieler\*in verschiedene Räume erkunden, Gegenstände sammeln und logisch einsetzen sowie mit unterschiedlich gesinnten Charakteren interagieren. Auf dem Weg zum Schatz gilt es, Rätsel zu lösen und Herausforderungen zu bestehen, wie das Überwinden von Hindernissen, das Entsperren geheimer Türen oder das Austricksen von Wächtern. Nur wer die Welt aufmerksam erforscht und die Möglichkeiten der gesammelten Objekte geschickt nutzt, kann das Ziel erreichen.

### 1.2 Vorgaben und technische Anforderungen

- Nutzen Sie konsequent die Prinzipien der objektorientierten Programmierung: Klassen, Objekte, Vererbung, Polymorphie und Kapselung.
- Mindestens folgende Klassenhierarchien müssen enthalten sein:
  - Raum (Basisklasse, abgeleitete Spezialräume z. B. „Schatzkammer“, „Gefängnis“, „Bibliothek“)
  - Gegenstand (z. B. „Schlüssel“, „Waffe“, „Heiltrank“)
  - Charakter (mindestens Spieler\*in und verschiedene NPCs mit eigenen Eigenschaften und Verhalten)

Jeder Raum soll individuelle Eigenschaften und Methoden besitzen (Beschreibung, Aktionen, angrenzende Räume).

Die Interaktionen (z. B. „aufheben“, „benutzen“, „reden“) werden durch Methoden der jeweiligen Klassen realisiert.

Ein Inventarsystem für die Spieler\*in ist zu implementieren.

Gesteuert wird das Spiel über eine Textschnittstelle (z. B. „gehe Norden“, „nimm Schlüssel“, „rede mit Wächter“).

Fokus auf saubere Strukturierung, Wiederverwendbarkeit und Erweiterbarkeit durch OOP.

Dokumentation und Kommentierung des Codes sowie eine Beschreibung des OOP-Konzepts und der Vererbung (1–2 Seiten).

### 1.3 Zeitrahmen und Abgabe

Der Arbeitsaufwand beträgt ca. 40 Stunden (eine Arbeitswoche). Reichen Sie den funktionierenden Quellcode, eine kurze Dokumentation und eine Anleitung zum Starten des Spiels ein.

## **2 Lösung**

**2.1 UML**

**2.2 Beschreibung des OOP-Konzepts**

**2.3 Relevante Codeausschnitte (mit Beschreibung)**