UNIKASSEL VERSITÄT

SOFTWARE ENGINEERING
UNIVERSITÄT KASSEL

Programmiermethodik

Übung 7

Wintersemester 2012 / 13

Fachgebiet Software Engineering

Tobias George

george@uni-kassel.de

Agenda

- **Besprechung HA 5**
- **Entwicklung von grafischen Oberflächen**
  - Mock-Ups
  - GUI Builder
    - Swing
    - SWT
  - Demo
- **Praktische Übung: Erstellen eines Diaballik Login Screens**
- **Vorschau HA 6**

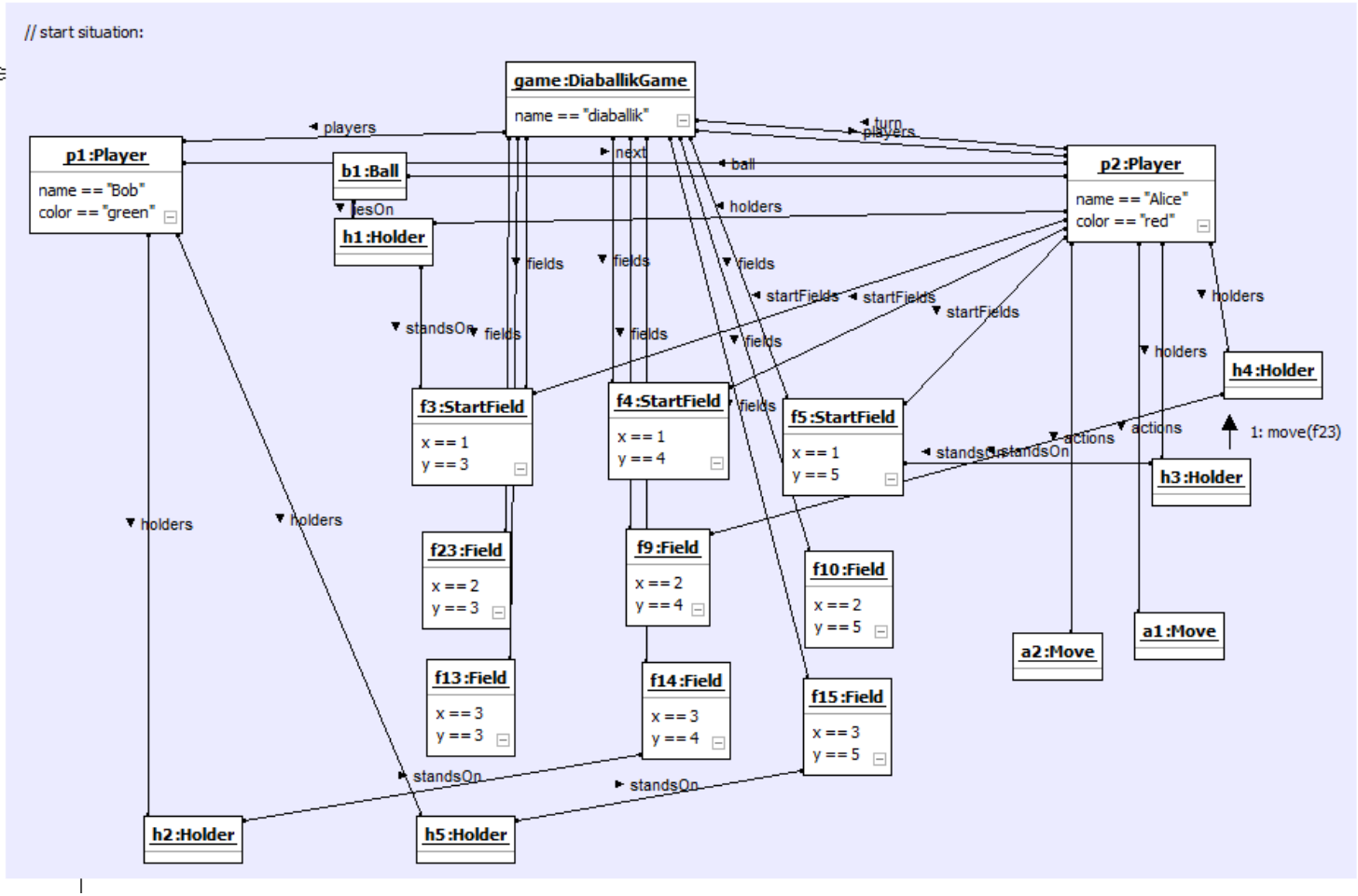
Besprechung HA5 I

- **SDMLib:**

```
@Test
public void horizantalMoveTest() {

    Scenario horizontalMoveTestScenario = new Scenario("genSrc");

    //Startsituation
    DiaballikGame game = new DiaballikGame()
    .withName("diaballik");


    Player p1 = new Player()
    .withName("Alice")
    .withColor("red")
    .withGame(game)
    .withCurrentGame(game)
    .withActions(new Move())
    .withActions(new Move());

    Player p2 = new Player()
    .withName("Bob")
    .withColor("green")
    .withGame(game);


    StartField f13 = (StartField) new StartField().withGame(game).withX(1).withY(3);
    Field f23 = new Field().withGame(game).withX(2).withY(3);
    Field f33 = new Field().withGame(game).withX(3).withY(3);

    StartField f14 = (StartField) new StartField().withGame(game).withX(1).withY(4);
    Field f24 = new Field().withGame(game).withX(2).withY(4);
    Field f34 = new Field().withGame(game).withX(3).withY(4);

    StartField f15 = (StartField) new StartField().withGame(game).withX(1).withY(5);
    Field f25 = new Field().withGame(game).withX(5).withY(3);
    Field f35 = new Field().withGame(game).withX(5).withY(4);


    Holder h3 = new Holder().withPlayer(p1).withField(f13);
    Holder h4 = new Holder().withPlayer(p1).withField(f24);
    Holder h5 = new Holder().withPlayer(p1).withField(f15);
    Holder h9 = new Holder().withPlayer(p2).withField(f34);
    Holder h10 = new Holder().withPlayer(p2).withField(f35);

    Ball b1 = new Ball().withPlayer(p1).withHolder(h3);

    horizontalMoveTestScenario.addObjectDiag(game);

    //Aktion
    h4.move(f23);
    horizontalMoveTestScenario.addObjectDiag(game);

    //Endsituation
    Assert.assertTrue(p1.getActions().size() == 1);
    Assert.assertTrue(h4.getField().equals(f23));

    horizontalMoveTestScenario.dumpHTML();
}
```

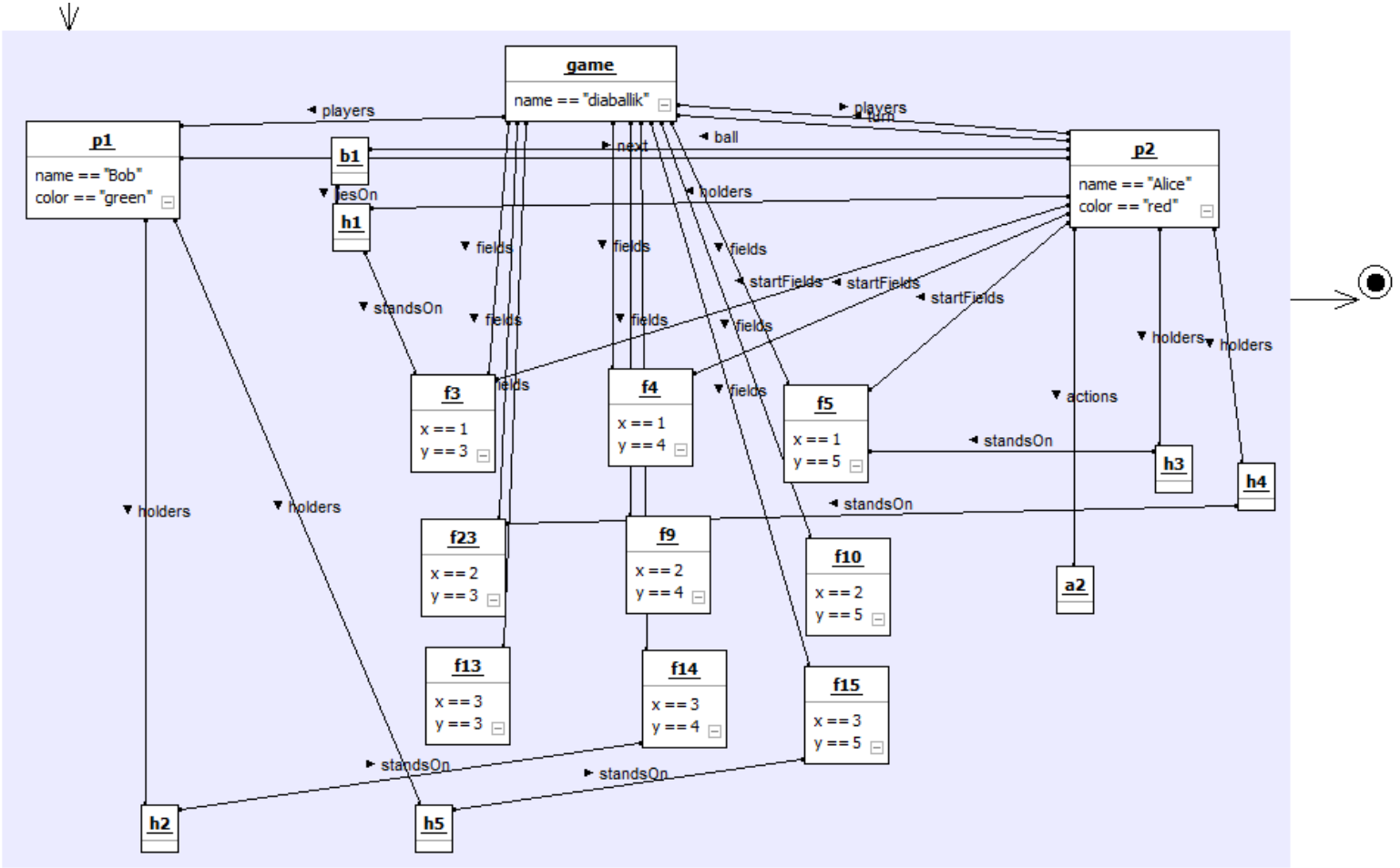
Besprechung HA5 II

- **Fujaba Startsituation:**



Besprechung HA5 III

- **Fujaba Endsituation:**



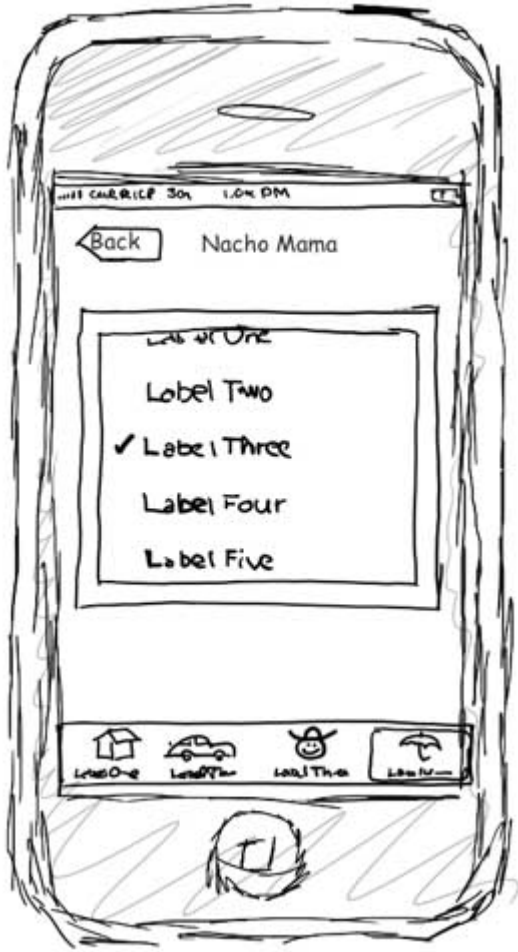
Besprechung HA5 IV

- **Aufgabe 2 Musterlösung Horizontales Bewegen:**

```
public void move( Field p0 )
{
    for(Action move : getPlayer().getActions())
    {
        //wenn der Spieler der aktuelle Spieler ist und ein Move-Objekt übrig hat
        if((getPlayer().getCurrentGame() != null) &&
           (move instanceof Move))
        {
            //wenn auf dem Zielfeld kein Spielstein steht und
            //es links oder rechts neben dem aktuellen Feld liegt
            if((p0.getHolder() == null) &&
               ((p0.getY() == getField().getY() + 1)  ||
               (p0.getY() == getField().getY() - 1)))
            {
                //verschiebe den Spielstein und lösche das Move-Objekt
                setField(p0);
                move.removeYou();
                break;
            }
        }
    }
}
```

Entwicklung von grafischen Oberflächen – Mock-Ups

- Bevor man anfängt zu coden: „Mock-Ups“
- In der Realität: Designer != Entwickler
- Erstellung von Mock-Ups kostet deutlich weniger Zeit
- Erleichtert die Implementierung, weil man schon weiß was herauskommen soll

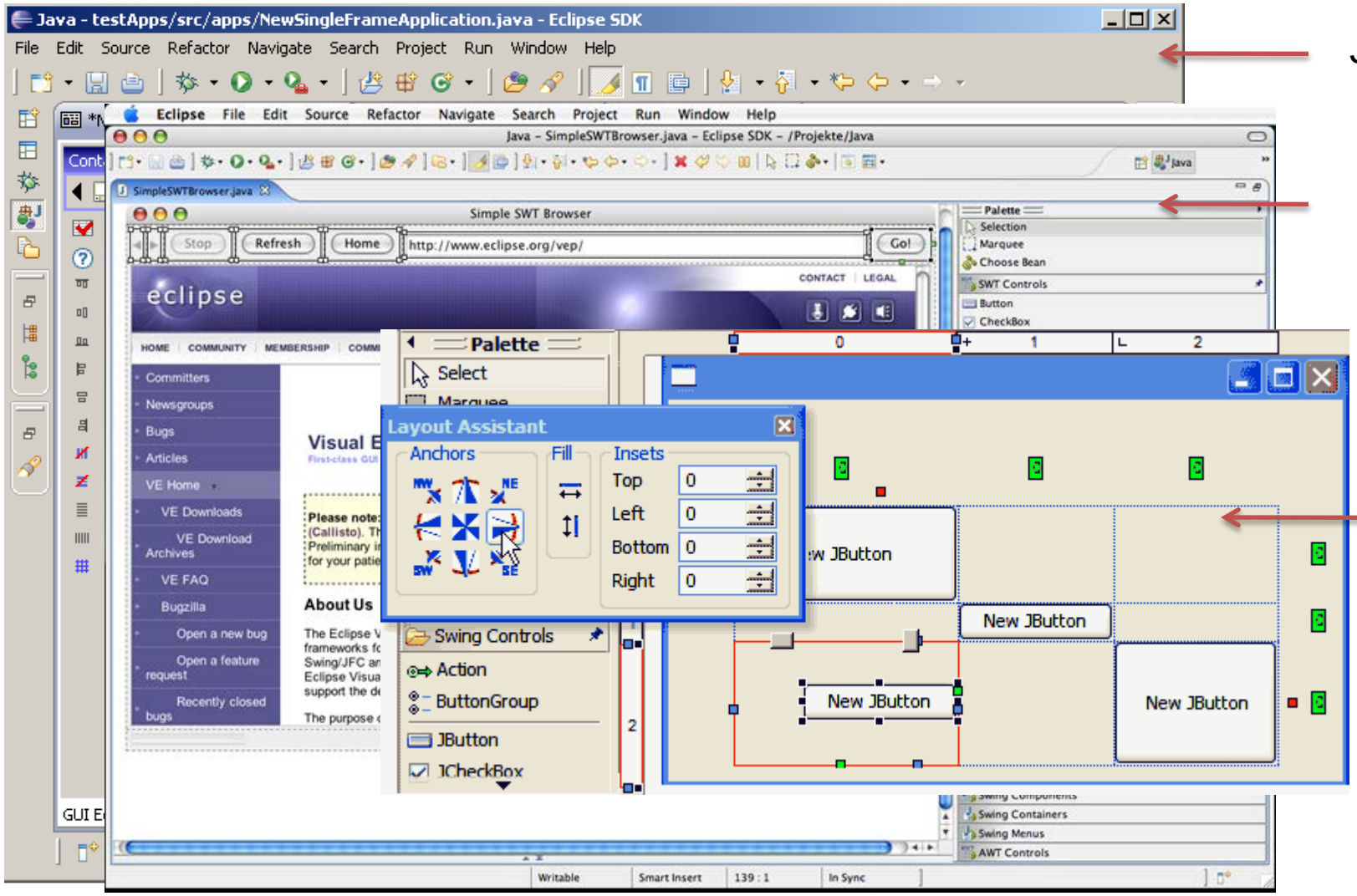


Quelle: http://media.konigi.com/notebook/iphone-mockup.jpg

Entwicklung von grafischen Oberflächen – GUI Builder I

- **Problem bei grafischen Oberflächen: Aufwendig in der Entwicklung**
- **Abhilfe sollen GUI-Builder schaffen (WYSIWYG Prinzip)**
- **Kommerzielle Tools (SWT/Swing):**
  - Jigloo
  - …
- **Freie Tools (SWT/Swing):**
  - Window Builder Pro
  - Visual Editor (Eclipse 3.2, veraltet, wird nicht mehr gepflegt)
- **Generieren (meist hässlichen) SWT/Swing Code**

Entwicklung von grafischen Oberflächen – GUI Builder II



Jigloo

Visual Editor

Window Builder Pro

Entwicklung von grafischen Oberflächen – Swing

- **Swing ist seit Java 1.2 Bestandteil der Java Runtime**
- **Baut auf dem Abstract Window Toolkit (AWT) auf**
- **Swing Komponenten werden direkt von Java gerendert**
  - Funktioniert auf allen Plattformen
  - Sieht überall gleich aus
  - NICHT nativ
- **Verschiedene Look&Feels**
  - Windows
  - Linux
  - Mac
  - …

Entwicklung von grafischen Oberflächen – SWT

- **Standard Widget Toolkit (SWT) http://www.eclipse.org/swt**
- **Wurde 2001 von IBM für Eclipse entwickelt**
- **NICHT Bestandteil der Java Runtime**
  - Bibliothek (inkl. nativen Bestandteilen) müssen mit ausgeliefert werden
- **Abstrahiert von nativer grafischer Benutzeroberfläche**
  - Einmal coden, überall nativ laufen lassen (theoretisch)
- **(Unter Windows) deutlich schneller als Swing**
- **Im Gegensatz zu Swing „schwergewichtig“, wegen der Verwendung von nativen Komponenten (statt sie selbst zu zeichnen)**

SWT – Tutorials und hilfreiche Links

- **SWT**
  - http://www.eclipse.org/articles/article.php?file=Article-Understanding-Layouts/index.html
  - http://www.eclipse.org/swt/widgets/
  - http://zetcode.com/tutorials/javaswttutorial/
  - http://www.vogella.de/articles/SWT/article.html

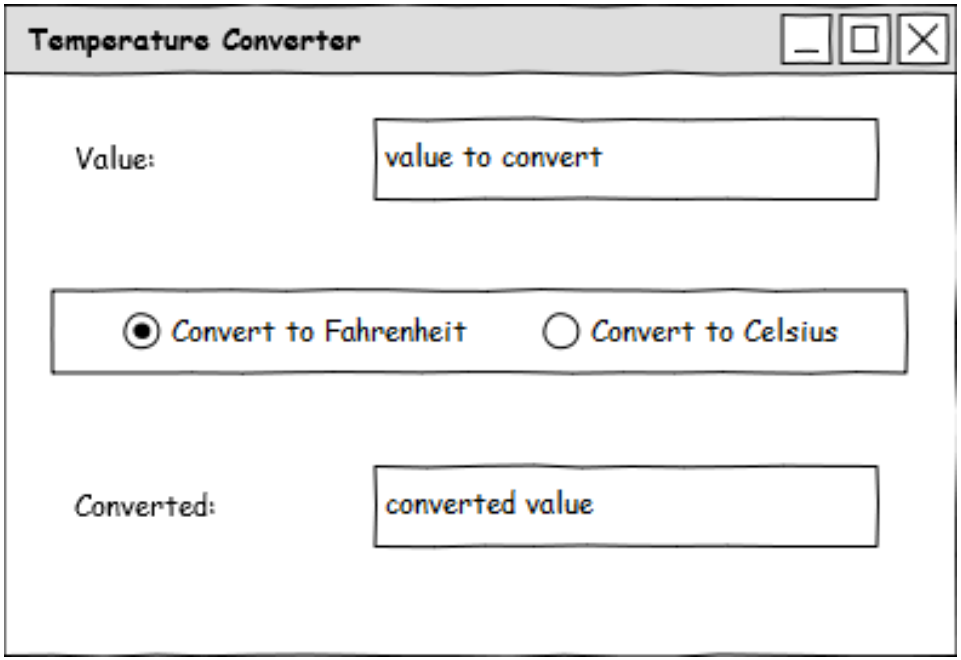
- **WindowBuilder Pro**
  - **Update Site:** http://dl.google.com/eclipse/inst/d2wbpro/latest/4.2

    bzw. http://dl.google.com/eclipse/inst/d2wbpro/latest/3.8

    bzw. http://dl.google.com/eclipse/inst/d2wbpro/latest/3.7
  - http://code.google.com/javadevtools/wbpro/index.html

Entwicklung von grafischen Oberflächen - Demo

- **Demo: Celsius – Fahrenheit Converter**
- **Umrechnungsformel:**
  - Celsius in Fahrenheit = ((TCelsius × 9) / 5) + 32
  - Fahrenheit in Celsius = (TFahrenheit - 32) × 5 / 9
- **Mock-Up**



Praktische Übung

- Entwicklung des Diaballik Login Screens
- Mock-Up:

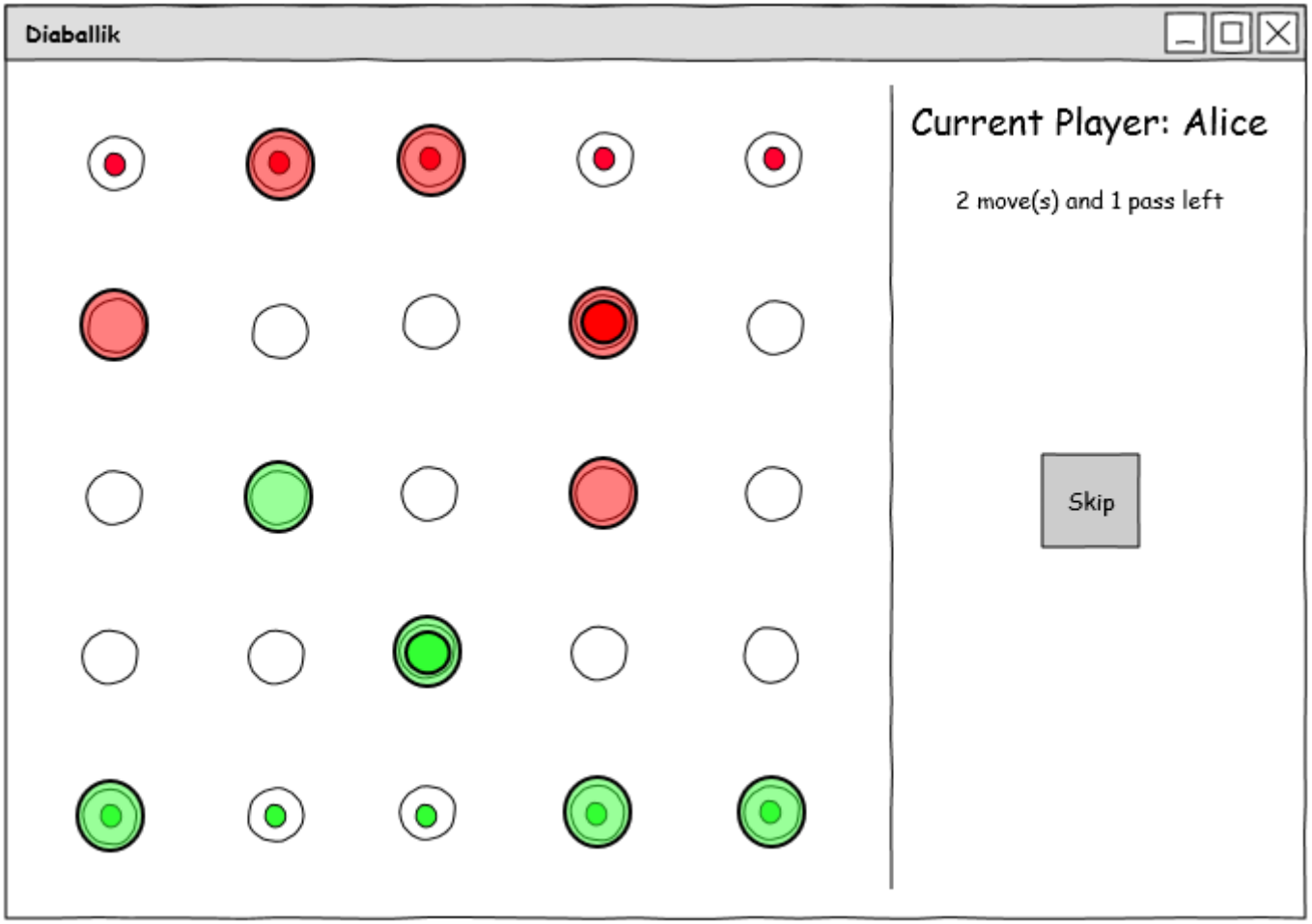
- Zeigt das Ergebnis einem Betreuer

Vorschau HA6 I

- **Deadline: 13.12.2011, 23:59 Uhr**
- **Aufgabe1: Diaballik Login Screen**

Vorschau HA6 I

- **Aufgabe2: Diaballik Game Screen**



Ende

Jetzt: Betreutes Arbeiten

Ansonsten: Schönes WE!