



Unterstützung von Programmier-Anfängern beim Aufbau Mentaler Modelle für Objekt-Orientierte Programmierung



Problemstellung

„Ich habe mir die Aufgabe angesehen, habe aber keine Ahnung, wie und wo ich anfangen soll“ oder „Ich weiß, wie ich eine for-Schleife schreibe, ich weiß nur nicht, wann und warum ich die hinschreiben soll.“

Es fällt vielen Programmieranfänger*innen schwer, korrekte mentale Modelle der Lerninhalte aufzubauen.



Ziele

**Didaktische Ansätze optimieren
IDE zur Unterstützung weiterentwickeln**

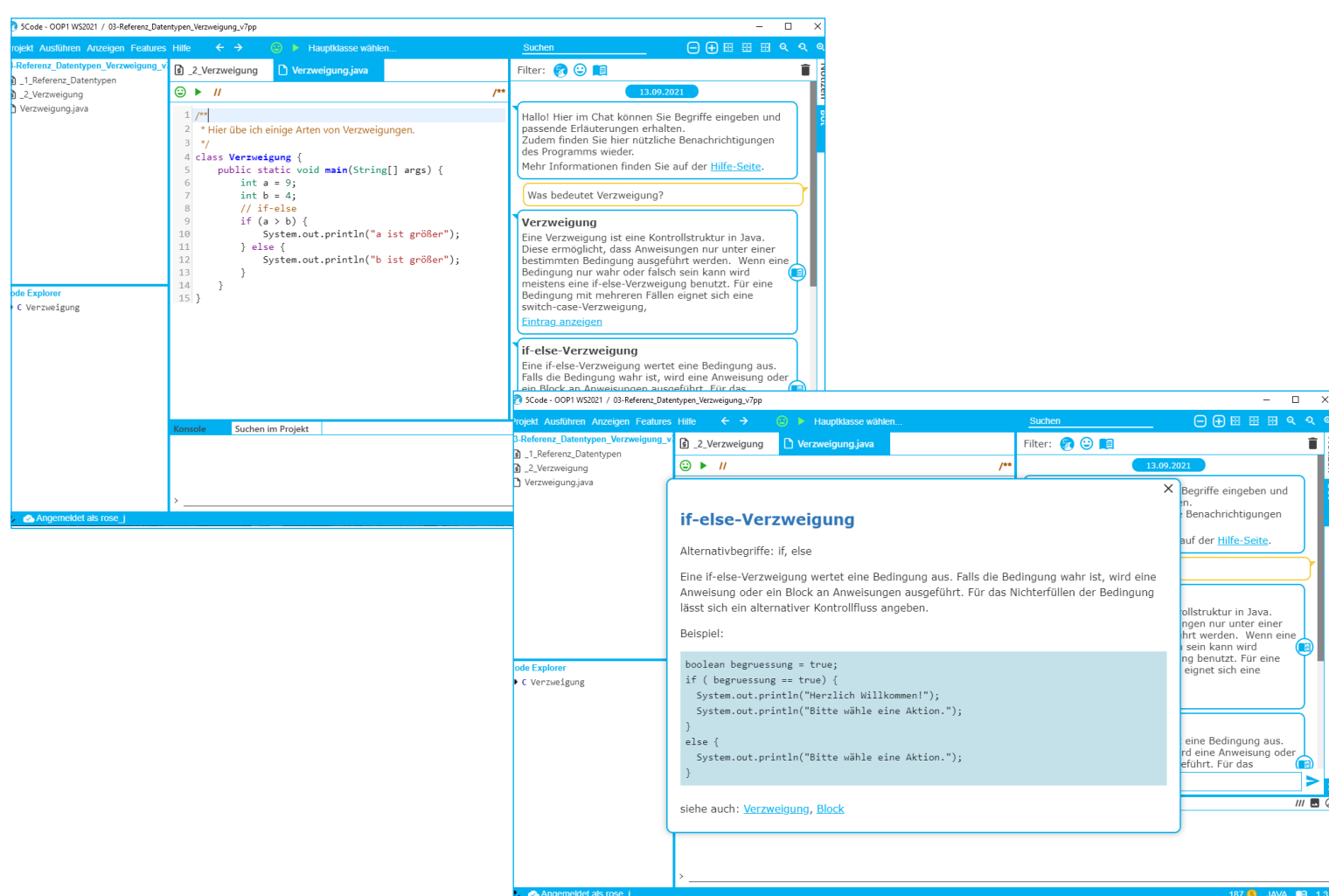


Mentale Modelle

beinhalten sowohl Bedeutungen der verschiedenen Programmierkonzepte als auch ihre Zusammenhänge, sowie Schritte und Vorgehensweisen beim Finden einer Lösung.
Die Konstruktion mentaler Modelle ist abhängig von dem Vorwissen und den bestehenden Vorstellungen der Lernenden.

Chat Bot

- Einfacher Chat-Bot zum Durchsuchen von Glossar-Einträgen
- Wichtige System-Meldungen
- Hinweise zu auftretenden Compiler- und Laufzeitfehlern
- Filtern von Einträgen
- Vorschau von und Verlinkungen zu längeren Info-Seiten



Visual Debugger

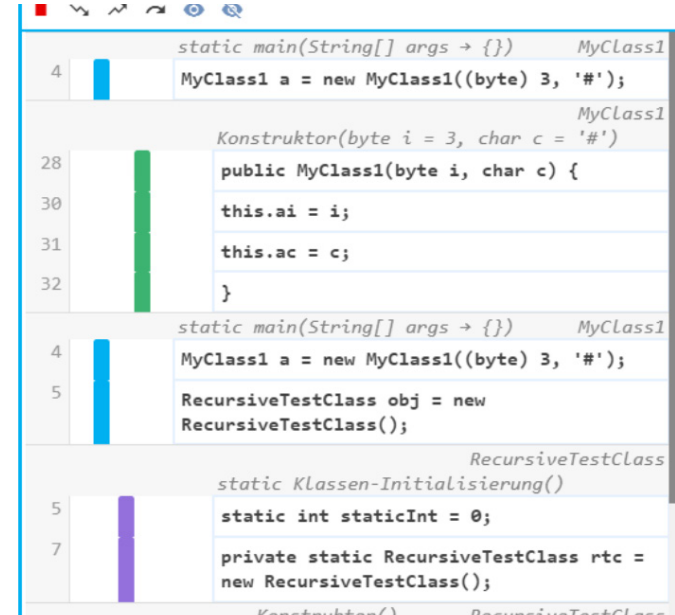
- Macht Debugging leichter zugänglich als professionelle Debugger
- Hilft dabei, Zusammenhänge zu verstehen

Ablauf-Verfolgung

- Grafische Darstellung des Ablaufs des eigenen Codes
- Visualisierung von Kontrollstrukturen wie Verzweigungen, Schleife
- Visualisierung des Aufrufs von Methoden
- Verstehen des echten Ablaufs von Code
- Rolle und Übergabe von Parametern

Objektdiagramm

- Verstehen von Objekten und ihren Instanz-Variablen
- Verstehen von Referenzen
- Verstehen von Arrays



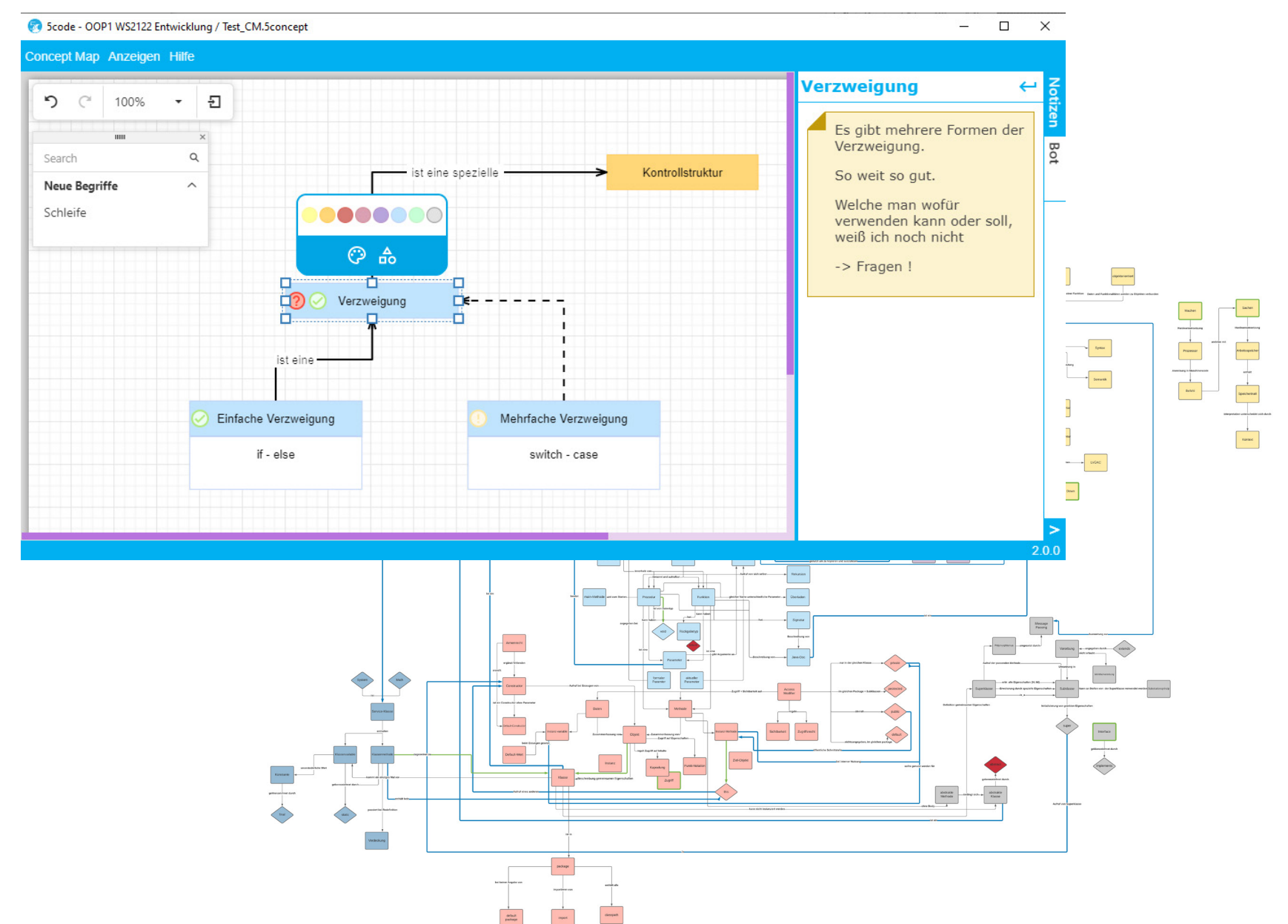
Peer Review / Grading

- Gegenseitige Begutachtung von Ergebnissen von Peers (Studierenden) untereinander
- Lernen und Aufbau des Mentalen Modells durch Verbalisierung
- Stellen von Fragen an Peers oder Betreuer*innen wird erleichtert
- Ausbau zur Prüfungsform

Concept Map

- Erstellung einer Concept Map schrittweise über das Semester hinweg
- Begriffe werden vorgegeben
- Ihre Beziehungen werden selber ergänzt und beschrieben
- Ergänzung durch Badges und eigene Notizen

- Abbildung der mentalen Modelle
- Dokumentation des Lernstands
- Fragen und Unklarheiten aufdecken



Reflect, Metriken

Reflect

- Reflektiert den selbst geschriebenen Code
- Generiert automatisch Ablaufdiagramme für gewünschte Code Passagen
- Bietet natürlichspachige Erklärungen für die verschiedenen Einzelteile einer Code Zeile an
- Kann so mögliche formale Probleme von Code aufzeigen
- Hilft den Studierenden die Funktionsweise des eigenen Codes besser zu verstehen

Metriken

- Automatische Analyse von Code nach Qualitätsmetriken
- Bietet schnelles und automatisches Feedback
- Leitet den Selbstlernprozess von Studierenden an
- Unterstützt das Entwickeln von besser les- und wartbaren Code
- Integrierter Auto-Formatierer lädt zum Erlernen von Code Styles ein

Kollaboration

- Direkten Austausch von Studierenden unterstützen
- Gemeinsames Arbeiten an Projekten
- Austauschen von Textnachrichten und Code-Snippets
- Live-Verfolgung von Änderungen im Code
- Austauschen von Projektdateien

