

Mensch und Computer 2015

Usability Challenge 2015
Serious Gaming

Phil Köster
Carolin Meier
Anika Merkens
Dominique Berners

Überblick

- Ideenfindung
- Vorgehen
 - Zeitplanung
 - Ideenfindung
 - Grundkonzept
 - Papierprototyp
 - 1. Elektronischer Prototyp
 - 2. Elektronischer Prototyp
- Darstellung
- Fazit und Ausblick

Aufgabenstellung und Vorgehen

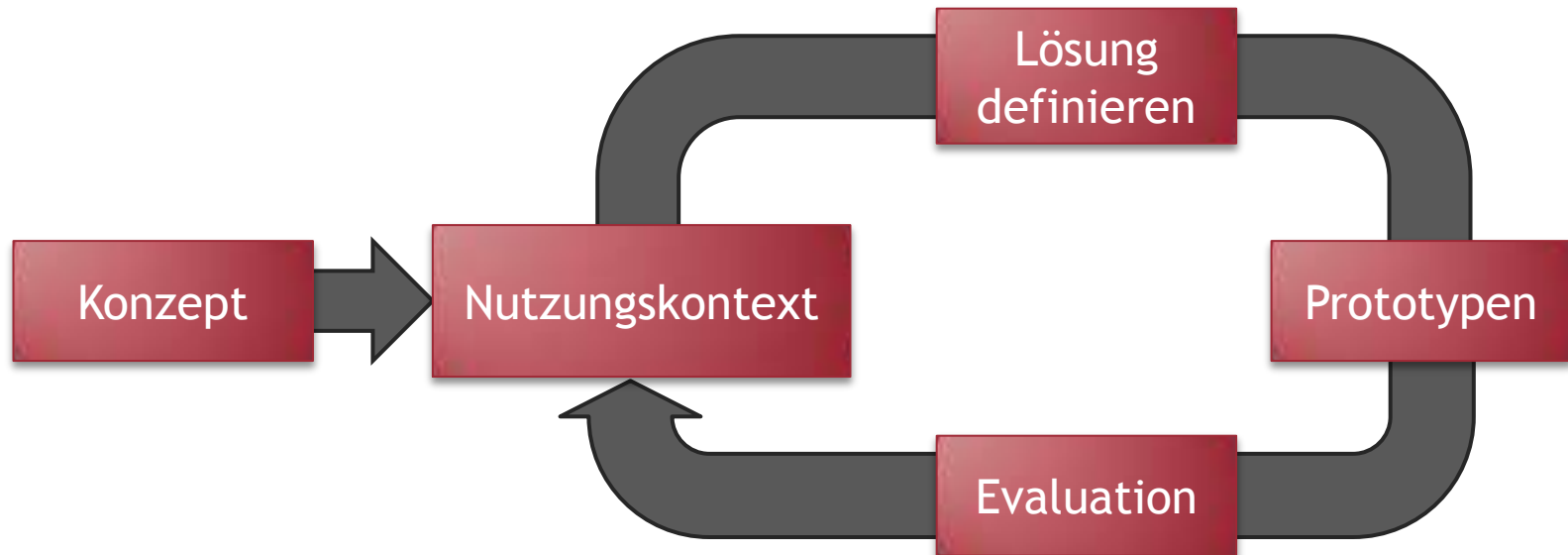
Usability Challenge

- Ausgerichtet von der Gesellschaft für Informatik (GI)
- Wettbewerb für Studierende
- Dokumentation der Konzeption und der Entwicklung einer Anwendung
- Jährlich wechselndes Thema
- Ziel: Auseinandersetzung mit Usability

Aufgabenstellung

- *„Konzeptentwicklung einer Anwendung, in der Ansätze aus dem Serious Gaming angemessen umgesetzt werden.“*
(Webpräsenz der Usability Challenge)
- Methodisch sauberes und gut dokumentiertes Vorgehen
- Mehrere Iterationen im User Centered Design
- Einreichen des Dokumentationsbeitrags

User Centered Design



Ideenfindung



Lernen
Lernen

Externes
Wissen

Infektions-
risiko

Simulation

Forschungs-
drang

Externes
Wissen

Zielgruppen
übergreifend

Rollen-
spiel
Informations-
Besieg
den
Prof

J
sprache
Doom vs Death

Simulation
Eigenes
Wissen

Wettbewerb

Kampf
um die
Plätze
(main FTH)

Quiz-
Game
=> Studium

Eigenes
Wissen

kurzfristig
Nutzen

Wettbewerb

Grundkonzept

- Lernspiel für die Programmiersprache Java
- Student kämpft sich durch Uni
- Spiele, die mit Java-Kenntnissen gelöst werden müssen (in Räumen der Uni)
- Freischalten neuer Räume bei Erfolg

Recherche - Ziel

- Interesse feststellen
- Anforderungen der Benutzer feststellen
- Diverse Fragen zu
 - Spielmodi
 - Darstellungsform
 - Design
 - Themen & Inhalt
 - Aufbau der Minispiele

4. Wie könnte man eine Wettbewerbssituation im Spiel hervorrufen?



5. Was würdest du bevorzugen, Singleplayer oder Multiplayer?


Singleplayer Multiplayer

6. Würdest du lieber auf einer mobilen Plattform spielen oder auf dem Desktop?

Mobil Desktop

7. Welche Darstellungsform würdest du dir dabei wünschen?



8. An welches Thema sollte das Design angelehnt sein? Wähle drei Favoriten (+) und 3 Themen die du ausschließt (-).

<input type="checkbox"/> Anime	<input type="checkbox"/> Sims	<input type="checkbox"/> Game of Thrones	<input type="checkbox"/> Star Wars
<input type="checkbox"/> Simpsons	<input type="checkbox"/> Super Mario	<input type="checkbox"/> Pokémon	<input type="checkbox"/> League of Legends
<input type="checkbox"/> Herr der Ringe	<input type="checkbox"/> Minions	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fragen

1. Wie hoch wäre dein Interesse ein Spiel zu dieser Thematik zu spielen?

- sehr niedrig niedrig mittel hoch sehr hoch

2. Aus welchem Grund würdest du dieses Spiel spielen?

- Lernunterstützung Lernerfolg reiner Spaß

3. Auf welche Art und Weise könnte man deiner Meinung nach den Aufbau von Teilspielen gestalten? (z.B.: ein Spiel, bei dem man Codeschnipsel in die richtige Reihenfolge ziehen muss)

4. Wie könnte man eine Wettbewerbssituation im Spiel hervorrufen?

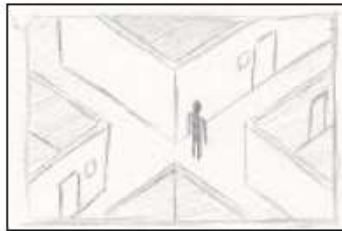
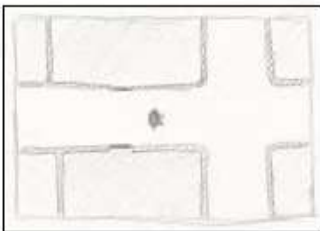
5. Was würdest du bevorzugen, Singleplayer oder Multiplayer?

- Singleplayer Multiplayer

6. Würdest du lieber auf einer mobilen Plattform spielen oder auf dem Desktop?

- Mobil Desktop

7. Welche Darstellungsform würdest du dir dabei wünschen?



????

8. An welches Thema sollte das Design angelehnt sein? Wähle drei Favoriten (+) und 3 Themen die du ausschließt (-).

- | | | | |
|---|--------------------------------------|--|--|
| <input type="checkbox"/> Anime | <input type="checkbox"/> Sims | <input type="checkbox"/> Game of Thrones | <input type="checkbox"/> Star Wars |
| <input type="checkbox"/> Simpsons | <input type="checkbox"/> Super Mario | <input type="checkbox"/> Pokémon | <input type="checkbox"/> League of Legends |
| <input type="checkbox"/> Herr der Ringe | <input type="checkbox"/> Minions | | |
| <input type="checkbox"/> _____ | <input type="checkbox"/> _____ | <input type="checkbox"/> _____ | |

9. Welche Themen würdest du dir zur Bearbeitung wünschen?

User-Recherche - Ergebnis

- Spiel als Lernunterstützung
- Singleplayer
- Desktopversion
- Isometrischer Darstellungsmodus
- Verspieltes Design (Super Mario)
- Unterschiedliche Ideen zu Minispielen

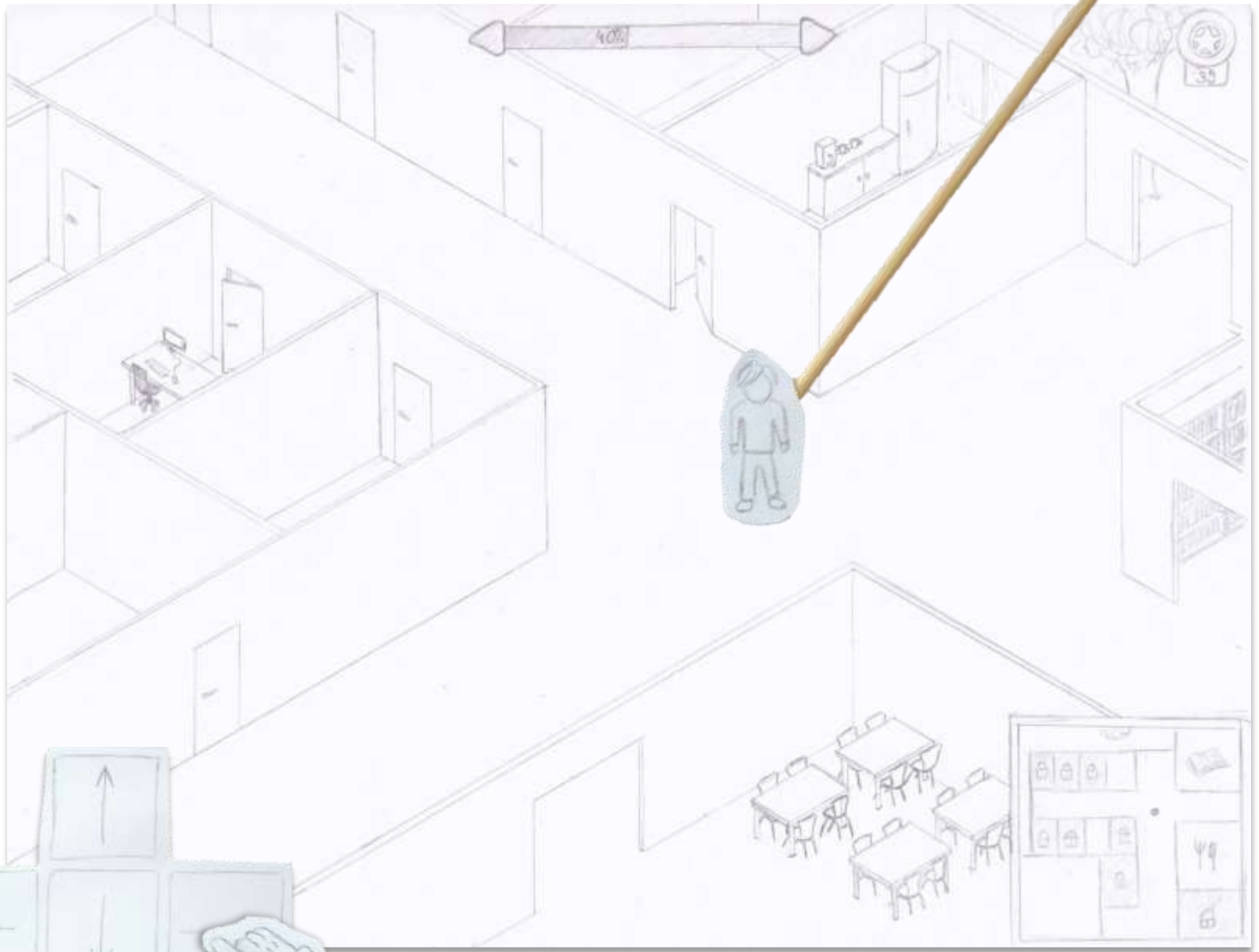
Papierprototyp

- Niedrige Hemmschwelle zum Kritisieren
- Steuerung mit Stäbchen und Berühren
- Fragestellungen:
 - Spaßfaktor groß genug?
 - Schwierigkeit angemessen?
- Zwei Unterschiedliche Testgruppen



Durchführung







Player 1

Fortschritt 40%



39



▼ Leistungen

Hörsaal freigeschaltet!

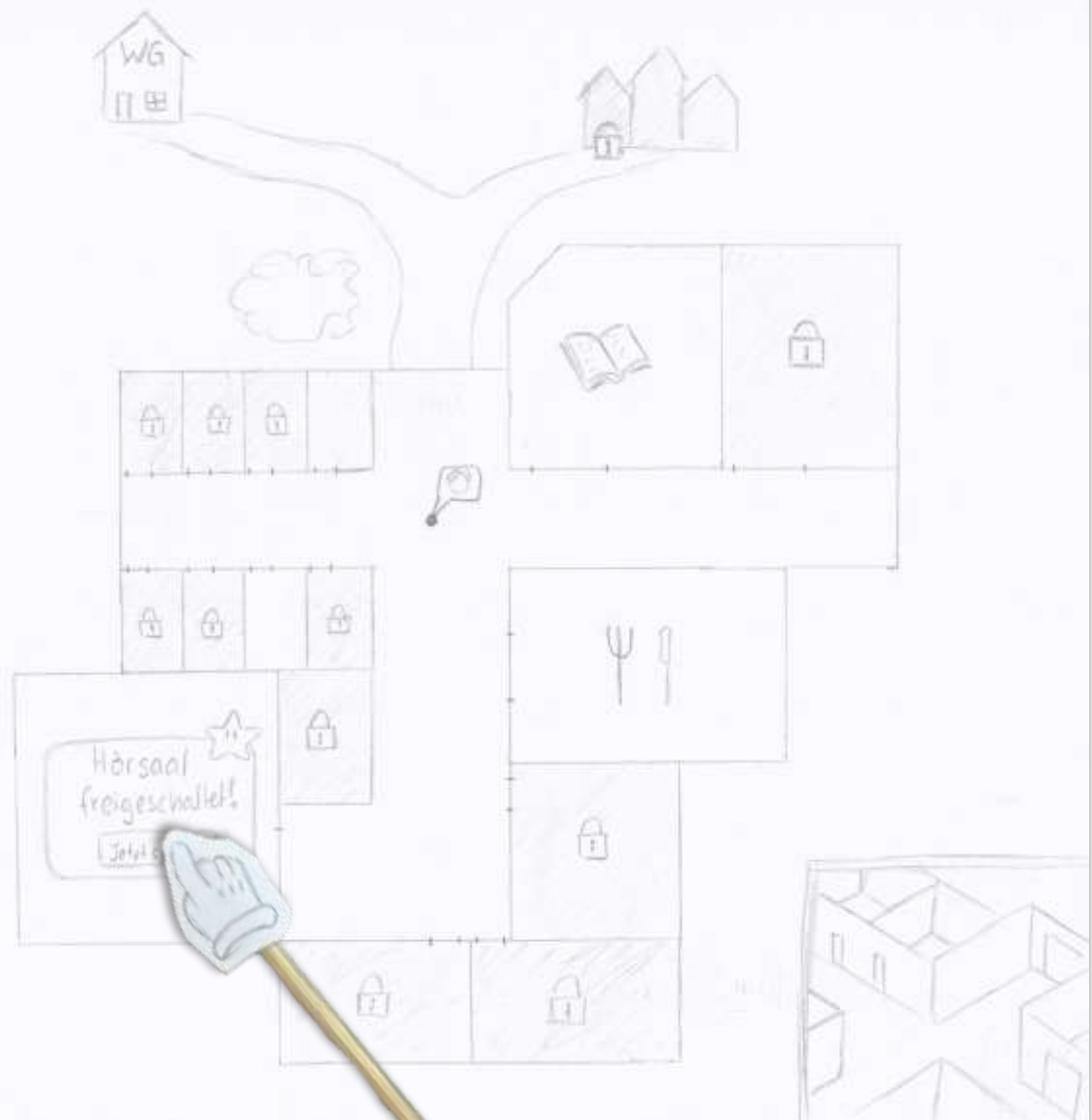
Datensalat geschafft!

► Abgeschlossene Themen

▼ Spiele

Datensalat Score 6820

Memory Score 5370



Papierprototyp - Auswertung

- Großes Interesse an Scores
- Zeitdruck überwiegend positiv aufgenommen
- Rahmenhandlung adäquat

- Längere Zurechtfindungsphase ohne Spieleinleitung
- Gute Zurechtfindung mit Einleitung

Elektronischer Prototyp

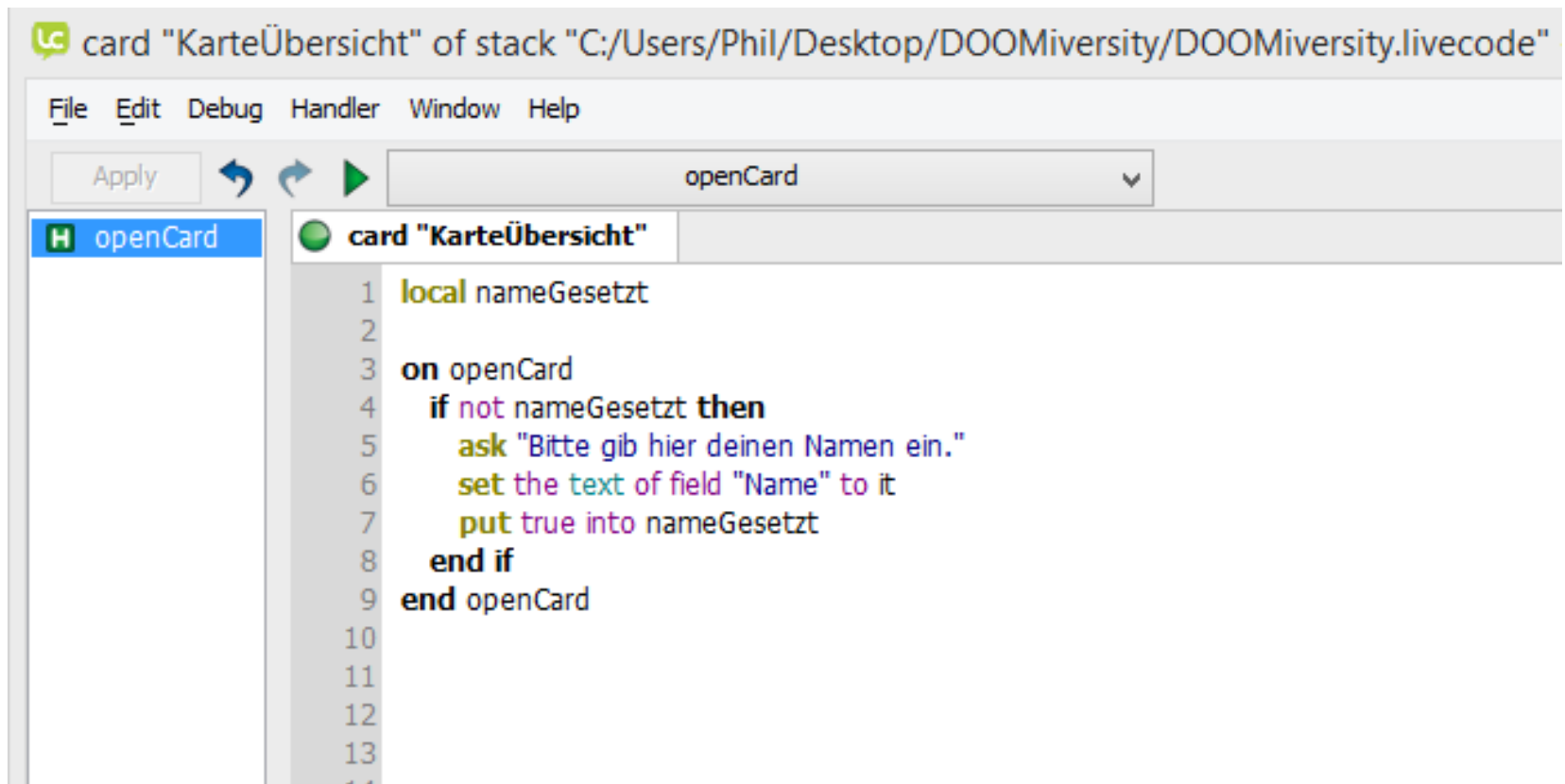
- Implementierung mit LiveCode
- Fragestellungen:
 - Gleiches Spielgefühl?
 - Unterschied zum Papierprototypen?
 - Spaßfaktor vorhanden?
 - Steuerung vergleichbar?
- Zwei Unterschiedliche Testgruppen

Warum LiveCode?

- Kickstarter Projekt
- Eigenrecherche
- Vorteile von LC:
 - Keine langwierige Einarbeitung
 - Gut verständlich, an Alltagssprache angelehnt
 - Dynamisches Prototyping-Tool



LiveCode - Eindruck



The screenshot shows the LiveCode IDE interface. At the top, the title bar reads "card 'KarteÜbersicht' of stack 'C:/Users/Phil/Desktop/DOOMiversity/DOOMiversity.livecode'". Below the title bar is a menu bar with "File", "Edit", "Debug", "Handler", "Window", and "Help". A toolbar contains an "Apply" button, a blue undo arrow, a blue redo arrow, a green play button, and a dropdown menu currently showing "openCard". On the left, a sidebar lists "H openCard" with a blue highlight. The main editor area shows the following code:

```
1 local nameGesetzt
2
3 on openCard
4   if not nameGesetzt then
5     ask "Bitte gib hier deinen Namen ein."
6     set the text of field "Name" to it
7     put true into nameGesetzt
8   end if
9 end openCard
10
11
12
13
14
```

Durchführung des ersten Elektronischen Prototypen



1. Elektronischer Prototyp - Auswertung

- Viele kleine Bugs
- Eine spielerklärende Einleitung fehlt
- Steuerung nicht zufriedenstellend umgesetzt
 - Pfeiltasten in Konflikt mit Laufrichtung
 - 2,5D



2. Elektronischer Prototyp

- Optimierung des ersten elektronischen Prototypen
- Fehlerbehebung
- Fragestellungen:
 - Steuerung angenehmer?
 - Spielspaß nur durch Bugs beeinträchtigt?
- Einführen einer Eingangsstory

Herzlich Willkommen in der DOOMiversity!



Du willst also lernen wie man in Java programmiert? Sehr gut! Dann bist du hier genau richtig!

Ich bin Professor M und stehe dir mit Rat und Tat zur Seite!

In der DOOMiversity wirst du immer mal wieder Räume mit offenen Türen finden! Dort warten tolle Spiele auf dich!



Wenn du die Spiele erfolgreich geschafft hast, öffnet sich irgendwo anders eine neue Tür!



Wenn du alle Spiele erfolgreich gespielt hast, wirst du bestimmt ein toller Programmierer sein!



Doch sei gewarnt: Bevor du deinen Abschluss machen kannst, musst du noch Professor Lookigi besiegen!



Ich hoffe du hast alles verstanden! Los geht's!



Tipp: Guck am Besten als erstes in der Fachschaft vorbei: Ich bräuchte da deine Hilfe bei etwas!

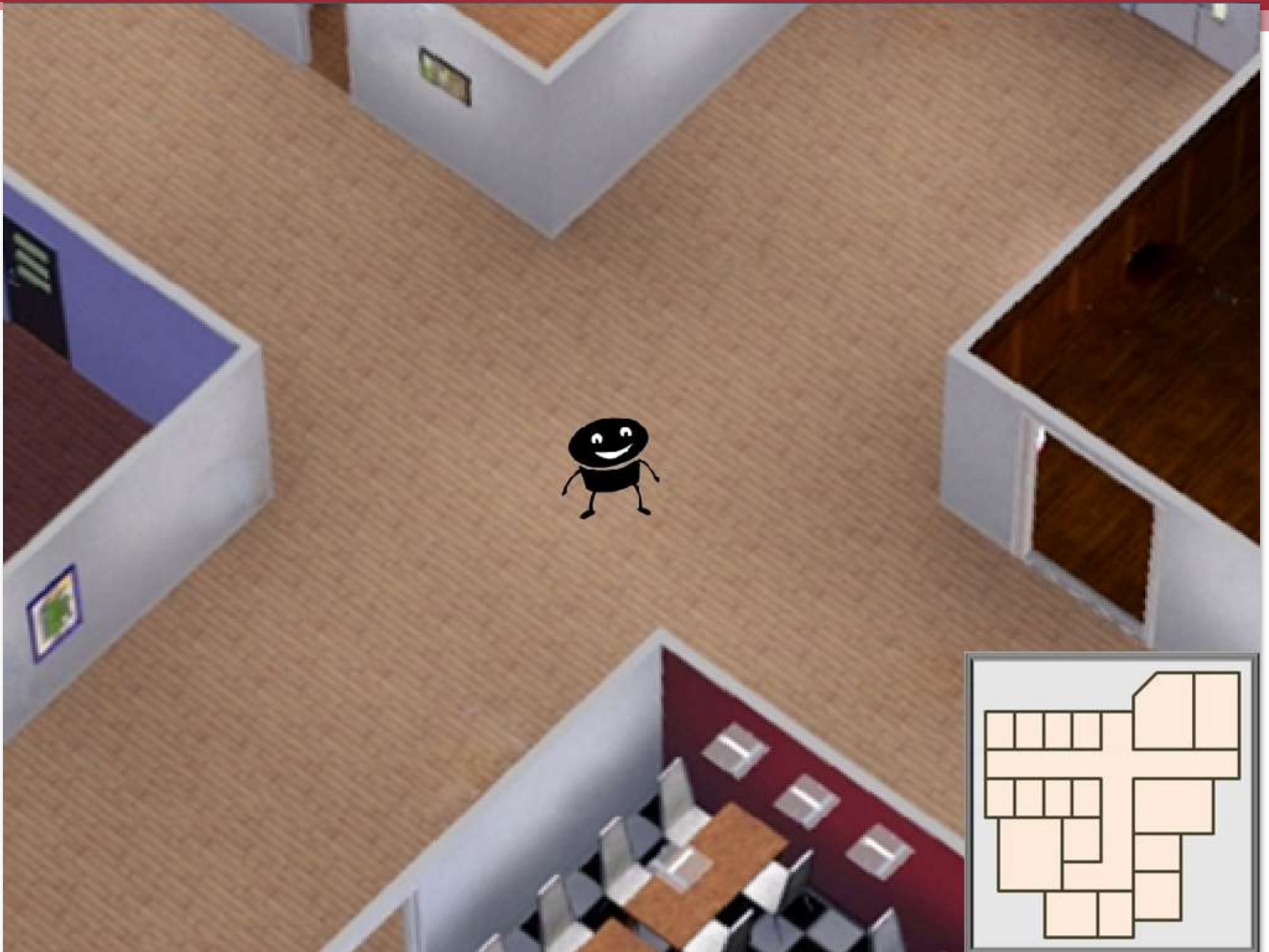
2. Elektronischer Prototyp - Auswertung

- Steuerung besser
- Einleitung angenehm aufgefallen
- Spiele ohne Fehler deutlich positiver aufgenommen

Gesamtes Spiel

Von der Papierskizze bis zum
elektronischen Prototypen







Fortschritt:



Coins:



0

Leistungen Themen Spiele

Wenn du deine Leistungen sehen und mit anderen vergleichen willst...

... dann kauf dir jetzt:

Die

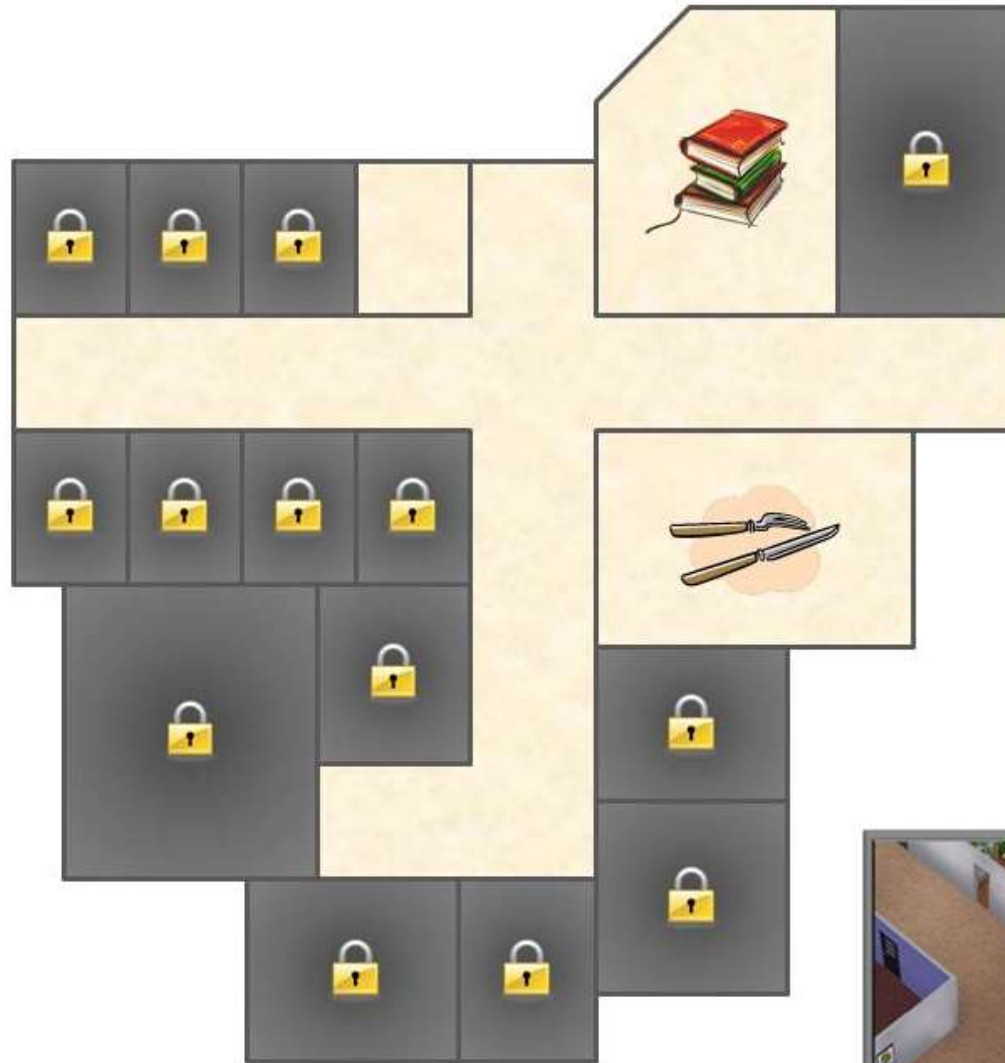
ULTIMATIVE

AEMBERAUBENDE

ULTRAKRASSE

Vollversion.

Für nur 29 Millionen Pokédollar.
Und 17 Pokécents.



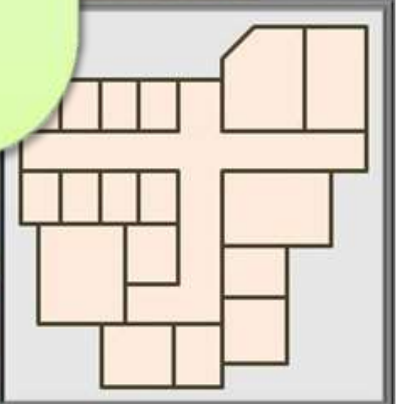


Herzlich Willkommen in der DOOMiversity!



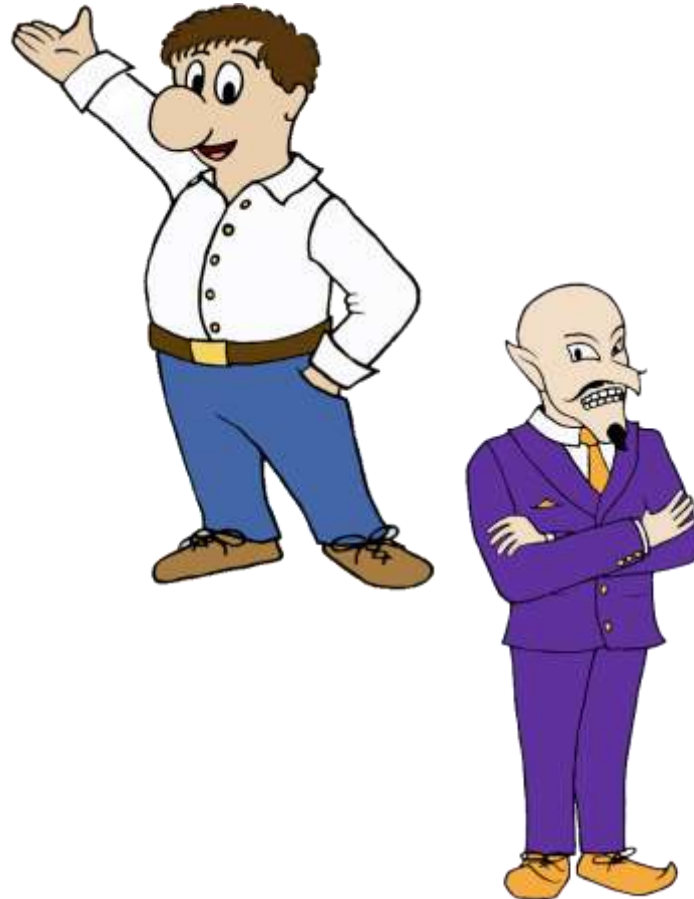
Du willst also lernen wie man in Java programmiert? Sehr gut! Dann bist du hier genau richtig!

Ich bin Professor M und stehe dir mit Rat und Tat zur Seite!





Darstellung

Minispiel:
Datensalat



Datensalat

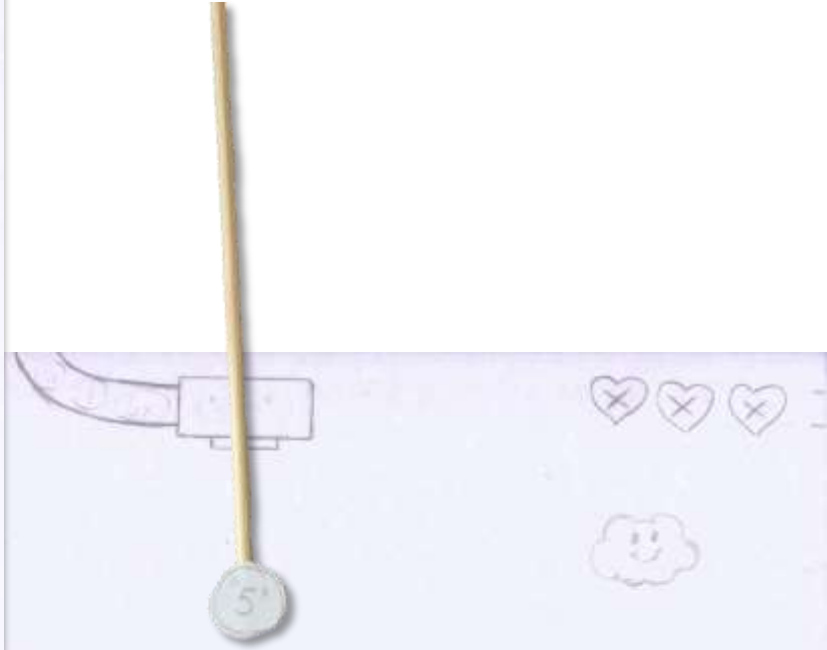
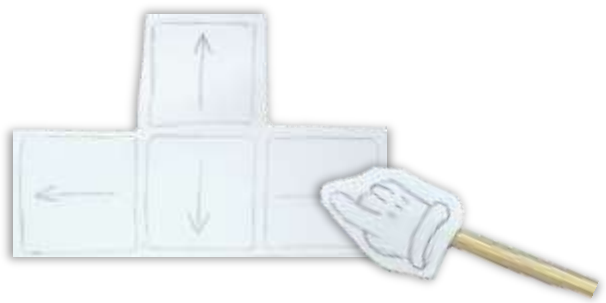
Meine Daten sind durcheinander gekommen!
Kannst du mir helfen sie wieder zu sortieren?

Die herunterfallenden Daten müssen mit den Pfeiltasten   in die Behälter der passenden Datentypen gelenkt werden!

Fehlerhafte Daten kommen in den Behälter mit dem 'X'!

Mit  kannst du das Fallen beschleunigen



Start




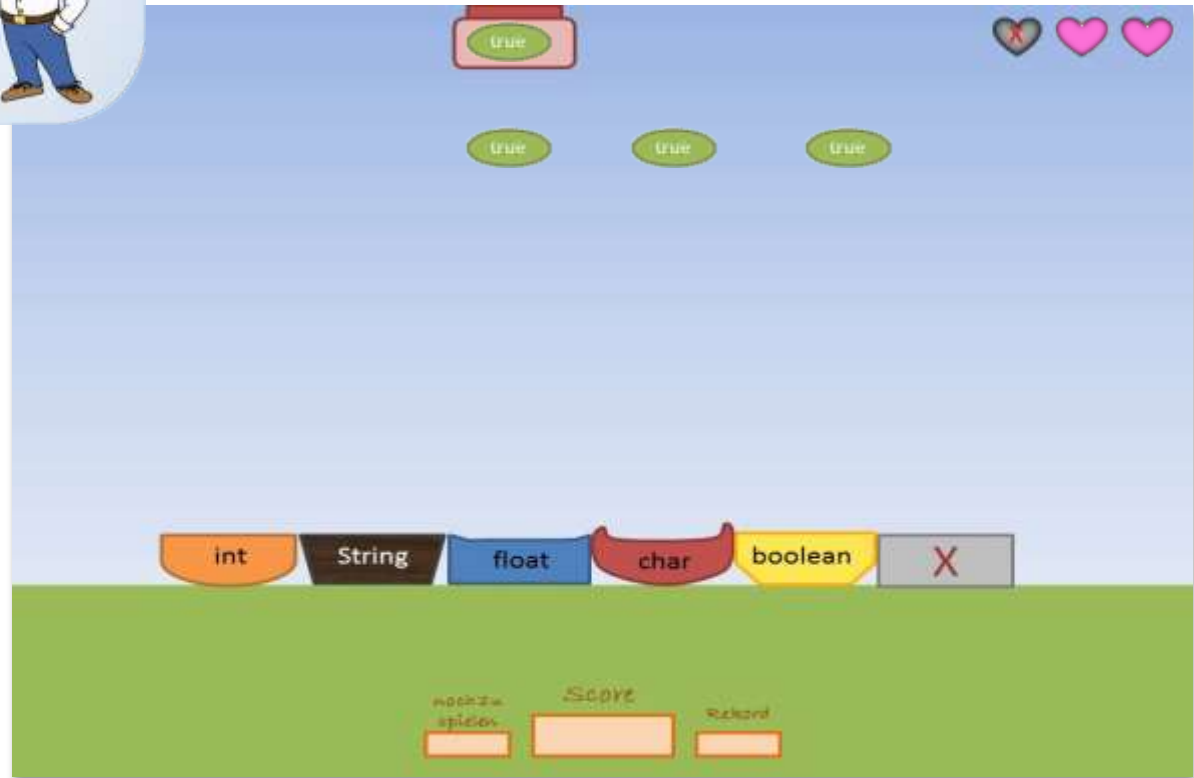
Meine ganzen Daten sind durcheinander gekommen!
Kannst du mir helfen sie wieder zu sortieren?



Du musst die herunterfallenden Daten in die passenden Datentypen-Behälter einsortieren.

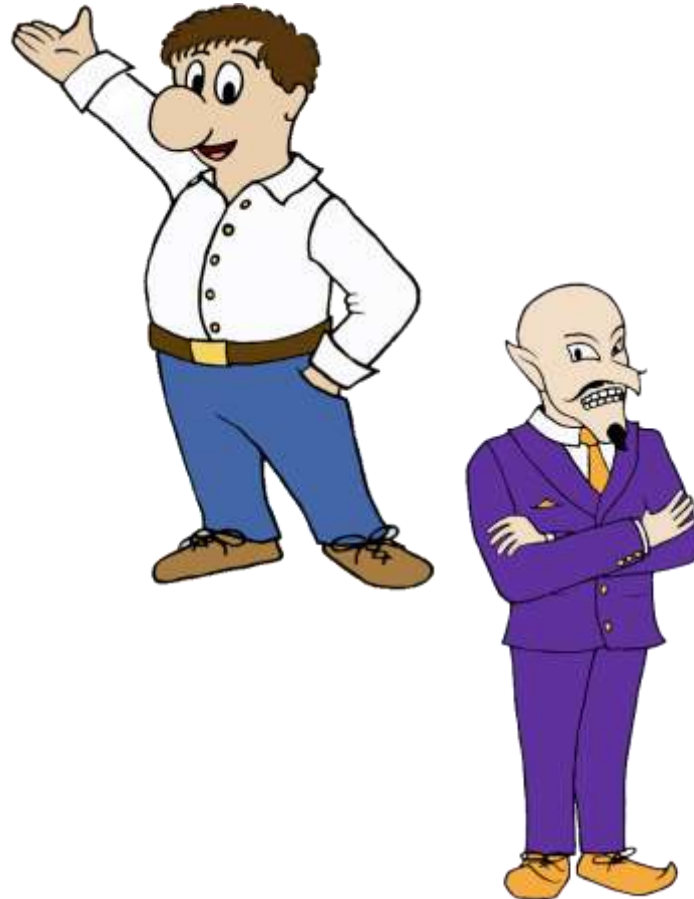
Beweg sie dazu einfach mit den Pfeiltasten   nach links oder rechts, bis sie über dem richtigen Behälter sind.

Dann kannst du sie mit der Pfeiltaste  nach unten fallen lassen!



Darstellung


Minispiel:
Codeschredderer



CODESCHREIBER mit Lakagi

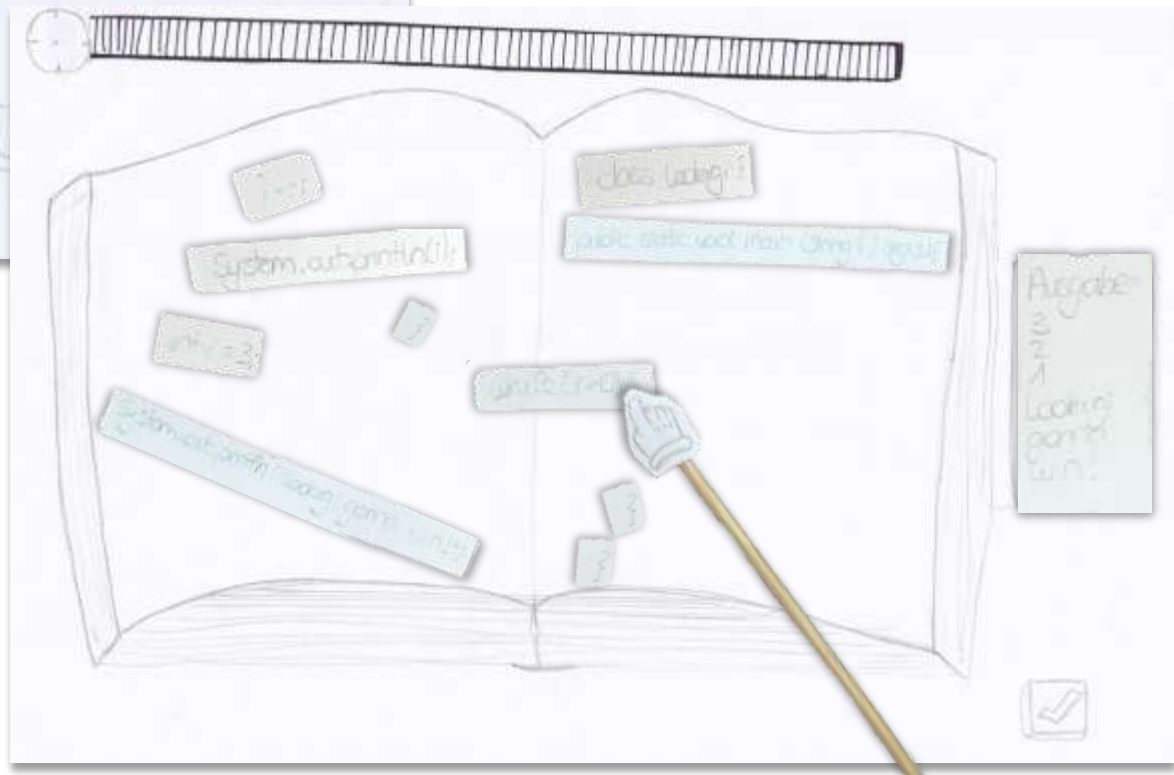


Wahahaha, ich habe deinen Code zer schnitten. Gross ist meine Spezialität, wie du weisst. Du solltest dich lieber beeilen das wieder in Ordnung zu bringen, wenn du dein Studium hier beenden willst. Um dich nicht ganz dumm darstellen zu lassen gebe ich dir eine Woche vor. Dann musst du darauf aufbauend deinen Code in Ordnung bringen.

Die Meisterhand  steht dir für Drop & Drop zur Verfügung. Und jetzt,...

...wirst du verlieren. Mach dich bereit. Wahahaha.

Ich mach das lieber später.



Haha! Ich hab deinen Code zerschreddert! Guck doch selbst, ob du ihn wieder zusammensetzen kannst!



Ziehe dazu die Code-Schnipsel auf die rechte Seite und sortiere sie!

Auf der rechten Seite findest du die Ausgabe, welche dein Code nachher erzeugen soll!

Ausgabe
Hier steht dann ein Text!



Wenn du glaubst, dass alles stimmt, dann klick das Häkchen!

while(i>0){

int i = 3;

System.out.println(i);

i--;

System.out.println("Lookigi gonna win!");

}

```
class Lookigi{
public static void main{
```

Ausgabe

3

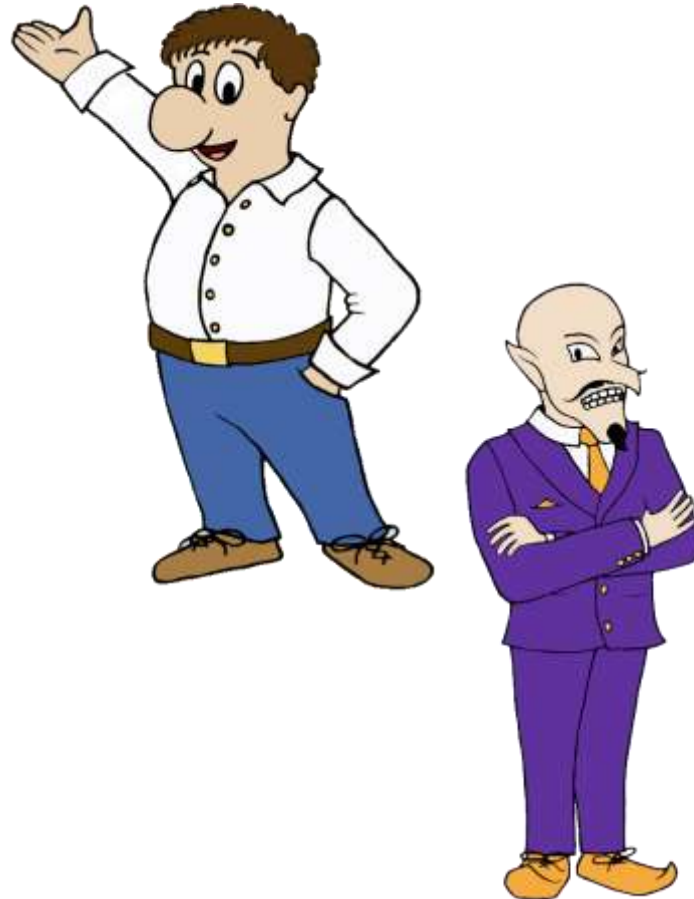
2

1

Looki
gi
gonna
win!

Darstellung

Minispiel:
DOOMemory



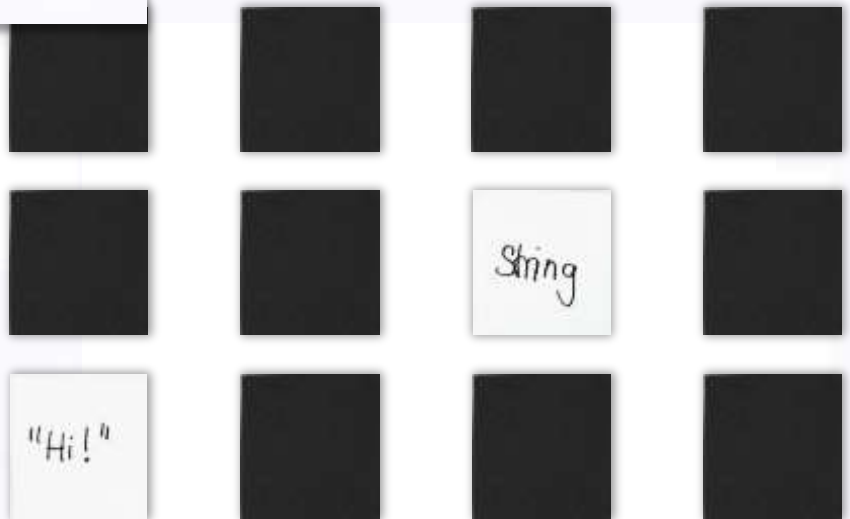


DOM Memory

Versuche durch anklicken die richtigen Pfläuschen zu finden. Zu jedem Dekortypen gibt es ein Beispiel.

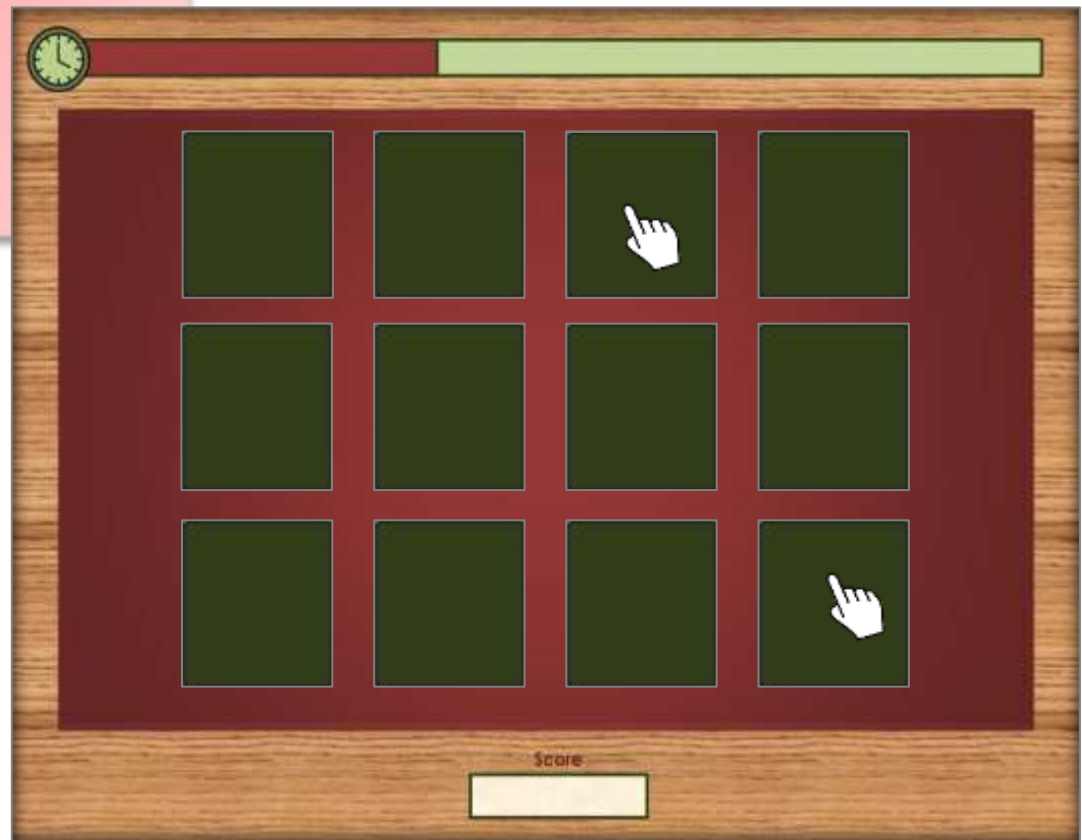
Start

Ich möchte später spielen



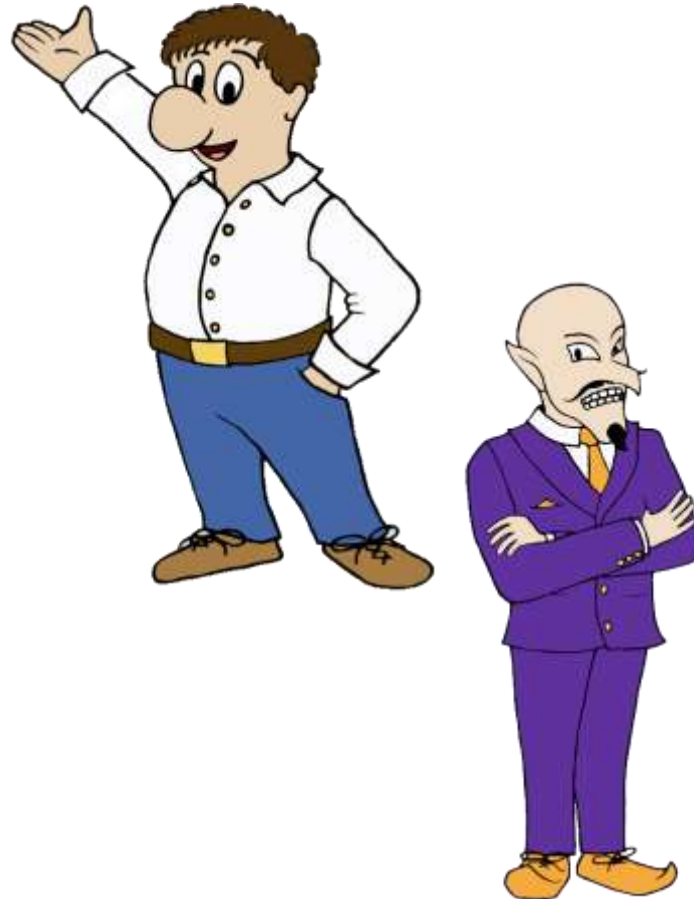
DOOM
Memory

Finde durch anklicken ein passendes Pärchen.
Zu jedem Datentyp gibt es ein Beispiel

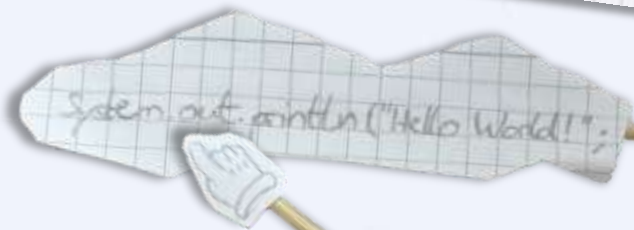
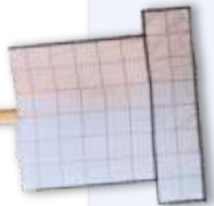


Darstellung

Minispiel:
Error Killer



Error Killer



1800
Score

Java-Laufzeitumgebung ♡ ♡ ♡

Oh nein! Der Compiler ist ausgefallen und nun kümmert sich niemand mehr darum, dass fehlerhafter Code nicht bis in das Java-Laufzeitsystem gelangt. Kannst du helfen? Der Compiler ist schließlich dein Freund! Schieße durch Antikörper die nicht kompromissbaren Codefragmente ab, um zu verhindern, dass sie in der Laufzeitumgebung Schaden anrichten. Du hast drei Leben! Auch für das Abschießen eines kompromissbaren Codefragmentes verlierst du ein Leben.

Oh nein! Der Compiler ist ausgefallen!
Nun kümmert sich niemand mehr darum, dass fehlerhafter Code
nicht bis in das Java-Laufzeitsystem gelangt.

Kannst du helfen? Der Compiler ist schließlich dein Freund!



Schieße durch Anklicken die nicht kompilierbaren Codefragmente ab,
um zu verhindern, dass sie in der Laufzeitumgebung Schaden anrichten.

Java - Laufzeitumgebung

Du hast drei Leben! Auch für das Abschießen eines
kompilierbaren Codefragmentes verlierst du ein Leben.

Jetzt nicht

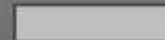
Start

Error Killer

```
char c = "T"
```



Score



Java - Laufzeitumgebung

Elektronischer Prototyp Minispiele

1. Elektronischer Prototyp

- Rasterung bei Datensalat fehlerhaft
- Erklärungen nicht eindeutig
- Zeitleisten konnten per Pfeiltasten manipuliert werden
- Keine Scores
- Teilweise Spiele zu schnell
- Kein Feedback

2. Elektronischer Prototyp

- Rasterung aufgehoben
- Erklärung optimiert
- Zeitleiste kann nicht mehr manipuliert werden
- Scores eingefügt
- Hilfe-Button mit Spielerklärung hinzugefügt
- Tempo reguliert
- Feedback für Richtig und Falsch

Fazit und Ausblick

Fazit

- Oberste Priorität: Ansichten der Probanden
- Gute Usability vermeidet Frust
- Anwenden von Vorlesungsinhalten (MCI)
- Strukturiertes Vorgehen und laufende Dokumentation sind hilfreich
- Arbeit in Gruppe und mit Testpersonen anregend und aufschlussreich

Ausblick

- DOOMisversity2 **LiveDemo**

Quellenverzeichnis

- Webpräsenz der Usability Challenge 2015 (www.usability-challenge.de, zuletzt abgerufen am: 23.03.2015)
- „LiveCode Platform for Developers“, Webpräsenz von LiveCode Ltd. (livecode.com/products/livecode-platform/livecode-for-developers, zuletzt abgerufen am: 23.03.2015)
- Historie von LiveCode (<http://livecode.com/about/>, zuletzt abgerufen am: 04.04.2015; <http://thenextweb.com/insider/2013/01/29/runrev-launches-kickstarter-campaign-to-create-open-source-version-of-livecode/>, zuletzt abgerufen am: 04.04.2015)
- DIN EN ISO 9241-210 - Prozess zur Gestaltung gebrauchstauglicher interaktiver Systeme (http://www.ergo-online.de/html/software/software_entwicklung_prototyp/benutzerzentrierte_softwarege.htm, zuletzt abgerufen am: 02.04.2015)
- User Centered Design (http://de.wikipedia.org/wiki/Nutzerorientierte_Gestaltung, zuletzt abgerufen am: 12.03.2015; <http://danielikim.com/what-is-user-centered-design/>, zuletzt abgerufen am: 02.04.2015)

Herzlichen Dank für Ihre
Aufmerksamkeit!

