



Vorlesung

Webbasierte Informationssysteme

(CS4130)

Einleitung

Professor Dr. rer. nat. habil. Sven Groppe

<https://www.ifis.uni-luebeck.de/index.php?id=groppe>

Hörerkreis



Studiengang	Bachelor / Master	Semester	Basismodul Praktische Informatik	Wahlpflicht
Informatik	Master	1 oder 2 (2 oder 3)	✓	(✓ als <i>Basismodul PI</i> im Schwerpunktfach <i>Software Systems Engineering (SSE)</i>)
Medizinische Informatik	Master	1 oder 2 (1 oder 2)	✓	(✓ eHealth / Informatik)
Entrepreneurship in digitalen Technologien	Master	1 oder 2	✓	
Medieninformatik	Master	beliebig		✓
IT-Sicherheit	Master	1 oder 2	✓ (Ab WS 19/20)	
Robotics and Autonomous Systems	Master	1 oder 2		✓ (Ab WS 19/20)

Zeiten und Räume



	Vorlesung	Übung 1	Übung 2	Übung 3
SWS	2	2		
Wochentag	Donnerstag	Montag	Montag	Donnerstag
Zielgruppe	Alle			
Beginn	12:15	14:15	16:15	14:15
Ende	13:45	15:45	17:45	15:45
Raum	AM S2	PC-Pool 3	PC-Pool 3	PC-Pool 2

Organisation

- [Internet-Seite](#)
- Skripte, Übungen, zusätzliches Material im [Uni-Moodle](#)
 - Eintragungen in Übungsgruppen über [Uni-Moodle!](#)
 - Eintragungen ab heute 14:15 Uhr



Ziele dieser Vorlesungseinheit

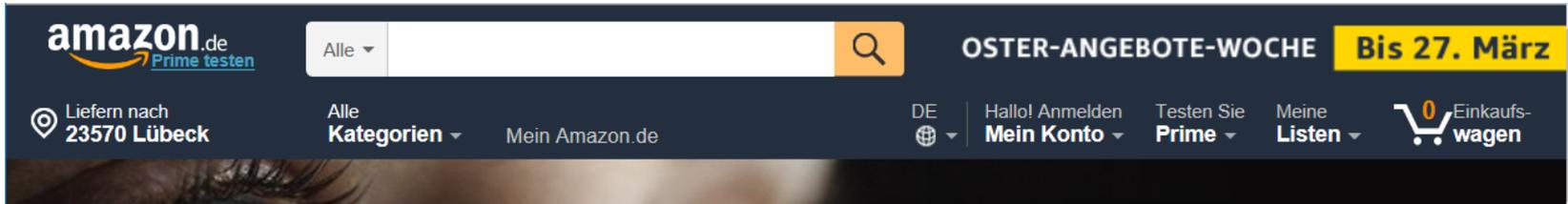


- Einführung in die Themen der Vorlesung
- Bedeutung der Themen für die berufliche Laufbahn
- Vorstellung des Lehr- und Übungskonzeptes dieser Veranstaltung
- Überblick über die Termine und Inhalte der Vorlesungseinheiten

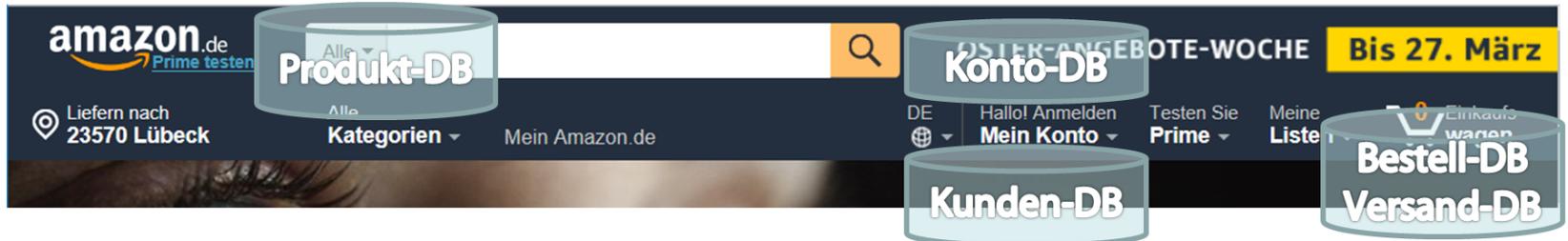


Chronologische Übersicht über die Themen

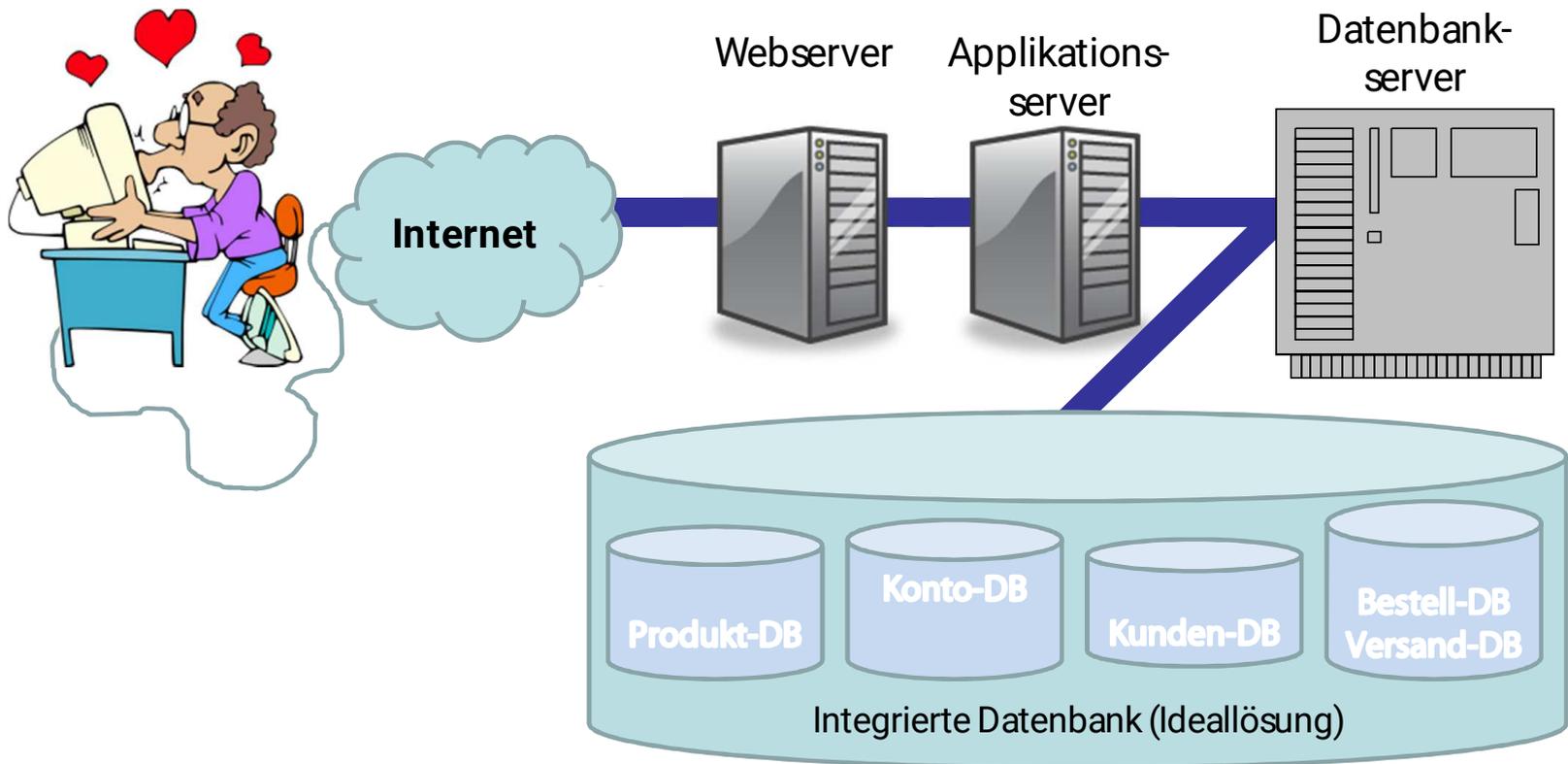
Typische Web-Applikation - Webshop



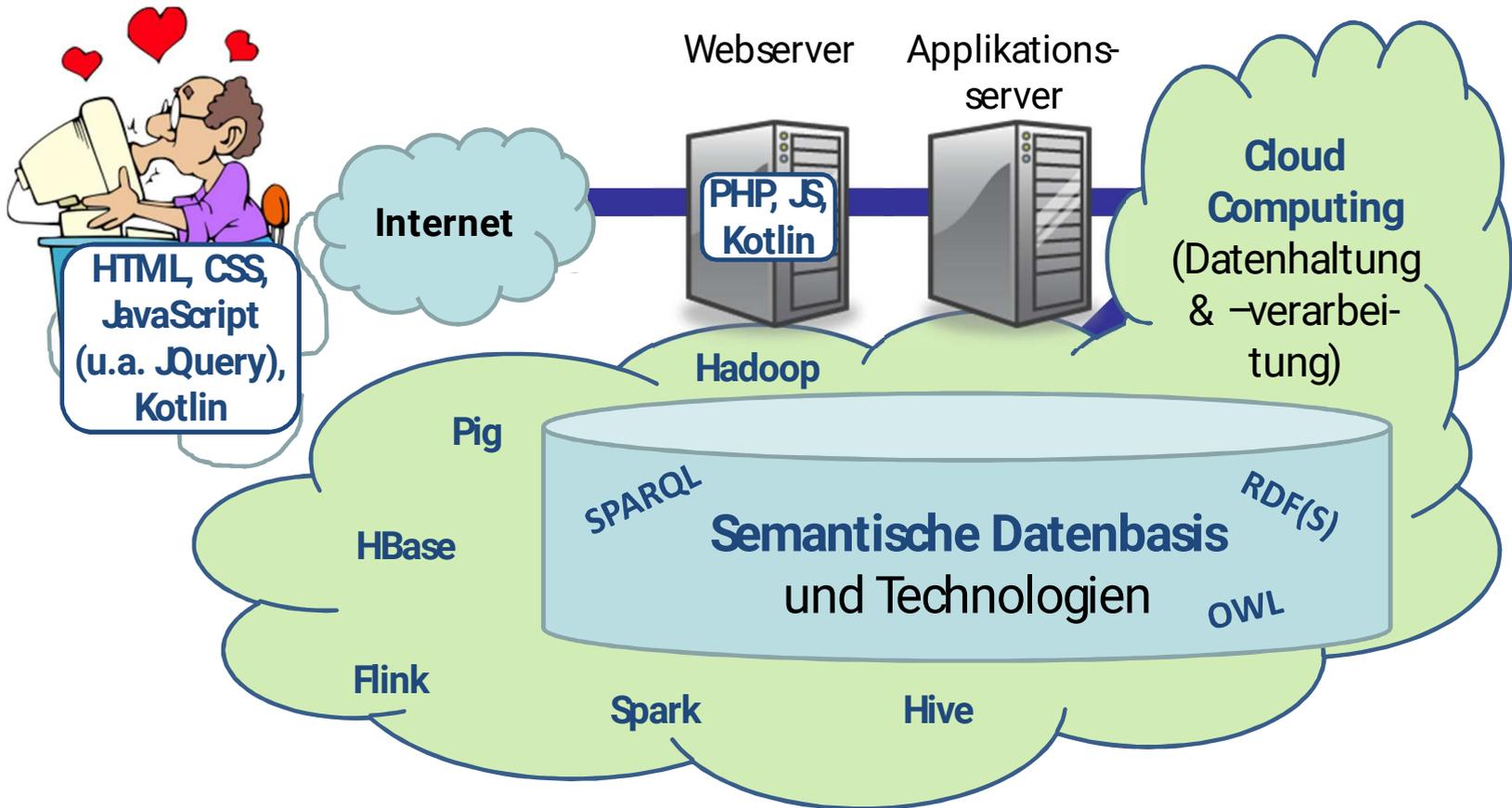
- Datenbanken in Verwendung:



Backend typischer Web-Applikationen (hier 4-Tier-Architektur)



Themen der Vorlesung überall einsetzbar



Collage aus aktuellen Stellenangebote aus dem Großraum Lübeck

(Inhouse) Webentwickler (m/w)

Ihr Profil

- Abgeschlossenes Studium der Informatik, Wirtschaftsinformatik o.ä.
- Sehr gute Kenntnisse in JavaScript/jQuery, HTML und CSS
- Erfahrung im Umgang mit Deployment-Verfahren und Versionierung
- Erste Erfahrung in den gängigen Webtechnologien (REST, SOAP)
- Grundlegendes Verständnis von SEO, SEM, Social Media, Conv Commerce
- Analytisches Denken, schnelle Auffassungsgabe und ein gesundes Selbstbewusstsein
- Gute kommunikative Fähigkeiten, Eigenverantwortlichkeit sowie Kenntnisse in den Bereichen

IT-Consultant Java (m/w) für den Standort Hamburg

Mein Kunde ist ein international tätiges Beratungsunternehmen. In anstehenden, großen Projekten wird schnellstmöglich eine Festanstellung gesucht. Sie arbeiten mit einem Team, das die Sicherheit eines großen Unternehmens garantiert und ein Arbeitsumfeld, das technologisch immer aktuell ist.

Ihre Aufgaben:

- Standort für diese Position ist Hamburg in attraktiver Innenstadtlage.
- Zum **Aufbau dieser Plattform** wird Verstärkung im Team **Application Management** gesucht. Spannend sind vor allem der Technologie Stack bestehend aus **Hadoop Clustern, Echtzeitsystemen in der Cloud (AWS), DWH-Systemen** und einem eigenem Backend zur Batchverarbeitung.

Ihr Profil

- Sie verfügen über ein abgeschlossenes Studium der Informatik / Computer-Linguistik oder eine vergleichbare Qualifizierung sowie Erfahrung in der Software-Entwicklung mit Java
- Ihre Kenntnisse in der semantischen Sprachverarbeitung oder **Semantic Web Technologien (OWL, RDF, Description Logic)** sind für Sie von Vorteil
- Abgeschlossenes Studium der Informatik (oder eines vergleichbaren Studiengangs)
- Sie beherrschen Technologien wie Spring, Hibernate und Maven
- Kenntnisse aus der Web-Entwicklung wie HTML/CSS/JavaScript sind gern gesehen
- Sie zeichnen sich durch Teamfähigkeit, Zielstrebigkeit und eine hohe Begeisterungsfähigkeit aus
- Sie möchten sich weiterentwickeln und Karriere machen

- Erfahrungen mit Microsoft Entwicklungsumgebungen, .NET Framework 4.0, Visual Studio 2010
- Gute Entwicklungserfahrungen in C#
- Kenntnisse in ASP.NET, MVC, JavaScript, CSS, jQuery und Ajax
- Kenntnisse in Datenbanktechnologien (SQL, T-SQL, Stored Procedures, Datenbankdesign), gerne auch NoSQL
- Gute deutsche Sprachkenntnisse
- Praktische Erfahrungen in gängigen Infrastruktur- und Integrationsarchitekturtechniken, wie z.B. SOA, Cloud, Mobile Apps, oder In-Memory Lösungen

Webentwickler (m/w)

Referenz 52519

Ihre Aufgaben:

- Neuentwicklung von Webanwendungen in HTML, CSS, Java Script, PHP
- Weiterentwicklung von Webanwendungen im Bereich Vertrags-, Dokumenten- und Lizenzmanagement

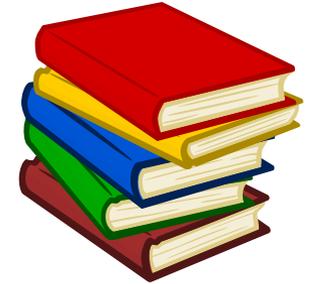
Fachliche Anforderungen:

- Abgeschlossenes Studium, z. B. der Informatik oder Wirtschaftsinformatik und/oder mehrjährige Berufserfahrung bezogen auf die Konzeption und Weiterentwicklung von Internetapplikationen
- Erfahrung in den gängigen Web-Technologien, wie HTML, CSS, XML. Außerdem Kenntnisse im Bereich der Sicherheit von Webanwendungen (XSS, CSRF, SQL Injections, Buffer Overflows)
- Gute Kenntnisse in JavaScript (jQuery), Ajax und SQL
- Erste Kenntnisse in Testing und Dokumentation von Softwareprojekten

Ihre Qualifikationen:

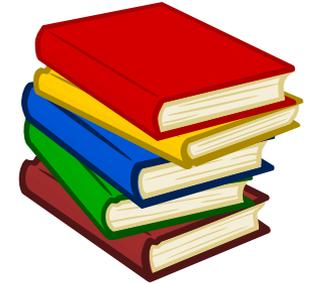
- Erfolgreich abgeschlossenes Studium der Informatik oder eine vergleichbare Ausbildung
- Fundierte Kenntnisse in der objektorientierten Entwicklung mit PHP sowie sicherer Umgang mit MySQL-Datenbanken, JavaScript, HTML und CSS-Kenntnisse
- Erfahrung mit Web-Frameworks sowie vertrautem Umgang mit Git
- Kenntnisse in der objektorientierten Entwicklung, wie z.B. SOA, Cloud, Mobile Apps, oder In-Memory Lösungen

Literatur



- Internetprogrammierung
 - R. W. Sebesta.
Programming the World Wide Web.
Pearson, 2015.
 - H. Balzert
Basiswissen Web-Programmierung.
Springer, 2011.

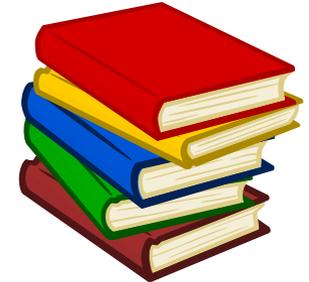
Literatur



- Semantic Web

- J. Domingue, D. Fensel, J.A. Hendler (Eds.).
Handbook of Semantic Web Technologies.
Springer, 2011.
- S. Groppe.
*Data Management and Query Processing in
Semantic Web Databases.*
Springer, 2011.

Literatur



- Cloud Computing
 - R. Wartala.
Hadoop: Zuverlässige, verteilte und skalierbare Big-Data-Anwendungen.
Open Source Press, 2012.
 - M. Kerzner, S. Maniyam.
Hadoop Illuminated.
[Open Source Hadoop Book](#), 2016.
- und jede Menge weiterer Bücher...

Konzept der Veranstaltung



- Trennung in
 - Vorlesung
 - 1-3 mal pro Vorlesungseinheit kleinere interaktive Tafelübungen
 - Rechnerübungen
 - „gelebte praktische Informatik“
 - direktes Einüben und Anwenden der in der Vorlesung vermittelten Konzepte, Programmiersprachen, Datenmodelle, Schnittstellen, Technologien...
 - „Hausaufgabe“
 - Programmierung der Bots eines Aufbauspiels in einem Code-Duell-System
 - Prüfungszulassung durch erfolgreiche Teilnahme

Konzept der Übung

- Restlos **alle Übungen am Rechner** im PC-Pool
- Alle Übungen **im Browser**
 - Tutorial-System
 - zur Auswahl einer Übungsaufgabe
 - zum direkten Editieren Ihrer Lösung
 - zum Vergleichen mit
 - der erwarteten Ausgabe
 - einer Beispiellösung
 - Dozent steht für Fragen zur Verfügung/
gibt Hilfestellung



Auswahl einer Übung

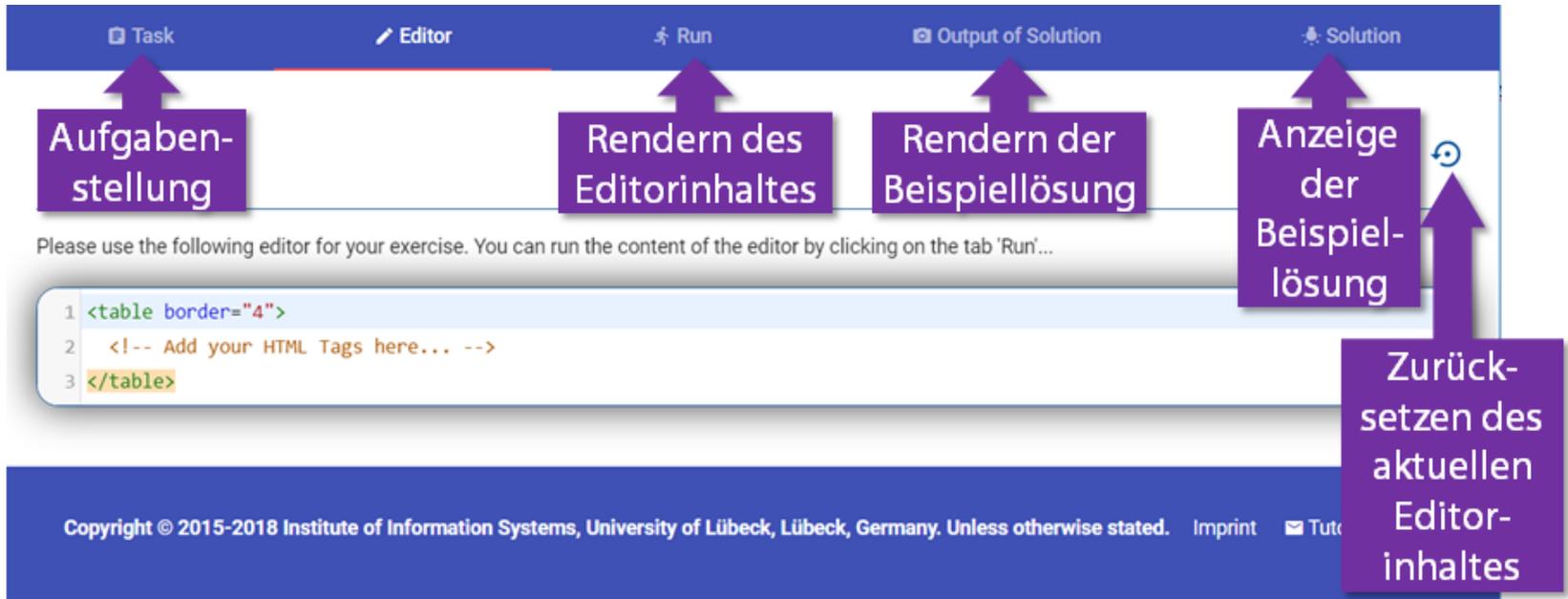
The screenshot shows a web browser at the URL <https://www.ifis.uni-luebeck.de/~groppe/webbased-information-systems/>. The page title is "Tutorial Webpage for Lecture" and the main heading is "Webbased Information Systems" by Sven Groppe. The page is part of the "Institute of Information Systems (IFIS)" and has a navigation bar with "Home", "Semantics", "Browser", and "Server".

Annotations on the page:

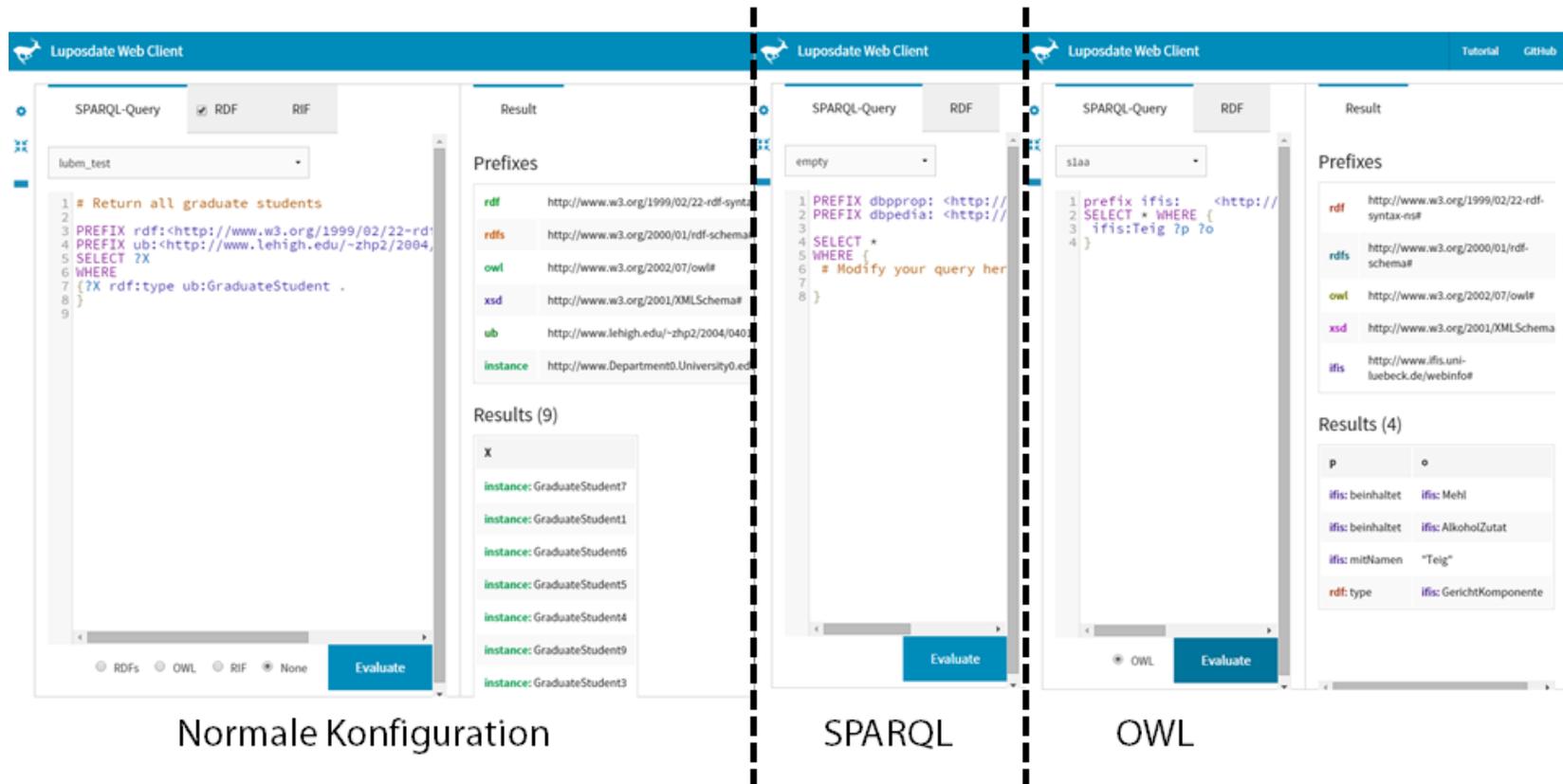
- Auswahl des übergeordneten Themenbereichs**: Points to the "Browser" link in the navigation bar.
- Auswahl der konkreten Übung**: Points to the "HTML" link in the "SHOW:" section.
- Starten der Aufgabe ⇒ Laden eines Coderrahmens in den Editor, der von Ihnen ergänzt werden muss**: Points to the "IN EDITOR" button in the "Solution:" section.
- Laden der Bsp.-Lsg. In Editor**: Points to the "IN EDITOR" button in the "Solution:" section.
- Ausgabe der Beispiel-lösung**: Points to the "OUTPUT" button in the "Solution:" section.

The "HTML EXERCISES" section contains "Exercise 1: HTML - Introduction" with the instruction: "Please add a paragraph with a short bio of yourself. Have fun!". Below this are buttons for "START", "OUTPUT", and "IN EDITOR".

Bearbeitung einer Aufgabe



Semantic Web-Übungen mit für die jeweilige Übung zugeschnittener Benutzeroberfläche



The image displays three side-by-side screenshots of the Luposdate Web Client interface, separated by vertical dashed lines. Each screenshot shows a different configuration for running a query.

Normal Configuration: The interface shows a SPARQL-Query editor with a dropdown menu set to 'luben_test'. The query is:


```
1 # Return all graduate students
2
3 PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX ub:<http://www.lehigh.edu/~zhp2/2004/04/ontology/ub.owl#>
5 SELECT ?X
6 WHERE
7 { ?X rdf:type ub:GraduateStudent .
8 }
9
```

 The result pane shows 9 results under the variable 'X', all of type 'instance: GraduateStudent'. The bottom of the interface has radio buttons for 'RDFs', 'OWL', 'RIF', and 'None', with 'None' selected. An 'Evaluate' button is at the bottom right.

SPARQL: The interface shows a SPARQL-Query editor with a dropdown menu set to 'empty'. The query is:


```
1 PREFIX dbpprop: <http://dbpedia.org/property/>
2 PREFIX dbpedia: <http://dbpedia.org/>
3
4 SELECT *
5 WHERE {
6 # Modify your query here
7 }
8
```

 The result pane is empty. The bottom of the interface has radio buttons for 'RDFs', 'OWL', 'RIF', and 'None', with 'None' selected. An 'Evaluate' button is at the bottom right.

OWL: The interface shows a SPARQL-Query editor with a dropdown menu set to 'saa'. The query is:


```
1 prefix ifis: <http://www.ifis.uni-luebeck.de/webinfo#>
2 SELECT * WHERE {
3 ifis:Teig ?p ?o
4 }
```

 The result pane shows 4 results in a table:

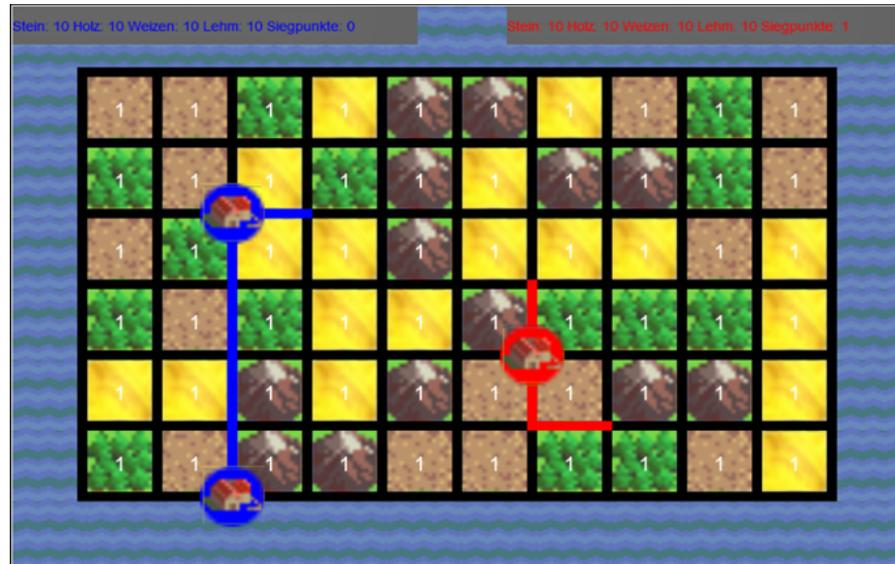
p	o
ifis: beinhaltet	ifis: Mehl
ifis: beinhaltet	ifis: AlkoholZutat
ifis: mitNamen	"Teig"
rdf: type	ifis: GerichtKomponente

 The bottom of the interface has radio buttons for 'RDFs', 'OWL', 'RIF', and 'None', with 'OWL' selected. An 'Evaluate' button is at the bottom right.

Code-Duell-System

- Aufbauspiel „Builders“
 - Inspiriert von Siedler von Catan
 - Rundenbasiert
 - Ein- und Ausgaben der zu entwickelnden Programme steuern Spielgeschehen
 - Testfälle müssen bestanden werden

Code-Duell-System - Spielidee und -regeln



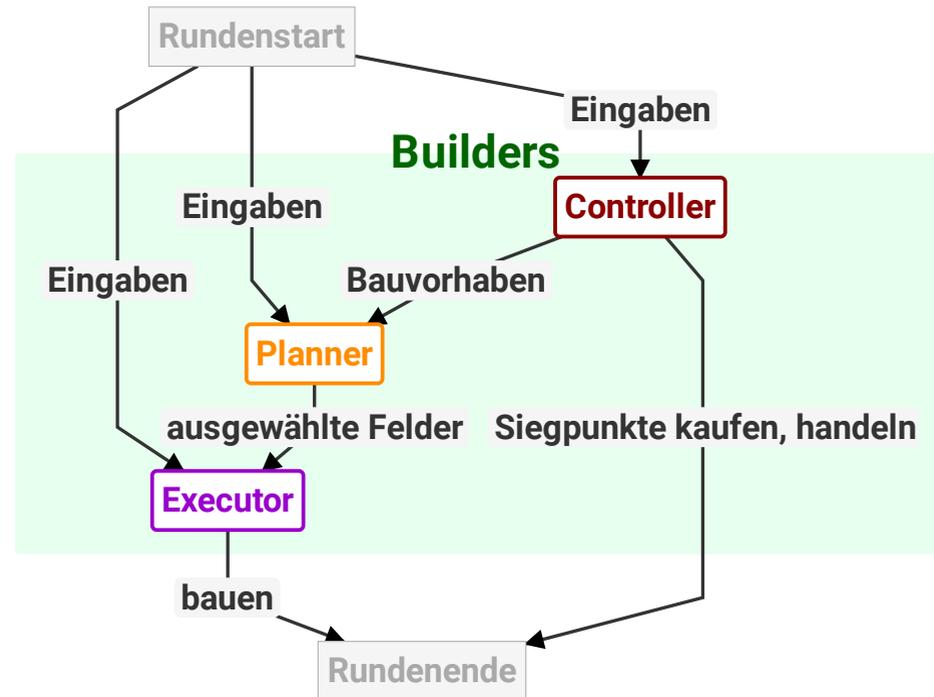
- Karte besteht aus Rohstofffeldern
- Spieler bauen Städte, um Rohstoffe abzubauen
- Städte müssen mit Straßen verbunden sein
- Rohstoffe können für Siegpunkte ausgegeben werden
- Spieler mit den meisten Siegpunkten gewinnt

Code-Duell-System - Teilspiele 1/2

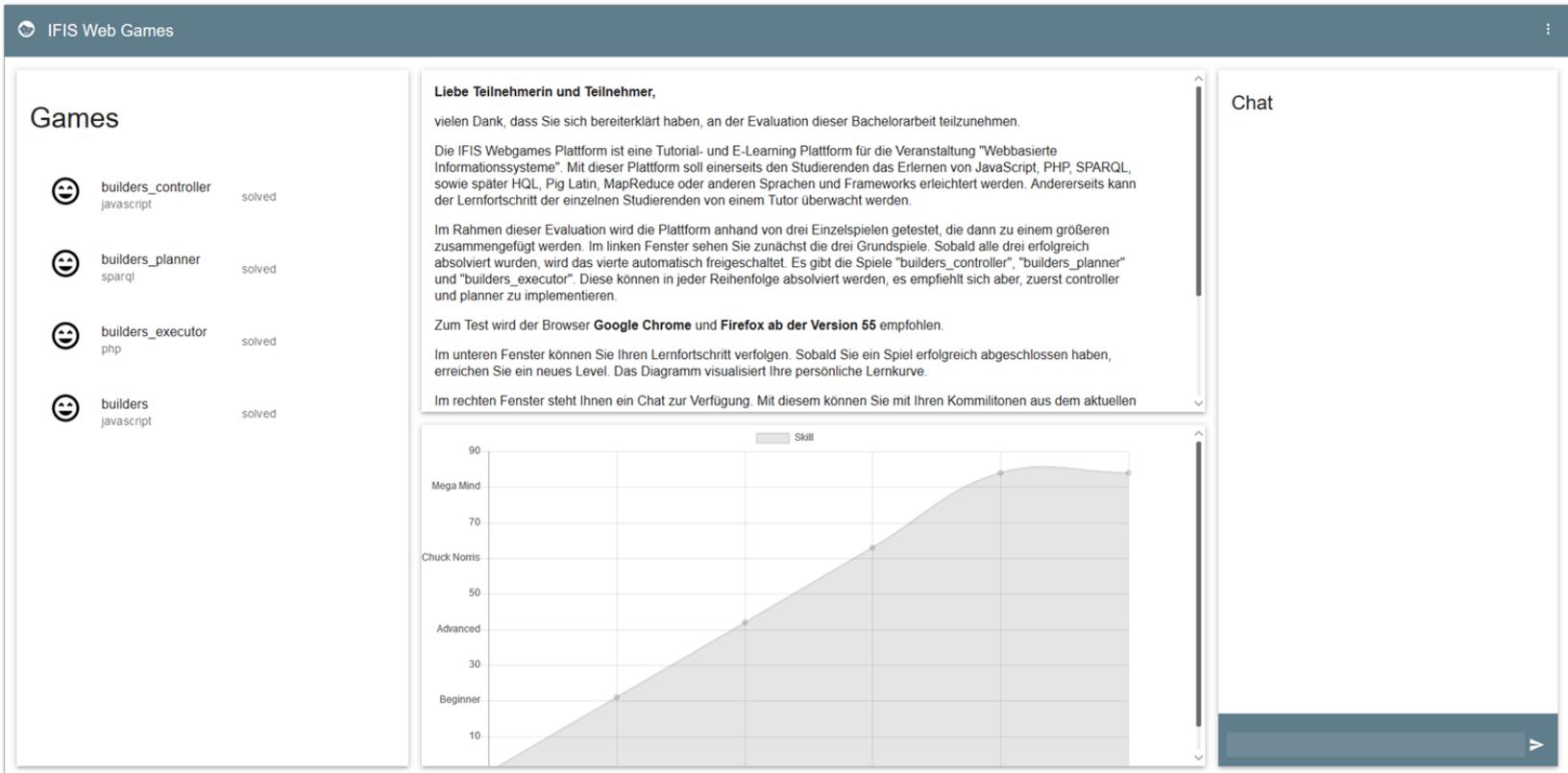
- Programmierung der KI von Bots von Teilspielen
 - SPARQL
 - JavaScript
 - PHP
- Alle 3 Teilspiele ergeben KI des Gesamtspiels

Code-Duell-System - Teilspiele 2/2

- **Controller**
 - Trifft Spielentscheidungen
 - Kauft Siegpunkte
- **Planner**
 - Wählt Bauplätze aus
- **Executor**
 - Setzt Bauvorhaben um



Code-Duell-System - Oberfläche nach dem Einloggen



The screenshot displays the IFIS Web Games interface. On the left, a 'Games' list shows four items, all marked as 'solved':

- builders_controller (javascript)
- builders_planner (sparql)
- builders_executor (php)
- builders (javascript)

The main content area contains a welcome message and instructions:

Liebe Teilnehmerin und Teilnehmer,

vielen Dank, dass Sie sich bereiterklärt haben, an der Evaluation dieser Bachelorarbeit teilzunehmen.

Die IFIS Webgames Plattform ist eine Tutorial- und E-Learning Plattform für die Veranstaltung "Webbasierte Informationssysteme". Mit dieser Plattform soll einerseits den Studierenden das Erlernen von JavaScript, PHP, SPARQL, sowie später HQL, Pig Latin, MapReduce oder anderen Sprachen und Frameworks erleichtert werden. Andererseits kann der Lernfortschritt der einzelnen Studierenden von einem Tutor überwacht werden.

Im Rahmen dieser Evaluation wird die Plattform anhand von drei Einzelspielen getestet, die dann zu einem größeren zusammengefügt werden. Im linken Fenster sehen Sie zunächst die drei Grundspiele. Sobald alle drei erfolgreich absolviert wurden, wird das vierte automatisch freigeschaltet. Es gibt die Spiele "builders_controller", "builders_planner" und "builders_executor". Diese können in jeder Reihenfolge absolviert werden, es empfiehlt sich aber, zuerst controller und planner zu implementieren.

Zum Test wird der Browser **Google Chrome** und **Firefox ab der Version 55** empfohlen.

Im unteren Fenster können Sie Ihren Lernfortschritt verfolgen. Sobald Sie ein Spiel erfolgreich abgeschlossen haben, erreichen Sie ein neues Level. Das Diagramm visualisiert Ihre persönliche Lernkurve.

Im rechten Fenster steht Ihnen ein Chat zur Verfügung. Mit diesem können Sie mit Ihren Kommilitonen aus dem aktuellen

Below the text is a 'Skill' progress chart showing a curve that rises from 'Beginner' to 'Mega Mind'.

Level	Approximate Skill Value
Beginner	10
Advanced	45
Chuck Norris	65
Mega Mind	85

The right side of the interface features a 'Chat' window, currently empty.

Code-Duell-System - Oberfläche für die Code-Entwicklung

IFIS Web Games
⋮

DESCRIPTION SIMULATION

Das Spiel

Im Builders Spiel soll eine Siedlungsgründung programmiert werden. Hierfür gibt es eine Karte, auf der verschiedene Rohstoffe verzeichnet sind, die an verschiedenen Standorten verfügbar sind.

Entlang der Rohstofffelder verlaufen Wege. Diese können zu Straßen ausgebaut werden. An den Ecken der Rohstofffelder können Städte gebaut werden, die die anliegenden Rohstoffe abbauen können. Die **erste Stadt** kann überall errichtet werden, anschließende Stadtgründungen müssen aber **mit einer Straße verbunden** werden.

Jede Runde wird zufällig ein Rohstoff ausgewählt, dieser wird jedem Spieler ausgegeben, der eine Stadt an einem solchen Rohstofffeld errichtet hat. Die Anzahl wird dabei durch den **Wert des jeweiligen Rohstofffeldes** festgelegt.

 Dieses Feld enthält den **Rohstoff Lehm** und den **Wert 1**. Wird Lehm als Rohstoff gewählt, erhält jeder der eine Stadt an diesem Feld besitzt 1 Lehm.

Die Rohstoffe

ID	Bild	Name
0		Stein
1		Holz
2		Weizen
3		Lehm

Bauplatzkoordinaten **starten bei 0**, ebenso Rohstofffeldkoordinaten. Die jeweils ersten und letzten Rohstofffelder einer Reihe und einer Spalte des Spielfeldes sind Meer-Felder.

Die Logik dieses Spiels wird in 3 Teile aufgeteilt, den **Controller (diesen Teil)** der grundlegende Entscheidungen trifft, den **Planner**, der Bauplätze auswählt und den **Executor**, der Bauvorhaben in die Tat umsetzt.

Code Editor

RESTORE SAVE

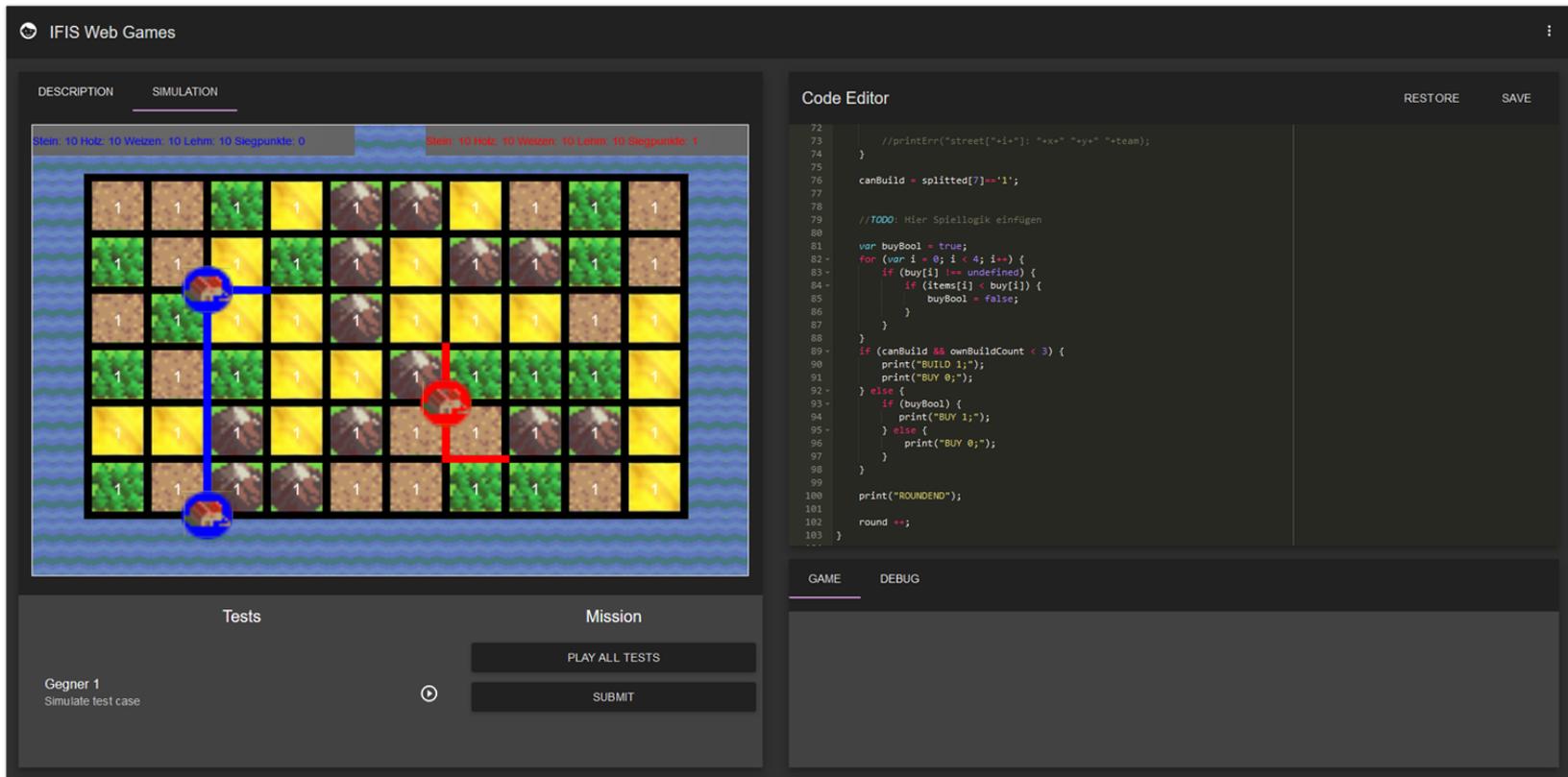
```

72
73 //println("street["+i+"]: "+x+" "+y+" "+team);
74 }
75
76 canBuild + splitted[7] += '1';
77
78
79 //TODO: Hier Spiellogik einfügen
80
81 var buyBool = true;
82 -
83 - for (var i = 0; i < 4; i++) {
84 -   if (buy[i] !== undefined) {
85 -     if (items[i] < buy[i]) {
86 -       buyBool = false;
87 -     }
88 -   }
89 -   if (canBuild && ownBuildCount < 3) {
90 -     print("BUILD 1;");
91 -     print("BUY 0;");
92 -   } else {
93 -     if (buyBool) {
94 -       print("BUY 1;");
95 -     } else {
96 -       print("BUY 0;");
97 -     }
98 -   }
99 - }
100 print("ROUNDEND");
101
102 round ++;
103 }

```

GAME
DEBUG

Code-Duell-System - Oberfläche für die Code-Entwicklung



The screenshot displays the IFIS Web Games interface. On the left, there's a 'DESCRIPTION' and 'SIMULATION' tab. The simulation shows a 10x10 grid with various terrain types (grass, dirt, water) and a path of blue and red lines connecting points. Below the grid are 'Tests' and 'Mission' sections with buttons for 'PLAY ALL TESTS' and 'SUBMIT'. On the right, a 'Code Editor' window shows JavaScript code for game logic, including a loop for buying items and a condition for building. The code is as follows:

```
72  
73 //println("street["+i+"]: "+x+" "+y+" "+team);  
74 }  
75  
76 canBuild = splitted[7]=='1';  
77  
78  
79 //TODO: Hier Spiellogik einfügen  
80  
81 var buyBool = true;  
82 for (var i = 0; i < 4; i++) {  
83   if (buy[i] !== undefined) {  
84     if (items[i] < buy[i]) {  
85       buyBool = false;  
86     }  
87   }  
88 }  
89 if (canBuild && ownBuildCount < 3) {  
90   print("BUILD 1;");  
91   print("BUY 0;");  
92 } else {  
93   if (buyBool) {  
94     print("BUY 1;");  
95   } else {  
96     print("BUY 0;");  
97   }  
98 }  
99  
100 print("ROUNDEND");  
101  
102 round ++;  
103 }
```

Code-Duell-System - Zulassung zur Prüfung

- Bestehen aller Testfälle für Einzelspiele im Code-Duell-System
- Einhalten der Deadlines
 - Deadlines werden noch bekannt gegeben
- Besiegen eines einfachen Gegners im Gesamtspiel

Code-Duell-System - Turnier

- Abgegebene Lösungen treten in einem Turnier an
- Jeder gegen jeden
- Einsehen der Ergebnisse möglich
- Kürung der Gewinner
 - in der letzten Vorlesungseinheit

Zusammenfassung

- Organisation
- Motivation von Webbasierten Informationssystemen
- Überblick über kommende
 - Vorlesungs- und
 - Übungsinhalte
- Lehrkonzept dieser Veranstaltung