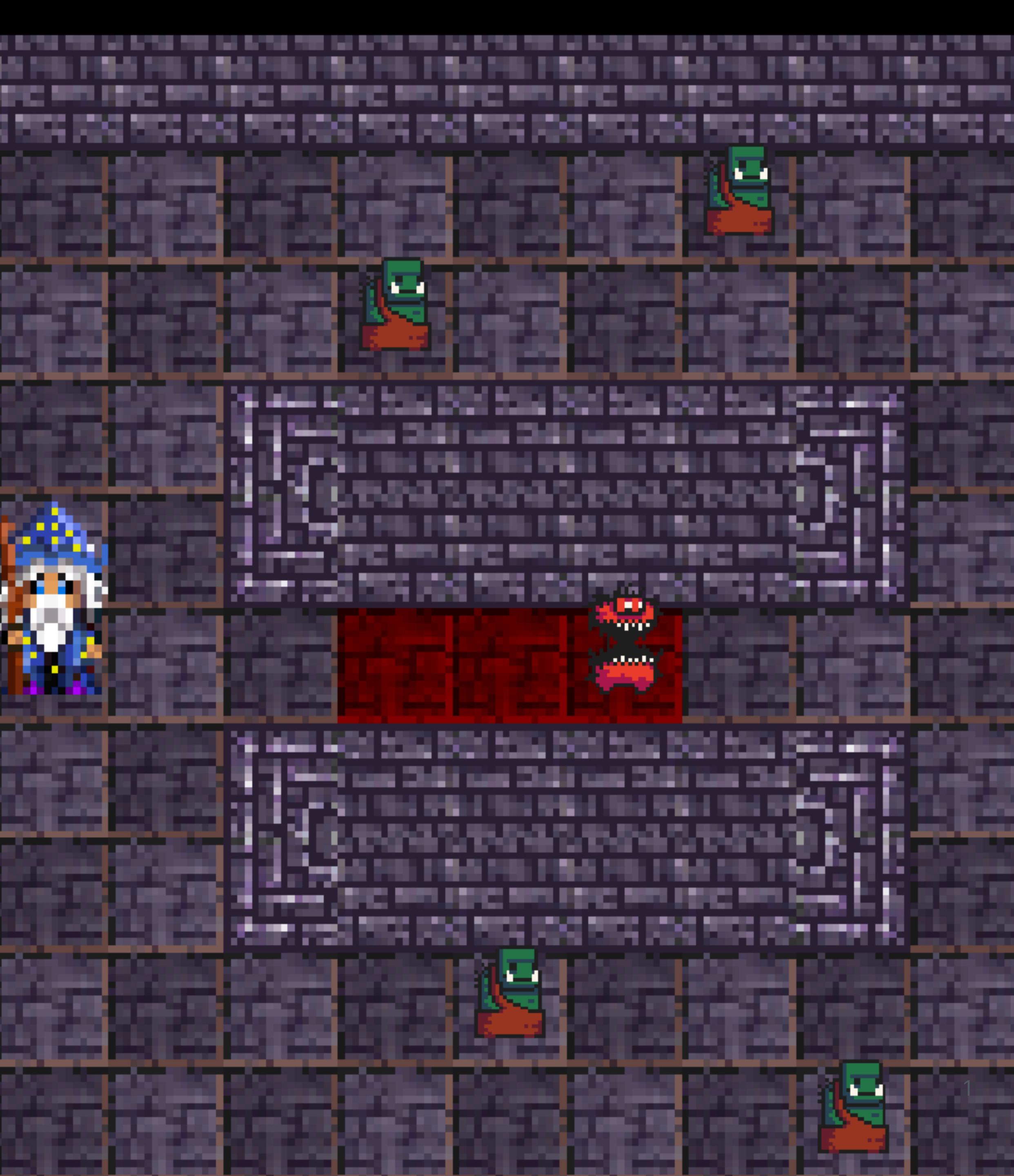


# Dungeonverse

Ein Universum aus Spielen,  
Code und Gamification

André Matutat



# Agenda

Ausgangssituation

Gamification

Vorstellung der Projekte



# Ausgangssituation: klassische Hochschullehre

- **Ausgangssituation**

- **Klassische Vorlesung**

- **Wöchentliche Praktikumsaufgaben mit Vorstellung und Bewertung**

- **Klausur am Ende der Vorlesungszeit**

- **Probleme**

- **Große Heterogenität der Studierenden**

- **Motivation bei wöchentlichen kleinen Aufgaben niedrig**

- **Kontaktzeit vor allem für Bewertung von Aufgaben**

KW	Vorlesung (Sprechstunde)	Praktikum (Rücksprache)
15	<a href="#">Orga (Zoom)</a>    <a href="#">Git 1: Intro</a>   <a href="#">Git 2: Basics</a>	- (extra Zoom für alle 10-11 Uhr)
16	<a href="#">Git 3: Branches</a>   <a href="#">Git 4: Branching-Strategien</a>    <a href="#">Logging</a>    <a href="#">Javadoc</a>	<a href="#">B1: PM-Dungeon und Held</a>
17	<a href="#">Git 5: Remote</a>    <a href="#">Visitor-Pattern</a>    <a href="#">Generics 1: Klassen &amp; Methoden</a>   <a href="#">Generics 2: Bounds &amp; Wildcards</a>	<a href="#">B2: Logging, Monster und Kampf</a>
18	<a href="#">Generics 3: Type Erasure</a>   <a href="#">Generics 4: Polymorphie</a>    <a href="#">Observer-Pattern</a>    <a href="#">Defaultmethoden</a>	<a href="#">B3: Loot, Taschen und Kisten</a>
19	<a href="#">E1</a>    <a href="#">JUnit 1: Testen mit JUnit</a>   <a href="#">JUnit 2: Testfallerstellung</a>	<a href="#">B4: HUD, Fallen, Erfahrung und Skills</a>
20	<a href="#">Bad Smells</a>   <a href="#">Refactoring</a>    <a href="#">Git 7: Bisect</a>    <a href="#">Strategy-Pattern</a>	<a href="#">B5: Quests, JUnit</a>
21	-	-
22	<a href="#">Funktions-Interfaces &amp; Lambdas</a>   <a href="#">Methodenreferenzen</a>    <a href="#">RegExp</a>    <a href="#">Git 6: Workflows</a>	<a href="#">B6: Fernkampf, schlaue Monster und Refactoring</a>
23	<a href="#">Annotationen</a>    <a href="#">Reflection</a>	<a href="#">Pitch Projekt</a>
24	<a href="#">E2</a>    <a href="#">CLI/Konfiguration</a>    <a href="#">Swing1</a>   <a href="#">Swing2</a>   <a href="#">Swing3</a>   <a href="#">Swing4</a>	<a href="#">Meilenstein 1</a>
25	<a href="#">Build 1: Ant</a>   <a href="#">Build 2: Gradle</a>   <a href="#">Build 3: CI</a>   <a href="#">Docker</a>	<a href="#">Meilenstein 2</a>
26	<a href="#">Threads 1: Intro</a>   <a href="#">Threads 2: Synchronisation</a>   <a href="#">Threads 3: High Level Concurrency</a>    <a href="#">Java2D</a>	<a href="#">Präsentation Projekt</a>
27	<a href="#">Build 4: Maven</a>    <a href="#">Enumerationen</a>    <a href="#">Rückblick</a>    <a href="#">Prüfungsvorbereitung</a>	-
28	-	-
29	<b>E3, E-Klausur Performanz</b>	-



# Gamification

Mit Spaß zum Lernerfolg

 Was ist Gamification?

Anwenden von spieltypischen Elementen (z.B Punkte, Level, Quests) In nicht spielerischen Kontexten

 Funktioniert das?

- Erhöht intrinsische Motivation
- Macht Lernfortschritt sichtbar und greifbar
- Liefert kontinuierliches Feedback

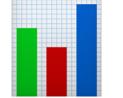


# Serious Games

Wenn das Spiel zum Klassenraum wird

 Was sind Serious Games?

Digitale Spiele, die **nicht primär** der **Unterhaltung** dienen, sondern **ernsthaften Zwecken**, dienen.

 Funktioniert das?

- Spielmechanik & Lernziel sind verknüpft
- Lernen ist integraler Bestandteil des Spiels
- Fortschritt im Spiel setzt Wissen oder Kompetenzen voraus



# Das Dungeon



# Dungeon-Projekte

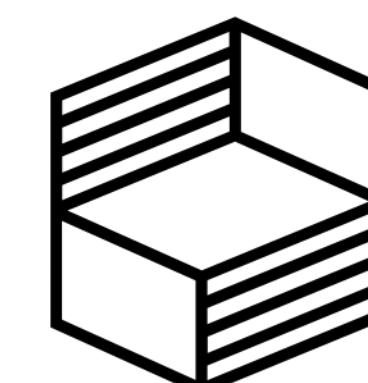
- Fellowships für Innovationen in der digitalen Hochschulbildung (2020/21)
- Freiraum 2022 (2022/23)
- Pakt für Informatik 2.0 (2025/26)
- Freiraum 2025 (2025/26)



Ministerium für  
Kultur und Wissenschaft  
des Landes Nordrhein-Westfalen



**EFRE/JTF**  
NRW 2021-27



**Stiftung  
Innovation in der  
Hochschullehre**



**Kofinanziert von der  
Europäischen Union**

Ministerium für Wirtschaft,  
Industrie, Klimaschutz und Energie  
des Landes Nordrhein-Westfalen

# Dungeon Team



Prof. Dr. Ing.  
Carsten Gips



André Matutat (M.Sc.)

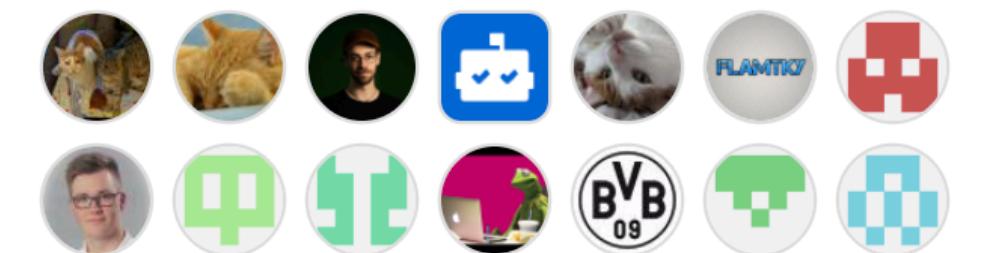


Dipl.-Inform.  
Birgit Christina George



Prof. Dr. math., Dipl.-Math.  
Jörg-Michael Keuntje

Contributors 27

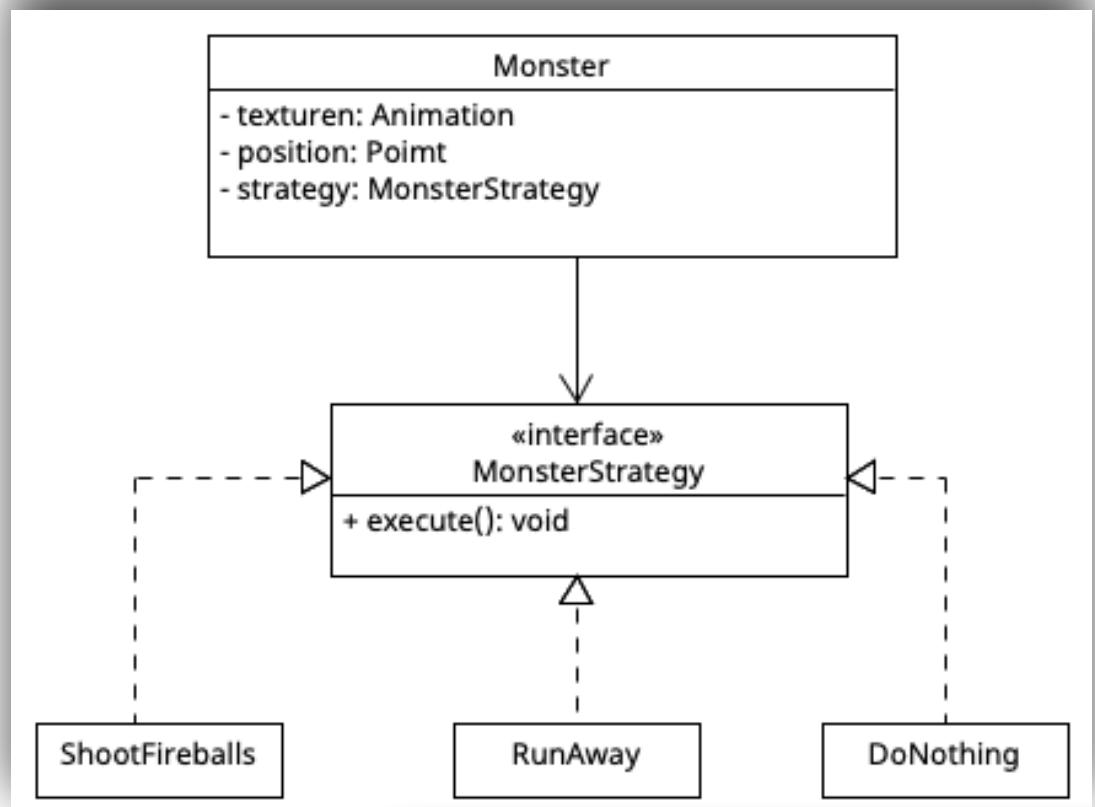


+ 13 contributors

# PM-Dungeon: Spiele-Framework für Die Lehre

## Aufgabe

Implementiere verschiedene Monster, mit verschiedenen Verhaltensweisen.



```
new
public class Monster {
    1 usage
    private Animation texture;
    1 usage
    private Point position;
    1 usage
    private MonsterStrategy strategy;

    no usages new *
    public Monster(Animation texture, Point position, MonsterStrategy strategy) {
        this.texture = texture;
        this.position = position;
        this.strategy = strategy;
    }
}
```



# Dungeon: Learning By Questing

## Aufgabe

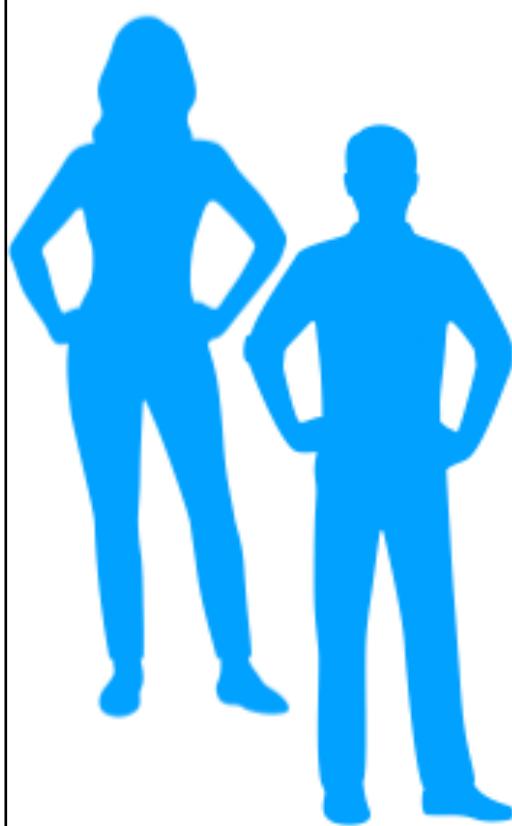


```
graph TD
    G--T--Q--D
    Q--F
    T--X--S
    G--W--E
    W--C--U
    C--N
}

task my_task {
    text: "Öffne die Schatztruhe, Code: Post-Order",
    task_type: TEXT,
    correct_answer: postorder(g)
}

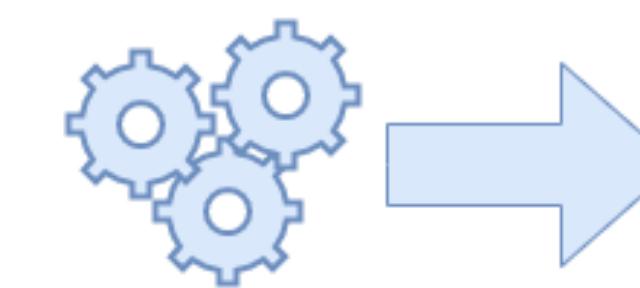
fn postorder(graph g) -> string[] {
    retval: string;
    if g.left
        retval += postorder(g.left)
    if g.right
        retval += postorder(g.right)
    retval += g.node_name
    return retval;
}

level_config config {
    task: my_task,
    // task-builder Funktion registrieren, sodass sie
    task_builder_functions: [build_chest_quiz]
}
```



Lehrende

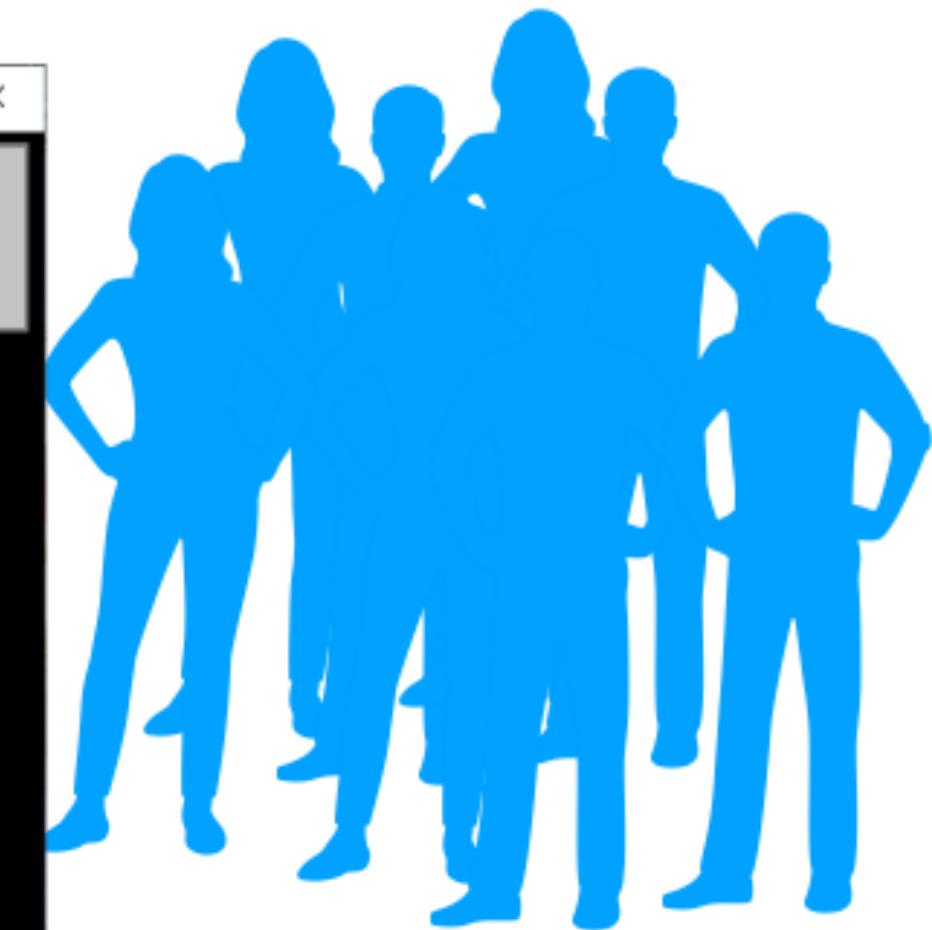
DSL Eingabe



DSL Interpreter



Generiertes Dungeon-Level  
und Quest



Studierende

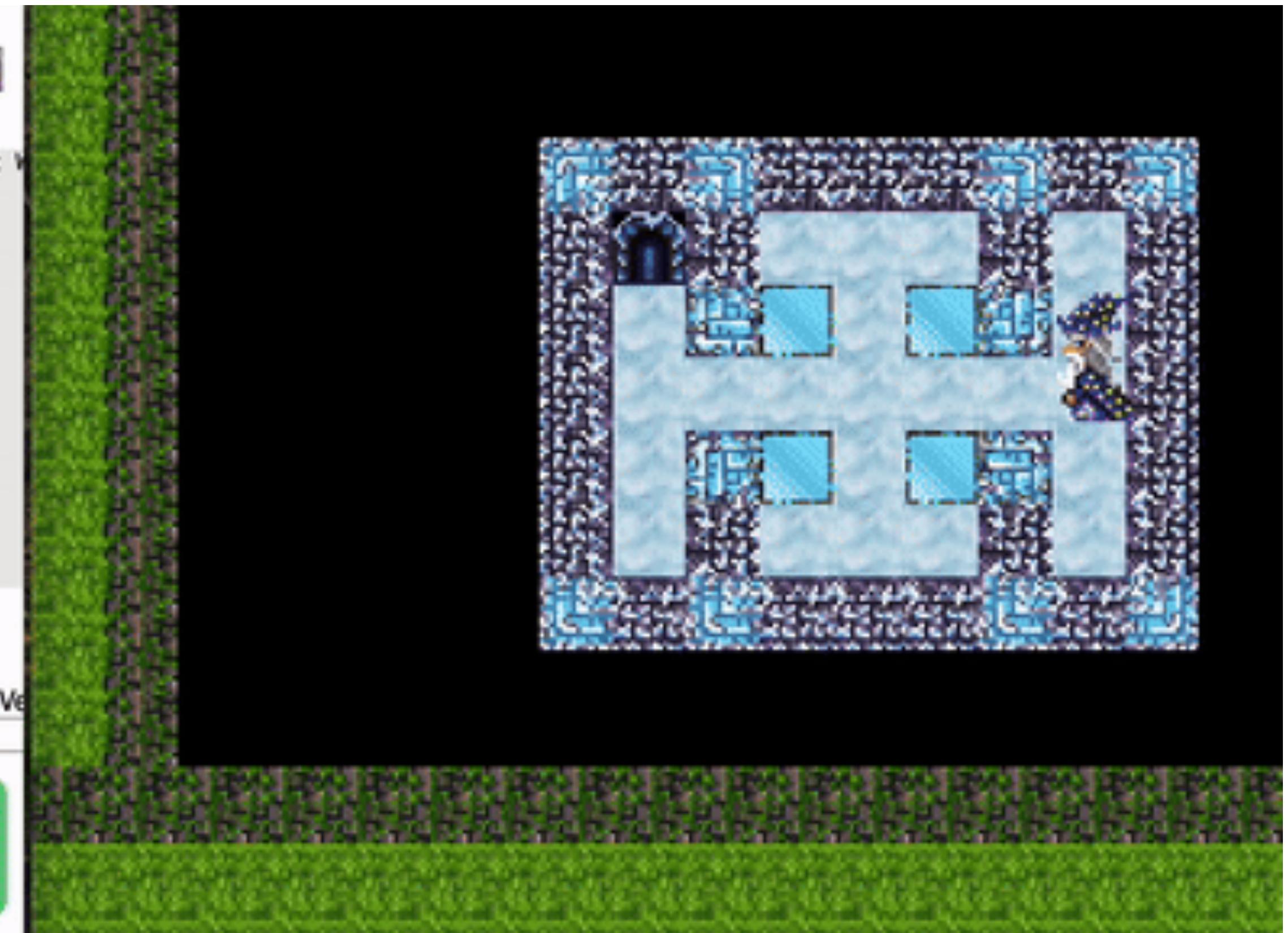
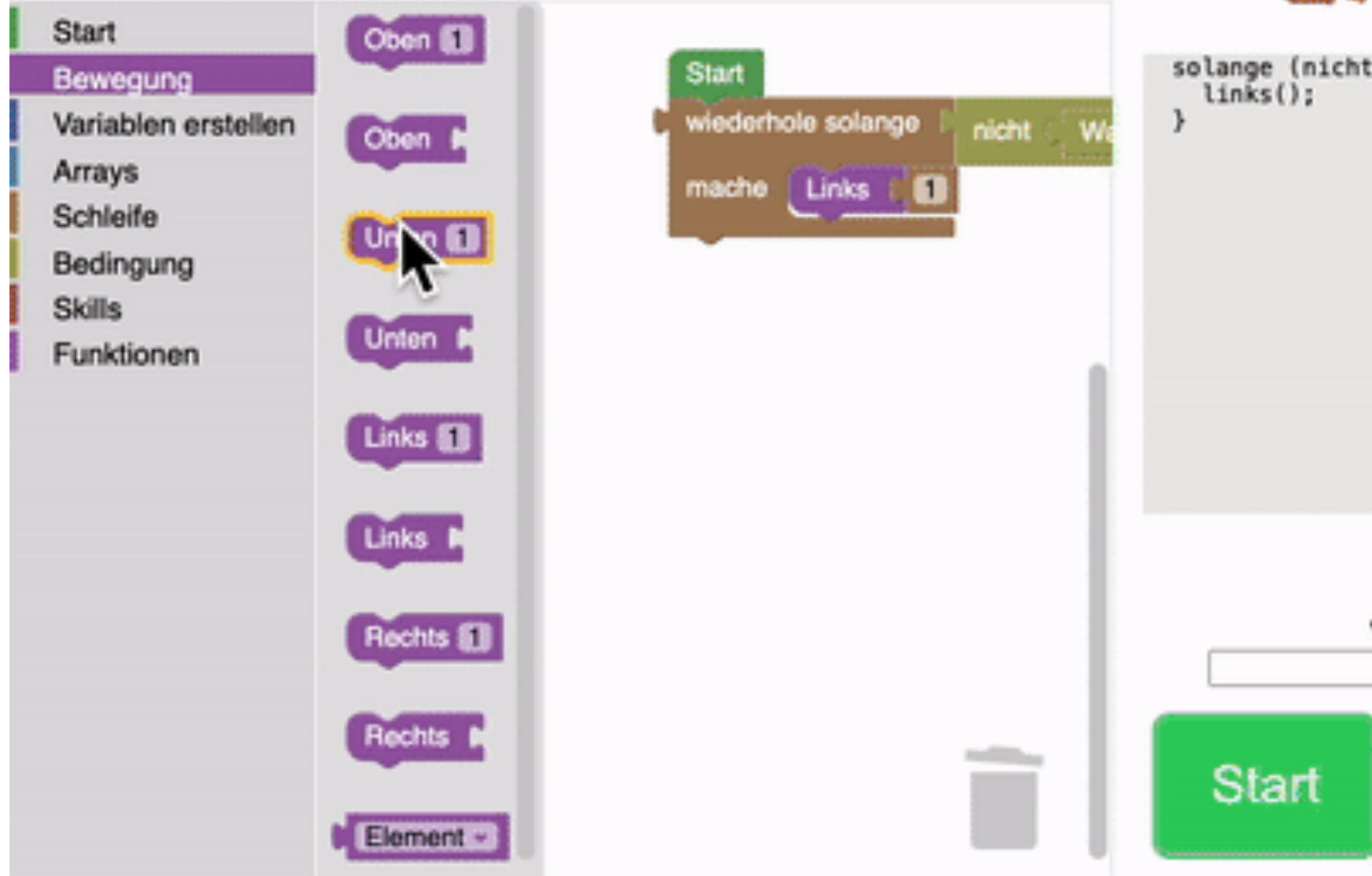


# BEISPIEL-AUFGABE: KOCHEN MIT FOTOSYNTHESE



# PRODUS

## Blockly Dungeon



# Produs + Dev-Dungeon



# Produs + Dev-Dungeon



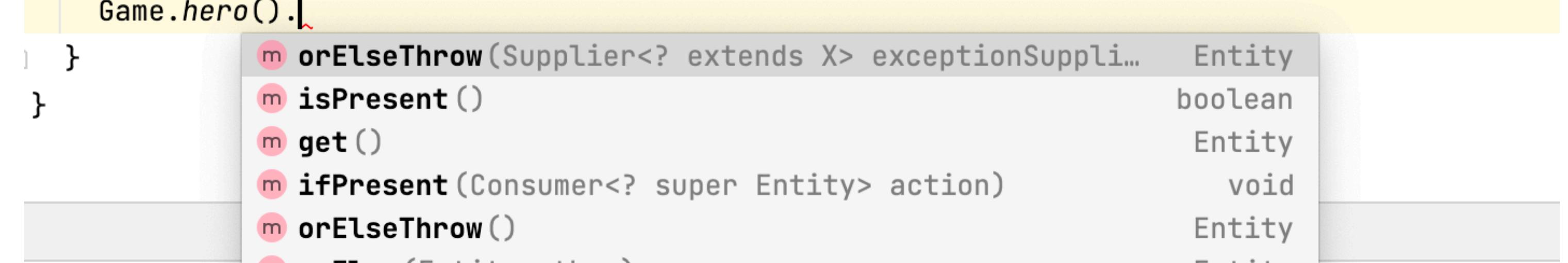
# Produs + Dev-Dungeon



# Prod

```
1 usage  Niklas Schumann
public SpeedEffect(float speedIncrease, int duration) {
    this.speedIncrease = speedIncrease;
    this.duration = duration;
}

 /**
 * Applies a temporary speed increase to the target entity, then reverts its speed to normal
 * after the specified duration. The increase in speed is applied immediately, and its reversal
 * is scheduled to occur after the duration expires.
 *
 * <p>TODO: Implement the applySpeedEffect method to schedule the speed increase and its
 * reversion.
 *
 * @param target The entity to which the speed effect will be applied.
 */
2 usages  Niklas Schumann *
public void applySpeedEffect(Entity target) {
    Game.hero().|_
}
```



m <b>orElseThrow</b> (Supplier<? extends X> exceptionSupplier)	Entity
m <b>isPresent</b> ()	boolean
m <b>get</b> ()	Entity
m <b>ifPresent</b> (Consumer<? super Entity> action)	void
m <b>orElseThrow</b> ()	Entity

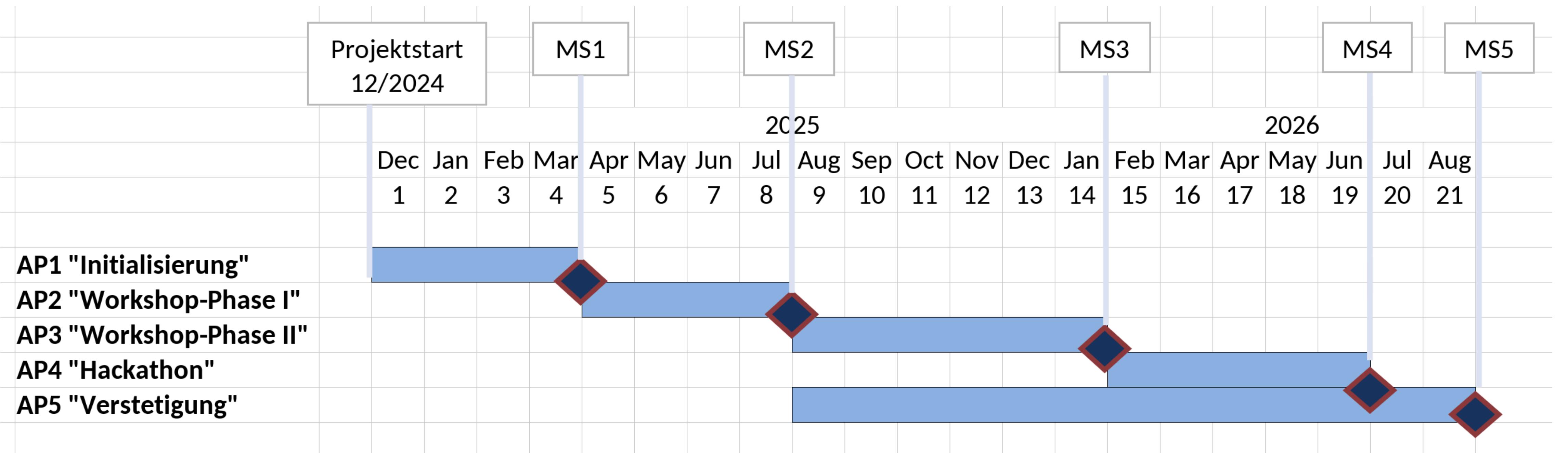


# Produs + Dev-Dungeon



# PRODUS: Zeitplan

ca. 5 Workshops je Schulhalbjahr (a 2x 3..5 Stunden), in BI und MI



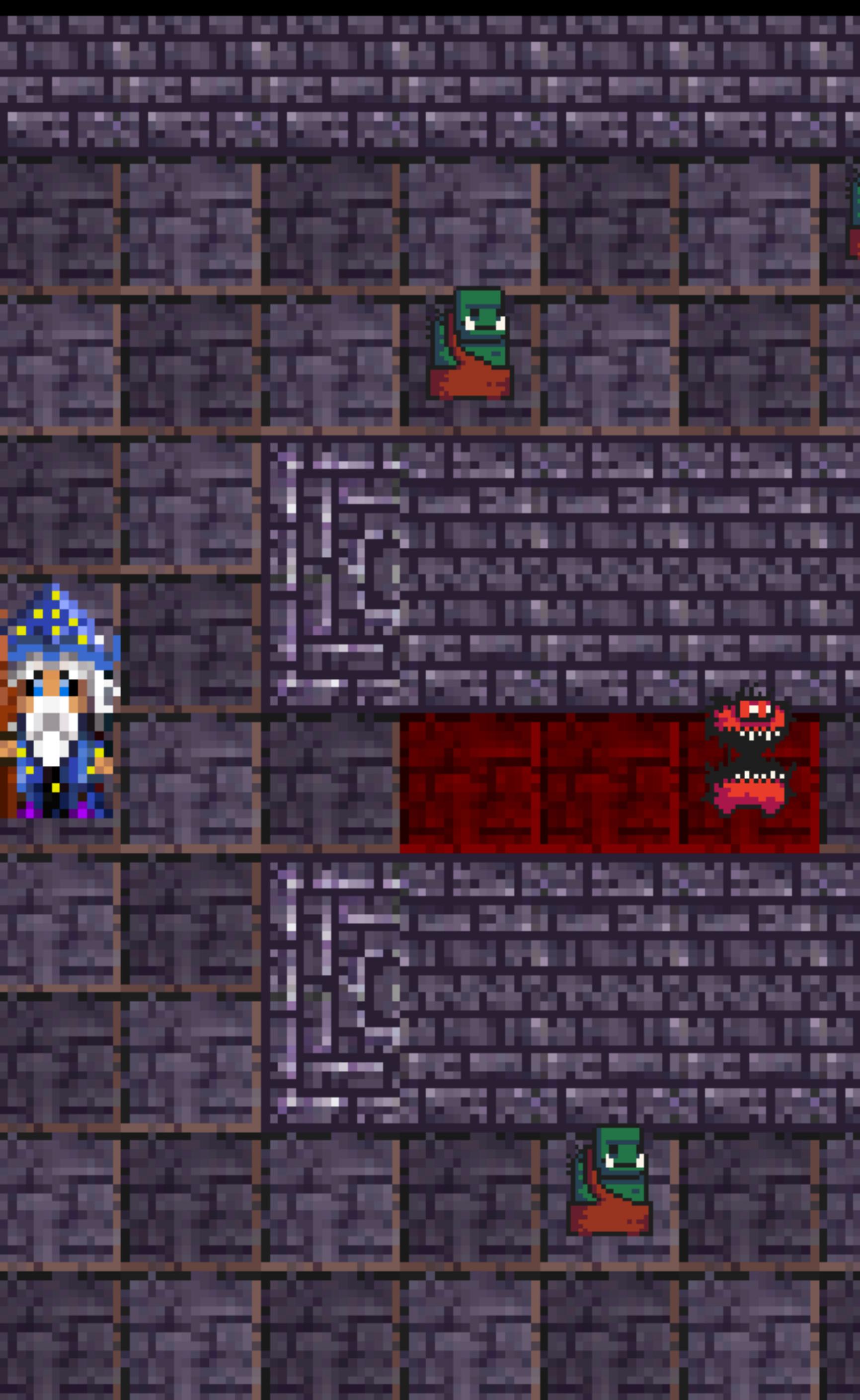
# L<ESC>rod

- Lehr-Escaperooms im und mit dem Dungeon
- Multiplayer: Rätsel gemeinsam lösen
- Framework zur einfachen Umsetzung von Lehr-Escaperoom

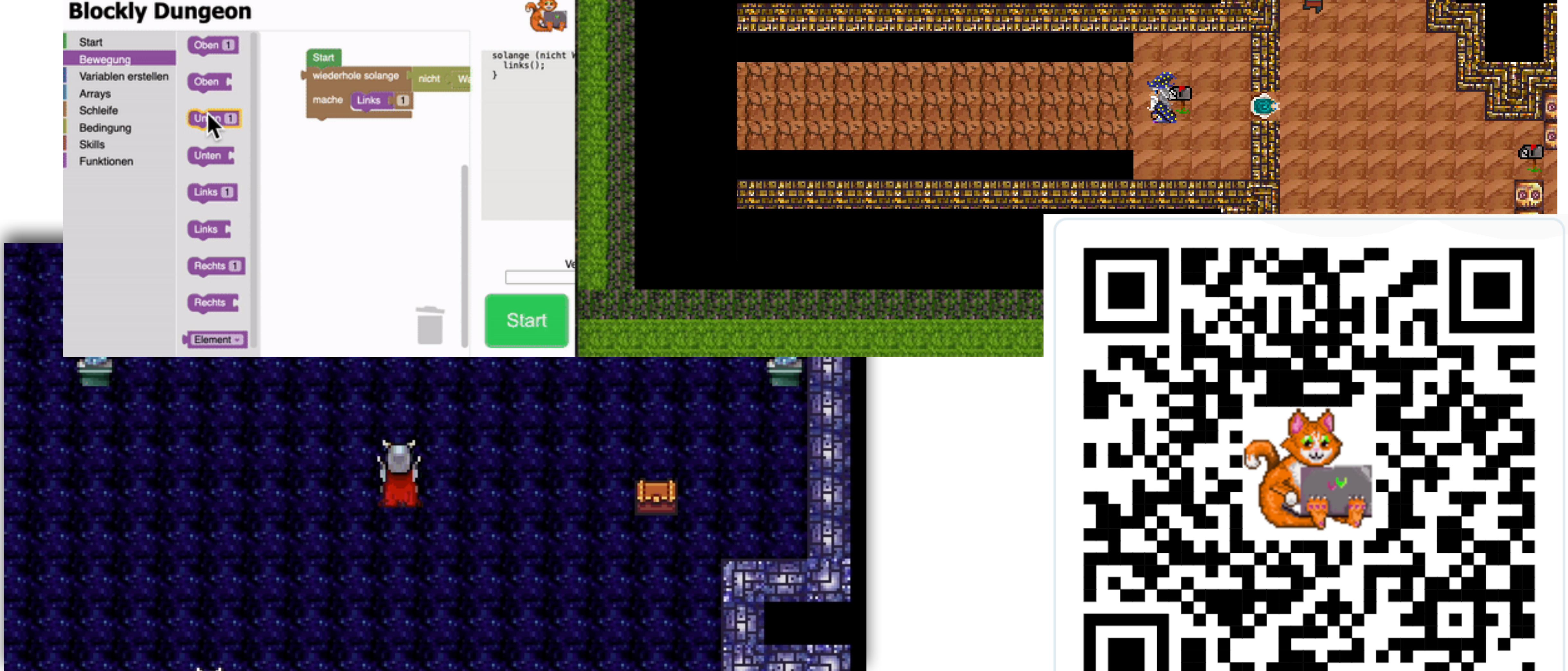


# Was ist das Dungeon?

- **Baukasten**
  - Lernprojekt für Programmierer
  - Lernspiel für alle
  - Werkzeug für Workshops
  - ...
- **Open Source und Open Education Resources**



# Fragen?





This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).