



ROBOSOCCER - ROAD TO FRANCE...

LEGOAL

Software-Entwicklungspraktikum (SEP)
Sommersemester 2016

Testprotokolle

Auftraggeber
Technische Universität Braunschweig
Institut für Programmierung und Reaktive Systeme
Prof. Dr. Ursula Goltz
Mühlenpfordtstrasse 23
38106 Braunschweig

Betreuer: Sascha Lity, Ken Rieke

Auftragnehmer:

| Name | E-Mail-Adresse |
|------------------|---------------------------------|
| Sören Christmann | s.christmann@tu-braunschweig.de |
| Alexander Joost | a.joost@tu-braunschweig.de |
| Mohamad Karaki | mo.karaki@tu-braunschweig.de |
| Bengt Kensy | b.kensy@tu-braunschweig.de |
| Anna Lörke | a.loerke@tu-braunschweig.de |

Braunschweig, 6. Juli 2016

Bearbeiterübersicht

| Kapitel | Autoren |
|---------|------------------------------------------------------------------------------------|
| 1 | Alexander Joost |
| 1.1 | Alexander Joost |
| 1.2 | Sören Christmann Alexander Joost Mohamad Karaki Bengt Kensy Anna Lörke |
| 1.3 | Alexander Joost |
| 2 | Anna Lörke |
| 2.1 | Alexander Joost |
| 2.2 | Sören Christmann Mohamad Karaki Bengt Kensy Anna Lörke |
| 2.3 | Alexander Joost |
| 3 | Anna Lörke |
| 3.1 | Alexander Joost |
| 3.2 | Alexander Joost Bengt Kensy Anna Lörke |
| 3.3 | Anna Lörke |

Inhaltsverzeichnis

| | |
|----------------------------------------|-----------|
| 1 Testdurchführung (2016-07-04) | 4 |
| 1.1 Testumgebung | 4 |
| 1.2 Testprotokoll | 5 |
| 1.3 Zusammenfassung | 48 |
| 2 Testdurchführung (2016-07-05) | 49 |
| 2.1 Testumgebung | 49 |
| 2.2 Testprotokoll | 50 |
| 2.3 Zusammenfassung | 56 |
| 3 Testdurchführung (2016-07-06) | 57 |
| 3.1 Testumgebung | 57 |
| 3.2 Testprotokoll | 58 |
| 3.3 Zusammenfassung | 69 |

1 Testdurchführung (2016-07-04)

Der erste Testdurchlauf beinhaltet sämtliche Unit-Tests aus dem fünften Kapitel der Testspezifikation.

Art des Tests: Unit-Test

Ausgeführte Testfälle: **T2001, T2002, T2003, T2004, T2005, T2006, T2007, T2008, T2009, T2010, T2011, T2012, T2013, T3001, T4001, T4002, T6001, T8001, T8002, T9001, T9002, T9003, T9004, T9005, T9006, T9007, T9008, T9009, T9010, T9011, T9012, T9013, T9014**

Beteiligte Tester: Sören Christmann, Bengt Kensy, Mohamad Karaki, Anna Lörke, Alexander Joost

Abgedeckte Funktionen: **F10, F20, F30, F40, F50, F60, F70, F80, F90, F100, F110, F130**

1.1 Testumgebung

Sämtliche Tests wurden im Legolabor des Informatikzentrums der Technischen Universität Braunschweig durchgeführt. Dabei wurden die vom Auftraggeber zur Verfügung gestellten Lego Mindstorms EV3 Roboter mit leJOS 0.9.1 und den dazugehörigen Raspberry Pis verwendet. Die verwendeten Computer liefen entweder mit Windows 7/10 oder mit Mac OS X 11. Als IDE wurde in beiden Fällen Eclipse 4.6 verwendet zusammen mit Java 8. Zusätzlich muss die Stabilität des Netzwerkes des Legolabors gewährleistet sein.

1.2 Testprotokoll

Im folgenden Abschnitt werden die durchzuführenden Testfälle protokolliert und die Ergebnisse näher erläutert.

| | |
|------------------------------------|-----------------------------------------------------------------------|
| Testfall | T2001 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Tores des eigenen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Toranzahl des eigenen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Toranzahl des eigenen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------|
| Testfall | T2002 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Tores des gegnerischen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Toranzahl des gegnerischen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Toranzahl des gegnerischen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------|
| Testfall | T2003 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Tores des eigenen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Toranzahl des eigenen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Toranzahl des eigenen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------|
| Testfall | T2004 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Tores des gegnerischen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Toranzahl des gegnerischen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Toranzahl des gegnerischen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------|
| Testfall | T2005 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Fouls des eigenen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Foulanzahl des eigenen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Foulanzahl des eigenen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------|
| Testfall | T2006 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Fouls des gegnerischen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Foulanzahl des gegnerischen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Foulanzahl des gegnerischen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------|
| Testfall | T2007 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Fouls des eigenen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Foulanzahl des eigenen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Foulanzahl des eigenen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------|
| Testfall | T2008 |
| Tester | Sören Christmann |
| Eingaben | Es wird der Button zum Zählen eines Fouls des gegnerischen Teams gedrückt. |
| Soll - Reaktion | Die Anzeige soll die Foulanzahl des gegnerischen Teams um 1 erhöhen. |
| Ist - Reaktion | Die Anzeige erhöht die Foulanzahl des gegnerischen Teams um 1. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-------------------------------------------------------|
| Testfall | T2009 |
| Tester | Bengt Kensy |
| Eingaben | Es wird der Button zum Starten des Timers angeklickt. |
| Soll - Reaktion | Der Timer sollte starten. |
| Ist - Reaktion | Der Timer startet. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Testfall | T2010 |
| Tester | Mohamad Karaki |
| Eingaben | Es wird auf den Button mit der Aufschrift „Homepage “ geklickt. Dabei ist die GUI derzeit im Modus des manuellen Spiels. |
| Soll - Reaktion | Die GUI soll die Startseite anzeigen. |
| Ist - Reaktion | Die GUI zeigt die Startseite an. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Testfall | T2011 |
| Tester | Mohamad Karaki |
| Eingaben | Es wird auf den Button mit der Aufschrift „Homepage “ geklickt. Dabei ist die GUI derzeit im Modus des manuellen Spiels. |
| Soll - Reaktion | Die GUI soll die Startseite anzeigen. |
| Ist - Reaktion | Die GUI zeigt die Startseite an. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T2012 |
| Tester | Mohamad Karaki |
| Eingaben | Als erstes wird der Button zum Spielstart angeklickt. Danach wird der Button zum Pausieren des Spiels gedrückt. Nun wird der Button zum Fortfahren des Spiels betätigt. Schlussendlich wird noch einmal der Button zum Pausieren des Spiels angeklickt und danach der Button zum Zurückfahren der Roboter auf die Startpositionen. |
| Soll - Reaktion | Zuerst sollten die Roboter das Spiel aktiv ausführen. Nach dem ersten Klick des Pausieren-Buttons sollten sie stehenbleiben. Nach dem Klick des Fortfahren-Buttons sollten sie weiterspielen. Ist der Pausieren-Button erneut angeklickt worden, so sollten die Roboter wieder stehen bleiben und das Spiel pausieren, bis auf der Zurückfahren-Button angeklickt wurde, wodurch die Roboter zu ihren Startpositionen zurückkehren sollten. |
| Ist - Reaktion | Die Roboter starten das Spiel, bleiben stehen, fahren weiter, bleiben stehen und kehren zu ihren Startpositionen zurück. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T2013 |
| Tester | Mohamad Karaki |
| Eingaben | Als erstes wird der Button zum Spielstart angeklickt. Danach wird der Button zum Pausieren des Spiels gedrückt. Nun wird der Button zum Fortfahren des Spiels betätigt. Schlussendlich wird noch einmal der Button zum Pausieren des Spiels angeklickt und danach der Button zum Zurückfahren der Roboter auf die Startpositionen. |
| Soll - Reaktion | Zuerst sollten die Roboter das Spiel aktiv ausführen. Nach dem ersten Klick des Pausieren-Buttons sollten sie stehenbleiben. Nach dem Klick des Fortfahren-Buttons sollten sie weiterspielen. Ist der Pausieren-Button erneut angeklickt worden, so sollten die Roboter wieder stehen bleiben und das Spiel pausieren, bis der Zurückfahren-Button angeklickt wurde, wodurch die Roboter zu ihren Startpositionen zurückkehren sollten. |
| Ist - Reaktion | Die Roboter starten das Spiel, bleiben stehen, fahren weiter, bleiben stehen und kehren zu ihren Startpositionen zurück. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T3001 |
| Tester | Sören Christmann |
| Eingaben | Die Roboter und der Ball werden in unterschiedlichen Positionen auf das Spielfeld gestellt. Danach wird die Methode manuell aufgerufen. |
| Soll - Reaktion | Innerhalb der GUI soll das Spielfeld mit ungefähren Positionen der Roboter und des Balls angezeigt werden. |
| Ist - Reaktion | Spielfeld wird mit ungefähren Positionen der Roboter und des Balls angezeigt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T4001 |
| Tester | Mohamad Karaki |
| Eingaben | Die Methode <code>receivePositionData()</code> wird manuell in der Klasse <code>CameraReceiver</code> aufgerufen. |
| Soll - Reaktion | Im Terminal sollten der übergebene String des Raspberry Pis der Kamera angezeigt werden. |
| Ist - Reaktion | Im Terminal wird der übergebene String des Raspberry Pis der Kamera angezeigt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | Bei unvorteilhaften Lichtverhältnissen können keine korrekten Positionsdaten berechnet werden, sodass dieser Test nur in der Ideal-situation fehlerfrei ausgeführt werden kann. |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T4002 |
| Tester | Alexander Joost |
| Eingaben | Die Methode fillCoordinate(String positionData) wird manuell aufgerufen mit originalen Kameradaten. |
| Soll - Reaktion | Die Variablen message, xBall und yBall und die Arrays xCoordinates, yCoordinates und roboRotation werden mit den entsprechenden Werten des positionData-Strings gefüllt. |
| Ist - Reaktion | Die Variablen und Arrays werden korrekt gefüllt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Es wird die Taste „q“ gedrückt, um das Drehen auf der Stelle auszuwählen. Nun wird zuerst die linke Pfeiltaste, dann die rechte Pfeiltaste und dann die Taste nach oben und danach nach unten gedrückt. Zwischen den Ausführungen der Pfeiltasten wird die Reaktion des Roboters abgewartet. |
| Soll - Reaktion | Der Roboter soll sich nach dem ersten Tastendruck, bis eine andere Taste gedrückt wird, auf der Stelle links herum drehen. Danach wird das gleiche Szenario getestet mit rechts herum drehen und anschließend lässt man den Roboter vorwärts und rückwärts fahren. Aufgrund der Bauweise ist bei letzterem eine „hopplige“ Fahrweise zu erwarten. |
| Ist - Reaktion | Der Roboter hat die Soll-Reaktion gezeigt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Es wird die Taste „w“ gedrückt, um das Drehen mit einem vorwärts drehenden Rad auszuwählen. Zuerst wird die linke Pfeiltaste, dann die rechte Pfeiltaste und dann die Taste nach oben und danach nach unten gedrückt. Zwischen den Ausführungen der Pfeiltasten wird die Reaktion des Roboters abgewartet. |
| Soll - Reaktion | Der Roboter soll sich nach dem ersten Tastendruck, bis eine andere Taste gedrückt wird, links herum drehen und dabei nur das rechte, vorwärts drehende Rad verwenden. Danach wird das gleich getestet mit rechts herum drehen (linkes, vorwärts drehendes Rad) und anschließend geradeaus und rückwärts. Bei Rückwärts fahren ist ein leichtes „Hoppeln“ des Roboters zu erwarten, was in der Bauweise begründet liegt. |
| Ist - Reaktion | Der Roboter hat die Soll-Reaktion gezeigt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Es wird die Taste „e“ gedrückt, um das Drehen mit einem rückwärts drehenden Rad auszuwählen. Zuerst wird die linke Pfeiltaste, dann die rechte Pfeiltaste und dann die Taste nach oben und danach nach unten gedrückt. Zwischen den Ausführungen der Pfeiltasten wird die Reaktion des Roboters abgewartet. |
| Soll - Reaktion | Der Roboter soll sich nach dem ersten Tastendruck, bis eine andere Taste gedrückt wird, links herum drehen und dabei nur das linke, rückwärts drehende Rad verwenden. Danach wird das gleich getestet mit rechts herum drehen (rechts, rückwärts drehendes Rad) und anschließend geradeaus und rückwärts. Bei Rückwärts fahren ist ein leichtes „Hoppeln“ des Roboters zu erwarten, was in der Bauweise begründet liegt. |
| Ist - Reaktion | Der Roboter hat die Soll-Reaktion gezeigt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Der Roboter wird mit Ball auf dem Spielfeld positioniert. Dann wird zuerst die Taste „1“ und dann die Leertaste gedrückt. |
| Soll - Reaktion | Der Ball wird leicht geschossen. |
| Ist - Reaktion | Der Ball bewegt sich etwa 10 cm über das Spielfeld. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Der Roboter wird mit Ball auf dem Spielfeld positioniert. Dann wird zuerst die Taste „2“ und dann die Leertaste gedrückt. |
| Soll - Reaktion | Der Ball wird mittel stark geschossen. |
| Ist - Reaktion | Der Ball bewegt sich etwa 30 cm über das Spielfeld. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Der Roboter wird mit Ball auf dem Spielfeld positioniert. Dann wird zuerst die Taste „3“ und dann die Leertaste gedrückt. |
| Soll - Reaktion | Der Ball wird stark geschossen. |
| Ist - Reaktion | Der Ball bewegt sich etwa 50 cm über das Spielfeld. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Es wird die Taste „B“ gedrückt. |
| Soll - Reaktion | Der Roboter macht einen Ton. |
| Ist - Reaktion | Der Ton ist zu hören. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Es werden nacheinander die folgend benannten Tasten gedrückt. Bevor die nächste Taste betätigt wird, wird die Reaktion abgewartet. Tastenreihenfolge ist „A“, „S“, „D“, „F“, „G“, „H“, „J“, „K“. |
| Soll - Reaktion | Der Roboter dreht sich erst 20 Grad nach links, dann 20 Grad nach rechts. Danach dreht er sich 45 Grad nach links und anschließend nach rechts. Folgend rotiert der Roboter zuerst 90 Grad linksherum und anschließend rechtsherum. Zuletzt soll sich der Roboter um 180 Grad nach links und fernem genauso weit rechts herum zurück drehen. |
| Ist - Reaktion | der Roboter führt die beschriebenen Drehungen durch. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T6001 |
| Tester | Mohamad Karaki |
| Eingaben | Die Pfeiltaste nach oben wird gedrückt gehalten. Zwischendurch wird mehrfach die Taste „M“ betätigt und abgewartet, ob der Roboter seine Geschwindigkeit verändert. Danach wird während der Fahrt mehrfach die Taste „N“ gedrückt und der Roboter wieder beobachtet. |
| Soll - Reaktion | Der Roboter beschleunigt während die Taste „M“ gedrückt wird. Wird „N“ gedrückt, verringert er die Geschwindigkeit. |
| Ist - Reaktion | Der Roboter verändert seine Geschwindigkeit wie beschrieben. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T8001 |
| Tester | Mohamad Karaki |
| Eingaben | Als erstes wird der Ball weit vom Roboter entfernt gelegt. Die Methode detectDistance(int robotID, int ballX, int ball) wird manuell aufgerufen und die Rückgabe wird auf Korrektheit überprüft. Ebenfalls wird die Methode isBallFree() manuell aufgerufen und die Rückgabe auf Korrektheit überprüft. Danach wird der Ball direkt vor den Roboter in den Arm gelegt. Die Methode isBallPossessed(int robotID) wird manuell aufgerufen und auf Korrektheit überprüft. |
| Soll - Reaktion | Der erste Methodenaufruf soll den Abstand zwischen Ball und Roboter wiedergeben. Der zweite Methodenaufruf sollte ein „true“ zurückgeben, sodass ersichtlich wird, dass sich der Ball derzeit in keinem Besitz befindet. Der dritte Methodenaufruf sollte ebenfalls ein „true“ zurückgeben, da der Roboter den Ball in Besitz hat. |
| Ist - Reaktion | Mit einer Abweichung von wenigen Zentimetern wurde die Distanz korrekt berechnet. Die beiden weiteren Methoden gaben ein „true“ zurück. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T8002 |
| Tester | Bengt Kensy |
| Eingaben | Die Roboter aus Team LeGoal werden mittig im eigenen Spielfeld platziert und sind im Ballbesitz. Die Roboter der Gegner werden in die Außenbereiche gestellt. Anschließend wird die Methode <code>communicateStrategies(int[] strategieID)</code> manuell aufgerufen. |
| Soll - Reaktion | Im ersten Feld des Arrays <code>strategieID</code> sollte eine 1 stehen. |
| Ist - Reaktion | Im ersten Feld des Arrays steht eine 1. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | Lichtverhältnisse im Raum müssen gut sein. |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T8002 |
| Tester | Bengt Kensy |
| Eingaben | Die Roboter des gegnerischen Teams werden in ihrer eigenen Hälfte platziert und ist im Ballbesitz. Die Roboter des Teams LeGoal werden ebenfalls in ihrer eigenen Hälfte aufgestellt. Anschließend wird die Methode <code>communicateStrategies(int[] strategieID)</code> manuell aufgerufen. |
| Soll - Reaktion | Im sechsten Feld des Arrays <code>strategieID</code> sollte eine 1 stehen. |
| Ist - Reaktion | Im sechsten Feld des Arrays steht eine 1. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | Lichtverhältnisse im Raum müssen gut sein. |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T8002 |
| Tester | Bengt Kensy |
| Eingaben | Ein Roboter des Teams LeGoal wird in am nächsten am Ball platziert als sämtliche andere Roboter. Anschließend wird die Methode <code>communicateStrategies(int[] strategieID)</code> manuell aufgerufen. |
| Soll - Reaktion | Im elften Feld des Arrays <code>strategieID</code> sollte eine 1 stehen. |
| Ist - Reaktion | Im elften Feld des Arrays steht eine 1. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | Lichtverhältnisse im Raum müssen gut sein. |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9001 |
| Tester | Bengt Kensy |
| Eingaben | Es wird mehrmals, manuell die Methode rotateRight(int degree) in der Klasse Motions aufgerufen. Die Reihenfolge lautet rotateRight(90), rotateRight(180), rotateRight(20). Durch entsprechende Markierungen am Boden, welche mit Hilfe eines Geodreiecks gesetzt wurden, kann der Test auf Erfolg überprüft werden. Abweichungen von ca. 5 Grad sind jedoch erlaubt. |
| Soll - Reaktion | Der Roboter dreht sich in drei Phasen zuerst 90 Grad, dann 180 Grad und zum Schluss 20 Grad im Uhrzeigersinn. |
| Ist - Reaktion | Entspricht mit erlaubten, leichten Abweichungen im Rahmen von unter 5 Grad der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9002 |
| Tester | Bengt Kensy |
| Eingaben | Es wird mehrmals, manuell die Methode rotateLeft(int degree) in der Klasse Motions aufgerufen. Die Reihenfolge lautet rotateLeft(90), rotateLeft(180), rotateLeft(20). Durch entsprechende Markierungen am Boden, welche mit Hilfe eines Geodreiecks gesetzt wurden, kann der Test auf Erfolg überprüft werden. Abweichungen von ca. 5 Grad sind jedoch erlaubt. |
| Soll - Reaktion | Der Roboter dreht sich in drei Phasen zuerst 90 Grad, dann 180 Grad und zum Schluss 20 Grad gegen den Uhrzeigersinn. |
| Ist - Reaktion | Entspricht mit erlaubten, leichten Abweichungen im Rahmen von unter 5 Grad der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9003 |
| Tester | Alexander Joost |
| Eingaben | Es wird mehrmals, manuell die Methode rotateRight(int degree) in der Klasse Motions aufgerufen. Die Reihenfolge lautet rotateRight(90), rotateRight(180), rotateRight(20). Durch entsprechende Markierungen am Boden, welche mit Hilfe eines Geodreiecks gesetzt wurden, kann der Test auf Erfolg überprüft werden. Abweichungen von ca. 5 Grad sind jedoch erlaubt. Zudem wird der Ball in den Arm des Roboters gelegt, damit überprüft werden kann, ob der Ball auch nach der Drehung noch in Besitz des Roboters ist. |
| Soll - Reaktion | Der Roboter dreht sich in drei Phasen zuerst 90 Grad, dann 180 Grad und zum Schluss 20 Grad im Uhrzeigersinn und behält dabei den Ball. |
| Ist - Reaktion | Entspricht mit erlaubten, leichten Abweichungen im Rahmen von unter 5 Grad der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9004 |
| Tester | Alexander Joost |
| Eingaben | Es wird mehrmals, manuell die Methode rotateLeft(int degree) in der Klasse Motions aufgerufen. Die Reihenfolge lautet rotateLeft(90), rotateLeft(180), rotateLeft(20). Durch entsprechende Markierungen am Boden, welche mit Hilfe eines Geodreiecks gesetzt wurden, kann der Test auf Erfolg überprüft werden. Abweichungen von ca. 5 Grad sind jedoch erlaubt. Zudem wird der Ball in den Arm des Roboters gelegt, damit überprüft werden kann, ob der Ball auch nach der Drehung noch in Besitz des Roboters ist. |
| Soll - Reaktion | Der Roboter dreht sich in drei Phasen zuerst 90 Grad, dann 180 Grad und zum Schluss 20 Grad gegen den Uhrzeigersinn und behält dabei den Ball. |
| Ist - Reaktion | Entspricht mit erlaubten, leichten Abweichungen im Rahmen von unter 5 Grad der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9005 |
| Tester | Alexander Joost |
| Eingaben | Es wird die Methode forward(int speed, int time) manuell aufgerufen: forward(500, 5000). |
| Soll - Reaktion | Der Roboter sollte fünf Sekunden lang geradeaus fahren. Die Fahrzeit sollte ohne Abweichungen eingehalten werden. Gewisse Abweichungen des Fahrweges nach links oder rechts sind erlaubt. |
| Ist - Reaktion | Der Roboter fährt zwar exakt fünf Sekunden, jedoch extrem schief. |
| Ergebnis | Der Testlauf war bedingt erfolgreich. Der Code funktioniert einwandfrei, jedoch gibt es noch hardwaretechnische Probleme |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | Nachdem Gewichte an die Roboter angebracht wurden, konnten diese nun eine deutlich geradere Fortbewegung vorweisen. |

| | |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9006 |
| Tester | Alexander Joost |
| Eingaben | Es wird die Methode forwardRotate(int speed, int degree) manuell aufgerufen: forwardRotate(500, 2000). Mithilfe von Messwerten kann die Genauigkeit der Methode bestimmt werden. |
| Soll - Reaktion | Der Roboter sollte die gewählte Gradzahl abfahren. |
| Ist - Reaktion | Der Roboter fährt die gewählte Gradzahl ab. |
| Ergebnis | Der Testlauf war erfolgreich |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9007 |
| Tester | Alexander Joost |
| Eingaben | Es wird die Methode forwardWithBall(int speed, int time) manuell aufgerufen: forwardWithBall(500, 5000). Im Arm des Roboters liegt der Ball. |
| Soll - Reaktion | Der Roboter sollte fünf Sekunden lang geradeaus fahren und dabei Rücksicht auf den Ball nehmen. Die Fahrzeit sollte ohne Abweichungen eingehalten werden. Gewisse Abweichungen des Fahrweges nach links oder rechts sind erlaubt. |
| Ist - Reaktion | Der Roboter fährt exakt fünf Sekunden und behält den Ball. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9008 |
| Tester | Alexander Joost |
| Eingaben | Es wird die Methode forwardRotateWithBall(int speed, int degree) manuell aufgerufen: forwardRotateWithBall(500, 2000). Mithilfe von Messwerten kann die Genauigkeit der Methode bestimmt werden. Im Arm des Roboters liegt der Ball. |
| Soll - Reaktion | Der Roboter sollte die gewählte Gradzahl abfahren und dabei Rücksicht auf den Ball nehmen. |
| Ist - Reaktion | Der Roboter fährt die gewählte Gradzahl ab und behält den Ball. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------|
| Testfall | T9009 |
| Tester | Anna Lörke |
| Eingaben | Es wird manuell die Methode strongKick() ausgeführt. |
| Soll - Reaktion | Der Roboter sollte im Vergleich zu den anderen beiden möglichen Schüssen den stärksten ausführen. |
| Ist - Reaktion | Der Roboter führt den im Vergleich zu den anderen beiden möglichen Schüssen den stärksten aus. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9010 |
| Tester | Anna Lörke |
| Eingaben | Es wird manuell die Methode mediumKick() ausgeführt. |
| Soll - Reaktion | Der Roboter sollte einen schwächeren Schuss als den strongKick() durchführen, jedoch einen stärkeren, als den poorKick(). |
| Ist - Reaktion | Die Ist-Reaktion entspricht der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------|
| Testfall | T9011 |
| Tester | Anna Lörke |
| Eingaben | Es wird manuell die Methode poorKick() ausgeführt. |
| Soll - Reaktion | Der Roboter sollte im Vergleich zu den anderen beiden möglichen Schüssen den schwächsten ausführen. |
| Ist - Reaktion | Der Roboter führt im Vergleich zu den anderen beiden möglichen Schüssen den schwächsten aus. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|------------------------------------------------------------------|
| Testfall | T9012 |
| Tester | Anna Lörke |
| Eingaben | Während der Roboter fährt wird diese Methode manuell aufgerufen. |
| Soll - Reaktion | Der Roboter sollte sofort stehenbleiben. |
| Ist - Reaktion | Der Roboter bleibt sofort stehen. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9013 |
| Tester | Alexander Joost |
| Eingaben | Der Roboter wird in die untere linke Ecke mit Blick in die Ecke gestellt und wird von der Kamera mit korrekten Positionsdaten erfasst. Die Methode rotateTo(400, 270) wird aufgerufen. Danach wird die Methode rotateTo(0,0) aufgerufen. |
| Soll - Reaktion | Der Roboter sollte sich Richtung obere rechte Ecke drehen. Danach sollte er sich Richtung untere linke Ecke drehen. |
| Ist - Reaktion | Der Roboter dreht sich nicht zu den Koordinaten 400 und 270, jedoch danach korrekt zu 0 und 0. |
| Ergebnis | Der Test war aufgrund von falschen Berechnungen nicht erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | Die Berechnungen zur korrekten Drehung müssen überarbeitet werden. |

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T9014 |
| Tester | Anna Lörke |
| Eingaben | Der Roboter wird in die untere linke Ecke mit Blick in die Ecke gestellt und wird von der Kamera mit korrekten Positionsdaten erfasst. Die Methode rotateTo(400, 270) wird aufgerufen. Danach wird die Methode rotateTo(10,10) aufgerufen. Auf dem Spielfeld werden weitere Roboter so platziert, dass sie den direkten Weg versperren. |
| Soll - Reaktion | Der Roboter dreht sich in die Richtung des Punktes 400/270. Danach bewegt sich der Roboter in die Richtung, weicht den im Weg stehenden Robotern entsprechend aus. Hat der Roboter das Ziel erreicht, so dreht er wieder um und weicht erneut den im Weg stehenden Robotern aus. Sollte er am Ende den Punkt 10/10 mit leichten Abweichungen erreicht haben, ohne einen anderen Roboter zu berühren, so ist der Test als erfolgreich einzustufen. |
| Ist - Reaktion | Ist-Reaktion entspricht der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

1.3 Zusammenfassung

Es gab bei den durchgeführten Tests nur wenige Komplikationen. So waren viele Schwierigkeiten hardwarebedingt. Die angebrachten Gewichte an den Robotern sorgten für deutlich genauere Bewegungen. Ebenfalls muss auf die Beleuchtung im Raum stets geachtet werden, damit die Kameradatenerfassung fehlerfrei funktioniert. Bei den Methoden der Klasse AiMovement mussten noch einige Berechnungen überarbeitet werden, damit die Rotation fehlerfrei funktionieren kann.

Zusammengefasst lässt sich sagen, dass die Software bei verlässlicher Hardware gute Ergebnisse hervorbringt.

2 Testdurchführung (2016-07-05)

Der zweite Testdurchlauf beinhaltet sämtliche Integrationstests aus dem vierten Kapitel der Testspezifikation.

Art des Tests: Integrationstest

Ausgeführte Testfälle: **T001, T002, T003, T004, T005, T006**

Beteiligte Tester: Sören Christmann, Bengt Kensy, Mohamad Karaki, Anna Lörke, Alexander Joost

Abgedeckte Funktionen: **F10, F20, F30, F40, F50, F60, F70, F80, F90, F100, F110, F130**

2.1 Testumgebung

Sämtliche Tests wurden im Legolabor des Informatikzentrums der Technischen Universität Braunschweig durchgeführt. Dabei wurden die vom Auftraggeber zur Verfügung gestellten Lego Mindstorms EV3 Roboter mit leJOS 0.9.1 und den dazugehörigen Raspberry Pis verwendet. Die verwendeten Computer liefen entweder mit Windows 7/10 oder mit Mac OS X 11. Als IDE wurde in beiden Fällen Eclipse 4.6 verwendet zusammen mit Java 8. Zusätzlich muss die Stabilität des Netzwerkes des Legolabors gewährleistet sein.

2.2 Testprotokoll

Die folgenden Tabellen beschreiben, wie der Testfall ausgeführt wurde und welches Ergebnis er geliefert hat. Da es bei Korrektur von Softwarefehlern oder anderen Gegebenheiten notwendig ist, einen Test mehrfach durchzuführen (Testläufe), ist jede Testdurchführung zu dokumentieren. Daher ist diese Tabelle für **jeden Testlauf** zu erstellen und **fortlaufend zu nummerieren**.

| | |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T001 |
| Tester | Sören Christmann |
| Eingaben | Innerhalb der Klasse des Displays wird auf die Klasse Camera-Receiver bzw. die Methode receivePosition() manuell zugegriffen. Dadurch erhält die Anzeige die aktuellen Kameradaten in bereits ausgewerteter Form. |
| Soll - Reaktion | Die Variablen für die Koordinaten in der Klasse des Displays sollten mit aktuellen Daten gefüllt sein. |
| Ist - Reaktion | Die Variablen für die Koordinaten in der Klasse des Displays sind mit aktuellen Daten gefüllt. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T002 |
| Tester | Mohamad Karaki |
| Eingaben | Die in der Testspezifikation genannten Methoden werden manuell aufgerufen. Danach wird die Beispielnachricht „hallo“ von der GUI aus gesendet. |
| Soll - Reaktion | Die Raspberry Pis sollten allesamt die Nachricht empfangen. |
| Ist - Reaktion | Die Raspberry Pis empfangen allesamt die Nachricht. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T003 |
| Tester | Mohamad Karaki |
| Eingaben | Die in der Testspezifikation genannten Methoden werden manuell aufgerufen. Von jedem Raspberry Pi wird die Beispielnachricht „hallo “ an die GUI gesendet. |
| Soll - Reaktion | Die GUI sollte dreimal in Folge, von jedem Raspberry Pi genau einmal, die Beispielnachricht erhalten. |
| Ist - Reaktion | Die GUI erhält dreimal die Beispielnachricht. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T004 |
| Tester | Bengt Kensy |
| Eingaben | Die drei in der Testspezifikation genannten Methoden werden manuell aufgerufen. Dabei werden die berechneten Werte auf Korrektheit überprüft, damit sicher gestellt werden kann, dass die berechneten Daten der richtigen Strategie zugeordnet werden können. In diesem Testfall werden die echten Kameradaten der Kameradatenauswertung verwendet. Hierfür wird der Ball in den Arm eines Roboters gelegt. |
| Soll - Reaktion | Die Methode detectDistance(int robotID, int ballX, int ballY) sollte für den Roboter, welcher im Ballbesitz ist, eine minimale Distanz von wenigen Zentimetern herausgeben. Die Methode isBallPossessed(int robotID) sollte ein „true“ herausgeben. Die Methode isBallFree() sollte ein „false“ herausgeben. |
| Ist - Reaktion | Die Ist-Reaktion entspricht der Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T005 |
| Tester | Mohamad Karaki |
| Eingaben | Die in der Testspezifikation genannten Methoden werden manuell aufgerufen. Von einem Raspberry Pi wird die Beispielnachricht „hallo “ an einen anderen Raspberry Pi gesendet. |
| Soll - Reaktion | Der andere Raspberry Pi soll die erhält Beispielnachricht erhalten.. |
| Ist - Reaktion | Der andere Raspberry Pi erhält die Beispielnachricht. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T006 |
| Tester | Anna Lörke |
| Eingaben | Innerhalb der Klasse der Strategien wird die AiMovement-Methode moveTo mit den Koordinaten des Balls, welcher auf dem Spielfeld platziert wird, aufgerufen. |
| Soll - Reaktion | Der Roboter, welcher mit der Methode aufgerufen wurde, soll zum Ball fahren. |
| Ist - Reaktion | Der Roboter fährt zum Ball. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

2.3 Zusammenfassung

Es gab bei den durchgeführten Tests keine nennenswerten Komplikationen. Der Nachrichtenaustausch unter den verschiedenen Geräten hat in allen Tests reibungslos funktioniert. Allerdings muss während der Durchführung ein stabiles Netzwerk gewährleistet werden.

Zusammengefasst lässt sich sagen, dass die Software bei verlässlicher Hardware gute Ergebnisse hervorbringt.

3 Testdurchführung (2016-07-06)

Zuletzt erfolgt der Abnahmetest, der im folgenden protokolliert ist.

Art des Tests: Abnahmetest

Ausgeführte Testfälle: **T100, T200, T300, T400, T500, T600, T700, T800, T900, T1000, T1100**

Beteiligte Tester: Bengt Kensy, Anna Lörke, Alexander Joost

Abgedeckte Funktionen: **F10, F20, F30, F40, F50, F60, F70, F80, F90, F100, F110, F130**

3.1 Testumgebung

Sämtliche Tests wurden im Legolabor des Informatikzentrums der Technischen Universität Braunschweig durchgeführt. Dabei wurden die vom Auftraggeber zur Verfügung gestellten Lego Mindstorms EV3 Roboter mit leJOS 0.9.1 und den dazugehörigen Raspberry Pis verwendet. Die verwendeten Computer liefen entweder mit Windows 7/10 oder mit Mac OS X 11. Als IDE wurde in beiden Fällen Eclipse 4.6 verwendet zusammen mit Java 8. Zusätzlich muss die Stabilität des Netzwerkes des Legolabors gewährleistet sein.

3.2 Testprotokoll

Im folgenden Abschnitt werden die durchzuführenden Testfälle protokolliert und die Ergebnisse näher erläutert.

| | |
|------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T100 |
| Tester | Bengt Kensy |
| Eingaben | Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt. |
| Soll - Reaktion | Es wird keine Fehlermeldung angezeigt. |
| Ist - Reaktion | Es gibt keine Fehlermeldung. |
| Ergebnis | Der Testfall war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T200 |
| Tester | Alexander Joost |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. |
| Soll - Reaktion | Ein Roboter des eigenen Teams mit Vorrang beginnt zu fahren, nach fünf Sekunden folgen die restlichen Roboter. |
| Ist - Reaktion | Ein eigener Roboter beginnt das Spiel und die anderen folgen nach fünf Sekunden. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T300 |
| Tester | Alexander Joost |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. |
| Soll - Reaktion | Die Positionen der einzelnen Roboter auf dem Spielfeld passen annähernd mit denen in der GUI dargestellten überein. |
| Ist - Reaktion | Die Positionen sind leicht versetzt. |
| Ergebnis | Auch wenn die Ist- und Soll-Reaktion nicht übereinstimmen, wird der Testlauf als erfolgreich betrachtet, da die versetzten Positionen mit dem Zeitverzug der gelieferten Positionsdaten durch die Kamera zu erklären sind. Daran lässt sich durch das gelieferte Softwareprodukt nichts verändern, da die Kamera im Aufgabenbereich des Kunden liegt. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T400 |
| Tester | Anna Lörke |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. Der Benutzer wählt nun den „Pausieren“-Button und wartet auf eine Reaktion. Danach wird das Spiel mit dem Klick auf „Fortsetzen“ weitergeführt. Anschließend wird das Spiel mit der Auswahl des „Stopp“-Buttons angehalten und mit „Start mit Vorrang für das eigene Team“ weiter gespielt. Dieser Test sollte zu einem späteren Zeitpunkt mit allen sechs Robotern wiederholt werden. |
| Soll - Reaktion | Ein eigener Roboter beginnt mit fünf Sekunden Vorsprung zu allen anderen Robotern das Spiel. Wird das Spiel pausiert, bleiben alle Roboter an ihren Positionen stehen und bewegen sich nach der Auswahl von „Fortsetzen“ weiter. Beim Stopp bleiben die Roboter wieder an ihren Positionen stehen und fahren für den Start mit Vorrang für das eigene Team zu ihren Startpositionen zurück. Ein Roboter des eigenen Teams beginnt. In der GUI sollte stets klar zu erkennen sein, welche Aktion gerade gewählt wurde. |
| Ist - Reaktion | Die Ist- Reaktion entspricht der Soll-Reaktion. |
| Ergebnis | Der Testlauf ist vorerst erfolgreich, sollte aber trotzdem mit allen Robotern wiederholt werden. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T500 |
| Tester | Bengt Kensy |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. |
| Soll - Reaktion | Der Roboter, der dem Ball am nächsten ist, fährt in den fünf Sekunden Vorrang am Anfang zum Ball. Außerdem fährt im Spielverlauf, wenn der Ball frei ist, der jeweils nächste eigene Roboter zum Ball. |
| Ist - Reaktion | Sowohl zu Spielbeginn, als auch im Spielverlauf, fährt der jeweils nächste Roboter zum Ball. |
| Ergebnis | Der Testlauf ist erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T600 |
| Tester | Anna Lörke |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. In einem späteren Testlauf, wenn beide Teams mit der Programmierung fertig sind, sollte der Test mit allen Robotern im Spiel wiederholt werden. |
| Soll - Reaktion | Die Roboter bewegen sich im Spielverlauf jeweils mit Ball über das Feld. Sie können vorwärts und Kurven fahren ohne den Ball zu verlieren sowie bremsen. |
| Ist - Reaktion | Die Roboter zeigen alle Fahrmanöver ohne den Ball zu verlieren. |
| Ergebnis | Der Testlauf ist vorerst erfolgreich, sollte aber trotzdem mit allen Robotern wiederholt werden. Die Bewegung wird maßgeblich durch die Einschränkungen, die durch die gegnerischen Roboter entstehen, beeinflusst und muss daher auch in dieser Situation getestet werden. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T700 |
| Tester | Anna Lörke |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. In einem späteren Testlauf, wenn beide Teams mit der Programmierung fertig sind, sollte der Test mit allen Robotern im Spiel wiederholt werden. |
| Soll - Reaktion | Im Spielverlauf passen sich zwei der Roboter einen Ball zu. |
| Ist - Reaktion | Der Pass wurde erfolgreich durchgeführt. |
| Ergebnis | Der Testlauf ist vorerst erfolgreich, sollte aber trotzdem mit allen Robotern wiederholt werden, da die Gegner einen großen Einfluss auf das Spiel- und Passverhalten der eigenen Roboter haben. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T800 |
| Tester | Alexander Joost |
| Eingaben | Es werden der Ball und die eigenen Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. In einem späteren Testlauf, wenn beide Teams mit der Programmierung fertig sind, sollte der Test mit allen Robotern im Spiel wiederholt werden. |
| Soll - Reaktion | Im Spielverlauf schießt einer der Roboter ein Tor. |
| Ist - Reaktion | Ein eigener Roboter hat ein Tor erzielt. |
| Ergebnis | Der Testlauf ist vorerst erfolgreich, sollte aber trotzdem mit allen Robotern wiederholt werden. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T900 |
| Tester | Alexander Joost |
| Eingaben | Es werden der Ball und alle sechs, gegnerische und eigene, Roboter spielbereit auf dem Spielfeld positioniert. Die GUI wird aufgerufen. Nun wählt man die linke Schaltfläche mit der Darstellung eines Fußballspielers. Dann wird der „connect“-Button und außerdem der „send Position“-Button ausgewählt und das Spiel mit Vorrang für das eigene Team, mit einem Klick auf den entsprechenden Button, gestartet. |
| Soll - Reaktion | Im Spielverlauf sollten die eigenen Roboter dem gegnerischen Team den Ball abnehmen, indem sie ihn bei Pässen abfangen, oder ihn an sich nehmen, wenn die Gegner ihn beim Bewegen verloren haben. |
| Ist - Reaktion | Der Testfall konnte noch nicht durchgeführt werden, da beide Teams noch mit den letzten Zügen der Programmierung und Optimierung beschäftigt sind. |
| Ergebnis | - |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------|
| Testfall | T1000 |
| Tester | - |
| Eingaben | Die optionale manuelle Steuerung via Controller wurde nicht implementiert und kann daher nicht getestet werden. |
| Soll - Reaktion | - |
| Ist - Reaktion | - |
| Ergebnis | - |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

| | |
|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Testfall | T1100 |
| Tester | Bengt Kensy |
| Eingaben | Die GUI wird gestartet und dann das rechte Symbol mit dem Controller für die manuelle Steuerung gewählt. Nun wählt man Roboter 6 aus und testet alle Bewegungsmethoden die auch in den Testdurchläufen zu <T6001> geprüft werden. |
| Soll - Reaktion | Siehe Testläufe zu <T6001>. |
| Ist - Reaktion | Siehe Soll-Reaktion. |
| Ergebnis | Der Testlauf war erfolgreich. |
| Unvorhergesehene Ereignisse | - |
| Nacharbeiten | - |

3.3 Zusammenfassung

Die Abnahmetests waren, sofern durchführbar, erfolgreich. Es müssen zu einem späteren Zeitpunkt jedoch viele Tests noch mal wiederholt, bzw. generell durchgeführt werden, da sie in Interaktion mit dem gegnerischen Team erfolgen müssen. Über die tatsächliche Funktionalität des Produkts lässt sich erst danach ein umfassendes Fazit erstellen. Auch müssen die Ungenauigkeiten der Hardware berücksichtigt werden, die nicht vom eigentlichen Produkt zu beeinflussen sind. Da die optionale Funktion <120> Manuelle Steuerung via Controller nicht implementiert wurde, ist der zugehörige Testfall <T1000> entfallen.

Dies war ein vorläufiger, interner Abnahmetest. Der finale Abgabetest mit dem Kunden erfolgt zu einem späteren Zeitpunkt.