

## 1. Einleitung

## 2. Vorgehen

### 2.1 Allgemeines

### 2.2 Team

### 2.3 CRaPI



## 3. Entwurfsentscheidung

### 3.1 Environment

### 3.2 Code

## 4. Team Configurator

### 4.1 Übersicht



## 4.2 Funktionsumfang

## 5. Aktueller Stand

### 5.1 Übersicht

### 5.2 Personalitäten

## 6. Kritische Reflektion



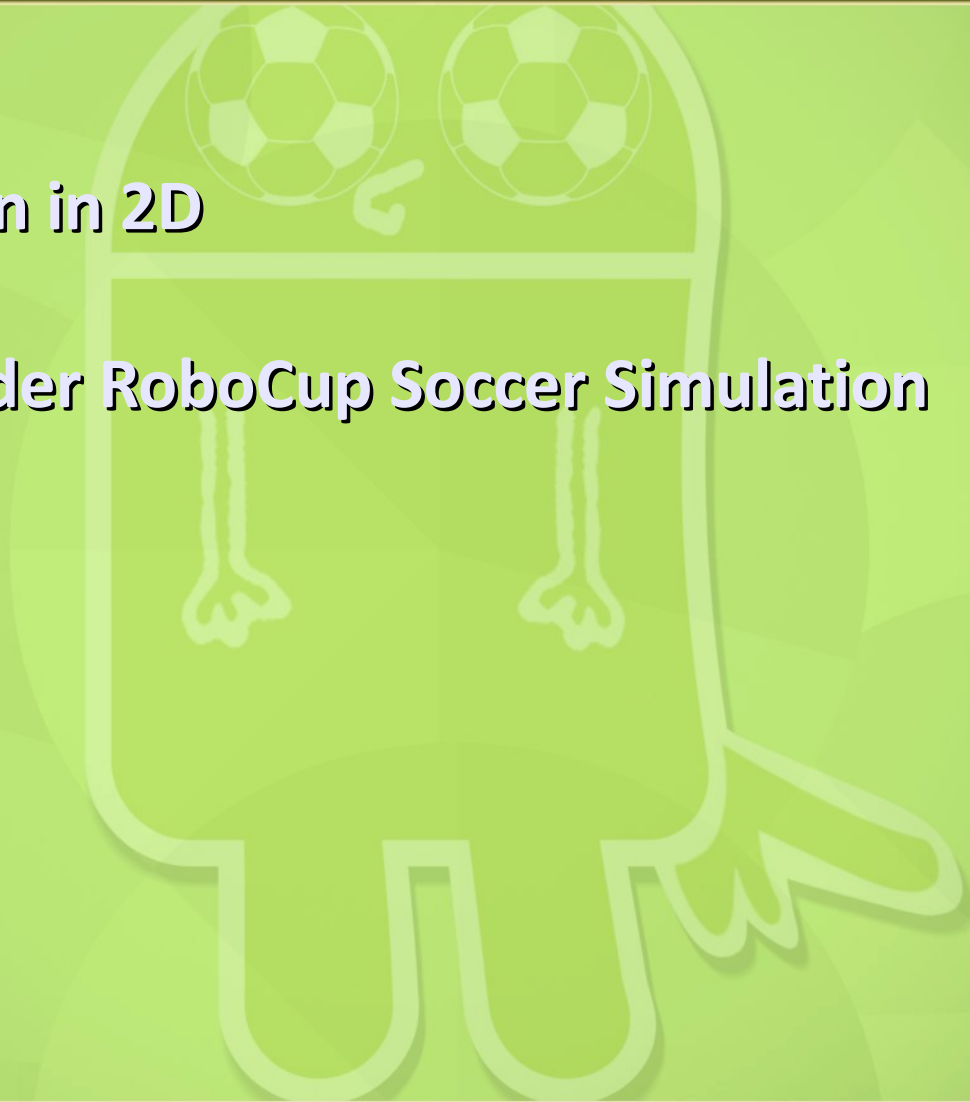
7. Aussichten

8. Quellen



RoboCup Soccer Simulation in 2D

Algorithmus für ein Team der RoboCup Soccer Simulation





## Agile Vorgehensweise

- Fokus auf Funktionalität
- Viel Kommunikation, wenig Werkzeuge und geplante Prozesse

Keine starre Aufgabenaufteilung

Pairprogramming

Dienstags Praktikum, Spieltests

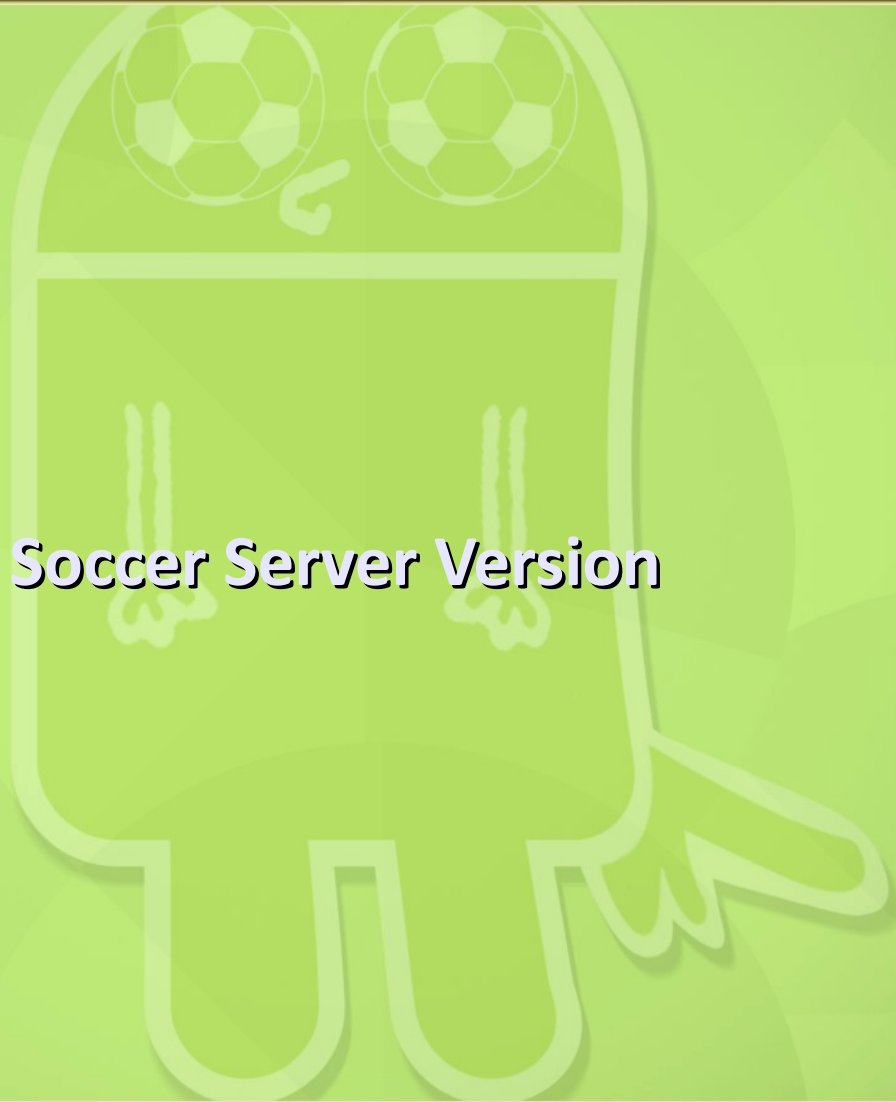
Donnerstags Teammeeting, Pairprogramming



Klassendiagramm erstellt

Funktionsweise analysiert

Anpassungen an die aktuelle Soccer Server Version





Sprache C#

CRaPI Framework (Correct robust and perfect api for simulated robocup)

Fedora 23 auf VirtualBox als Soccerserver und SoccerMonitor

Entwicklungsumgebungen Visual Studio 2012 und MonoDevelop 5 genutzt

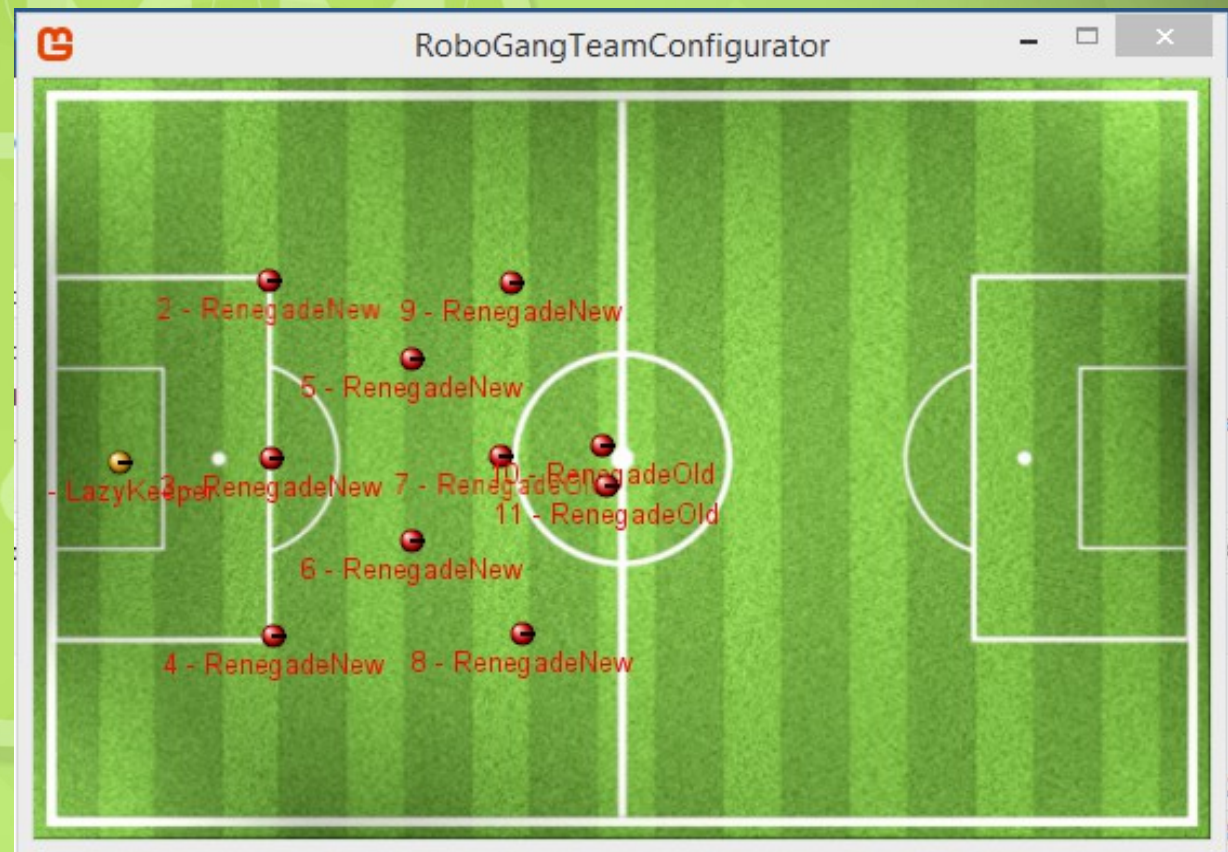
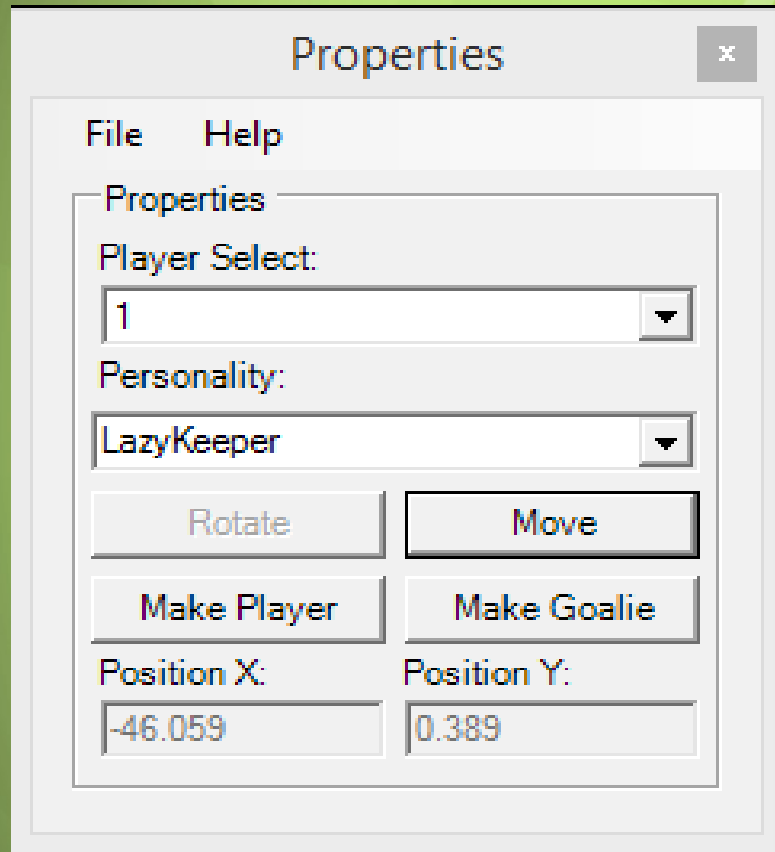
Plattformübergreifende Ausführbarkeit angestrebt (GNU/Linux, Windows)

Team Configurator für die Aufstellung

Position in Personalität aufteilen

Agierendes Verhalten abhängig vom aktuellen Spielmodus

Modularer Aufbau: Team → Player → Personality → PlayModeActions



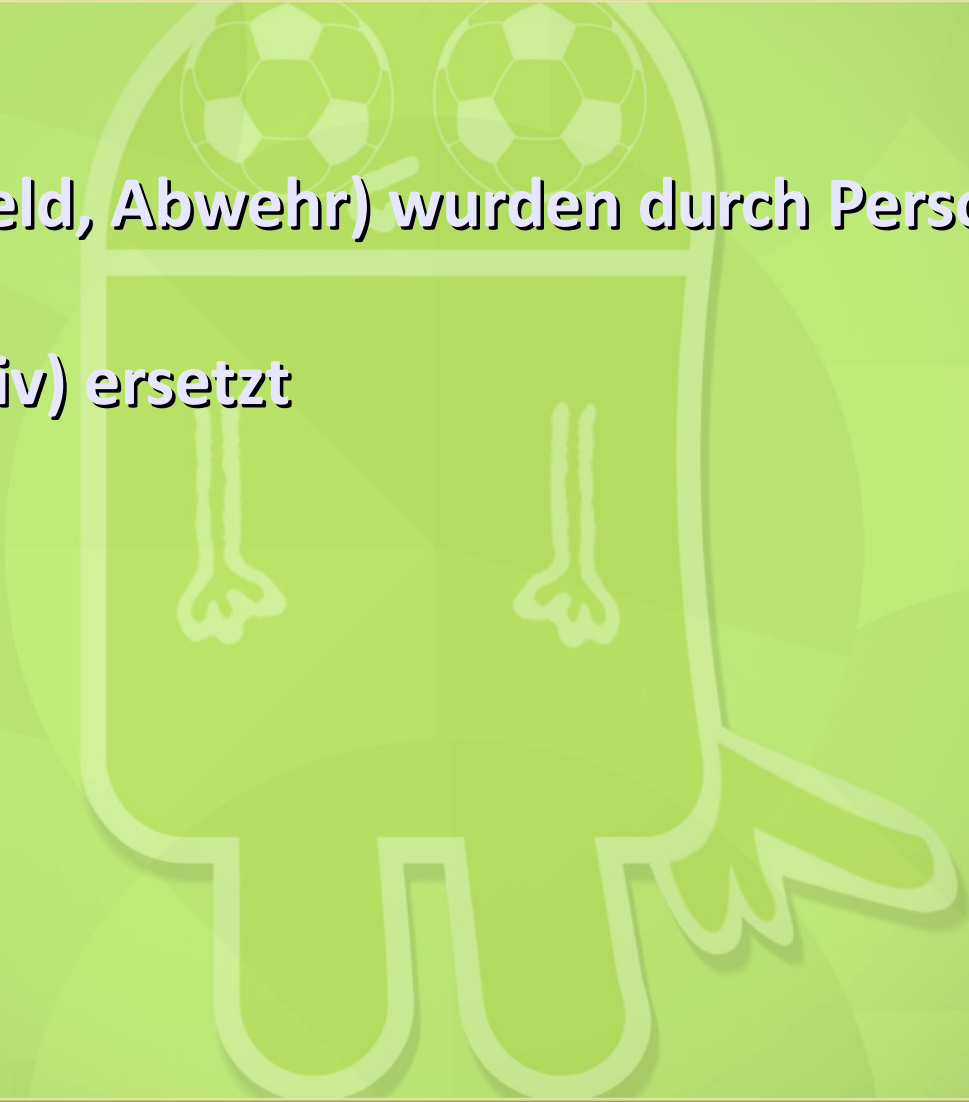
Visuelle Platzierung der Spieler

Verschiedene Personalitäten auswählbar

Binäres Auswählen des Goalie-Attributs



Positionen (Sturm, Mittelfeld, Abwehr) wurden durch Personalitäten  
(Offensiv, Defensiv) ersetzt





Torwart

Defensive

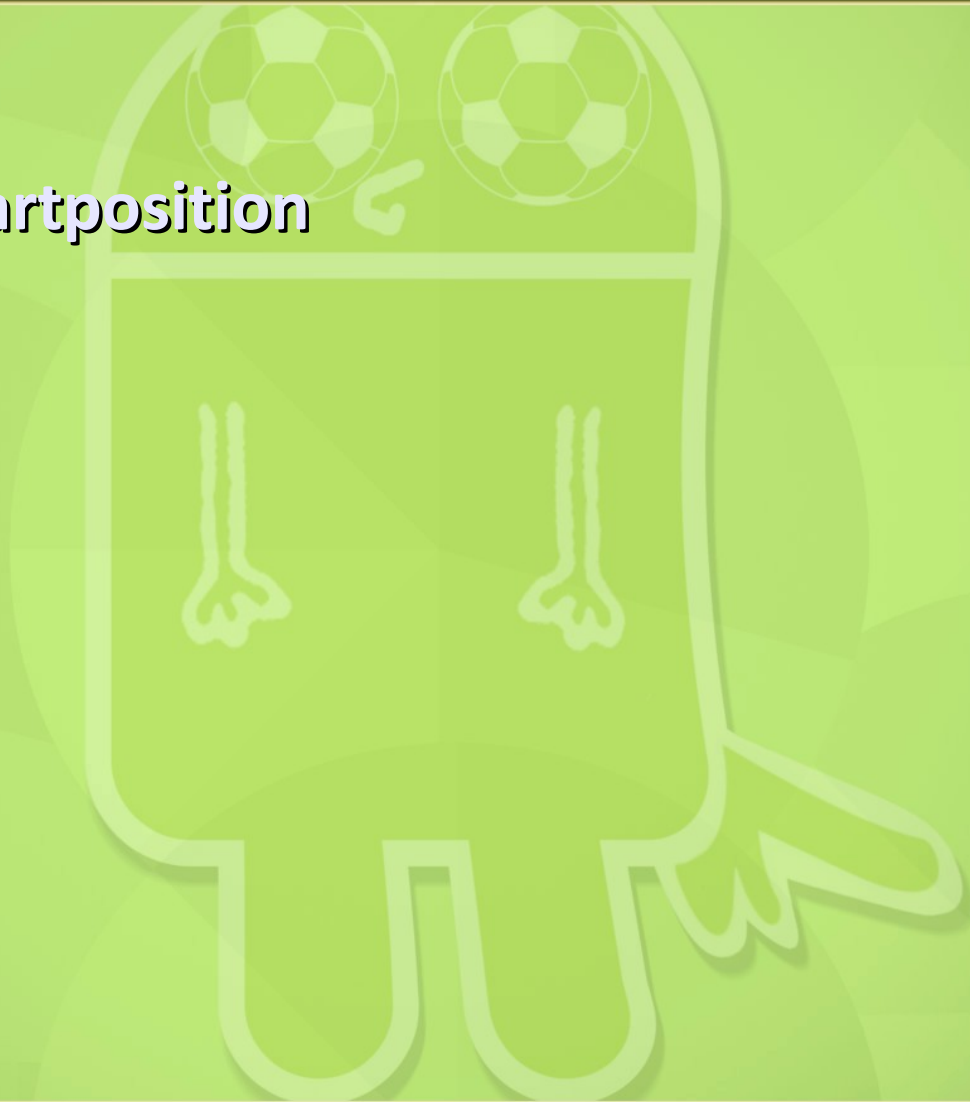
Offensive



Orientiert sich an seine Startposition

Agiert nur im Strafraum

Blockiert Torschuss

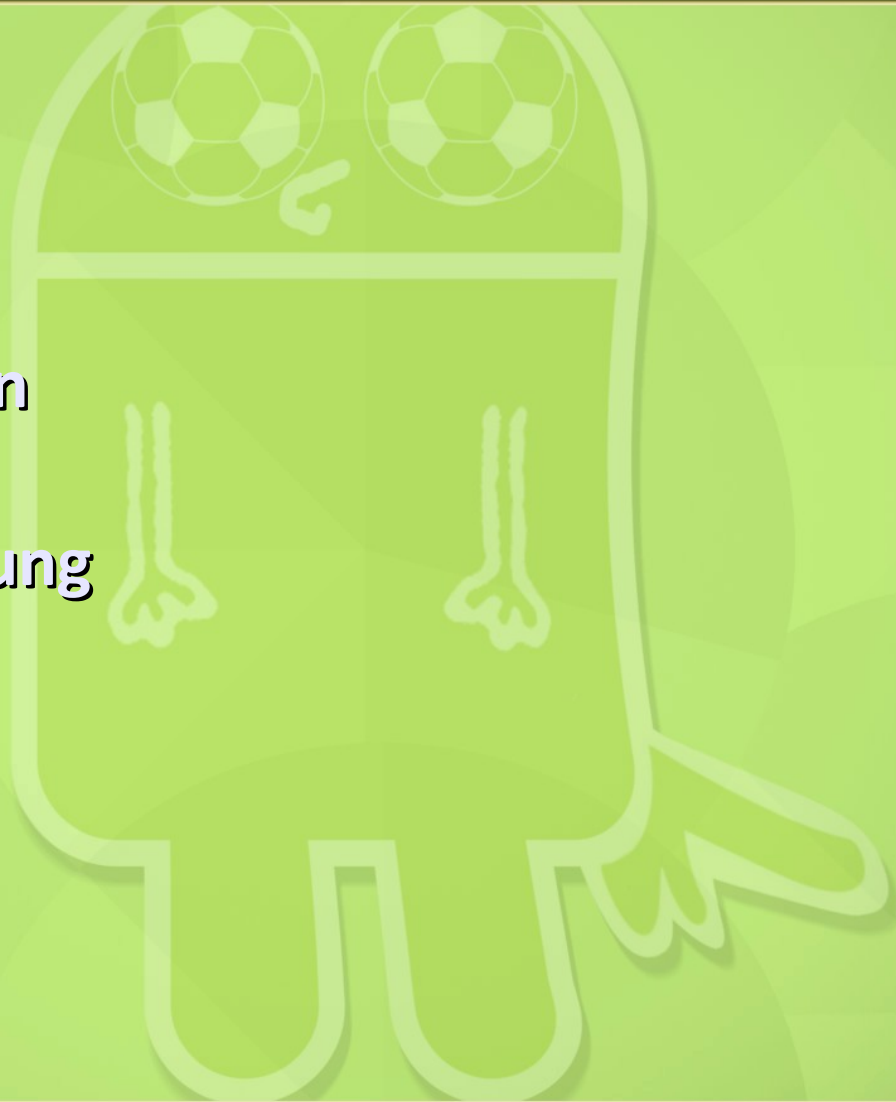


Fokus auf Passspiel

Abwehrorientiertes Verhalten

Unterbindung von Rudelbildung

Manndeckung



Aggressives Zweikampfverhalten

Ballorientiertes Spiel

Sucht den Abschluss

Weniger ausgeprägtes Teamverhalten



Ansatz international nicht konkurrenzfähig

Teamspiel schwer ohne AI realisierbar

Vorhersage der Spielsituation nicht ideal (Vorlegen)

Verlassen auf erfahrungsgemäße MagicNumbers



'Say and Hear' Funktion des Soccer Server nutzen

Optimalere Raumnutzung

Fokus auf Kommunikation legen

Framework CraPI mit Coach Funktion ausstatten

On-The-Fly Personalities austauschen

MagicNumbers in Datenbank erfassen

Testläufe automatisieren

Optimalere Werte automatisiert ermitteln

Optimierung der Nutzung von Blickfeld und -richtung

Crapi:

<http://crapi.sourceforge.net/>

