

**Installation**

Semesterprojekt WS 2017/18

3D-Scanner  
Schienensystem

# Inhaltsverzeichnis

Inhaltsverzeichnis 2

1 Einleitung 3

1.1 Projektstruktur 3

1.2 Überblick der verwendeten Komponenten für die Bedienungsoberfläche 3

1.3 Überblick der verwendeten Komponenten für das Schienensystem 5

2 Bedienungsoberfläche 7

2.1 Einrichten der Entwicklungsumgebung 7

2.1.1 Java installieren 7

2.1.2 Installation der leJOS Tools 7

2.1.3 Eclipse aufsetzen 7

2.2 Vorbereitung des EV3 Bricks 10

2.2.1 Formatieren der MicroSD Karte 10

2.2.2 MicroSD Karte an EV3 Brick anschließen 11

2.3 Herstellen einer Bluetooth-Verbindung von Rechner zu EV3 Brick 11

2.4 Entwicklung eines leJOS Programms 12

2.4.1 Erstellen eines Eclipse Projekts 12

2.4.2 Ausführen eines leJOS Programms 12

2.5 Entwicklung der Kontrolloberfläche 15

2.5.1 Scanner 15

2.5.2 ScannerEV3 17

3 Schienensystem 18

# Einleitung

## Projektstruktur

Dieses Dokument dient dazu, projektfremden Personen das Aufsetzen der Entwicklungsumgebung für das Projekt 3D-Scanner Schienensystem zu ermöglichen.

Das Projekt ist in zwei Hauptbereiche gegliedert:

* **Bedienungsoberfläche (Software)**: ein eigens entwickelte Scanner-Kontrolloberfläche in Kombination mit der Sense Software
* **Schienensystem (Hardware)**: das gesamte Gerüst, worauf der 3D-Scanner angebracht und um die zu scannende Person/das zu scannende Objekt bewegt wird

Genaueres zur Systemarchitektur finden Sie im Dokument /Documentation/TechnicalDocumentation/**SystemArchitecture.docx**.

## Überblick der verwendeten Komponenten für die Bedienungsoberfläche

Zur Entwicklung des Teilbereichs für die Bedienungsoberfläche sind einige Werkzeuge notwendig, um Source-Code zu verfassen, testen sowie eine Release-Version zu erstellen.

Systemvoraussetzung für dieses Projekt ist **Windows 10**.

Folgende Software und Hardware wird zur Entwicklung der Bedienungsoberfläche verwendet (wie und wo eine Komponente jeweils verwendet wird, wird in späteren Abschnitten erläutert):

|  |  |
| --- | --- |
| **Software und Version** | **Details** |
| **LEGO Mindstorms EV3 Brick**  https://sh-s7-live-s.legocdn.com/is/image/LEGO/45500?$PDPDefault$ | <https://shop.lego.com/de-AT/Intelligenter-EV3-Stein-45500>  Lego Mindstorms Brick, der verwendet wird, um die Motoren anzusteuern; Brick ist in jedem Lego Mindstorms Bausatz enthalten. |
| **Sense 3D-Scanner (1st gen)** | <https://www.3dsystems.com/shop/sense> |
| **Sense Software** | <https://www.3dsystems.com/shop/support/sense/downloads> |
| **USB Kabel** |  |
| **MicroSD Card** |  |
| **MicroSD Card Reader** | Notwendig ist jedenfalls ein Adapter, mit dem die MircoSD Karte schließlich an den Entwicklungsrechner angeschlossen werden kann. |
| **Java JDK 1.7** | <http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html>  Auf dem Brick ist derzeit keine neuere Java-Version lauffähig. |
| **Java JDK 1.8** | [http://www.oracle.com/technetwork/java/ javase/downloads/jdk8-downloads-2133151.html](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html)  Zur Ausführung von Eclipse ist Java 8 empfehlenswert. |
| **Eclipse IDE für Java Developers** | <https://www.eclipse.org/downloads/> |
| **leJOS Plugin for Eclipse** | siehe Eclipse Market Place |
| **leJOS EV3 0.9.1-beta (EXE) Tools** | <https://sourceforge.net/projects/ev3.lejos.p/files/> |
| **JRE für Lego Mindstorms** | <http://www.oracle.com/technetwork/java/embedded/downloads/javase/javaseemeddedev3-1982511.html> |

## Überblick der verwendeten Komponenten für das Schienensystem

|  |  |
| --- | --- |
| **Software und Version** | **Details** |
| **LEGO Mindstorms Bausatz** | <https://www.lego.com/en-us/mindstorms/build-a-robot> |
| **2x Holzplatte**  Bildergebnis für holzplatte | 150cm x 75cm x 1,9cm (LxBxH) |
| **2x Scharniere**  Bildergebnis für scharniere |  |
| **Aktives USB Kabel** | Optimale Länge: 10m |
| **Holzkeile**  Bildergebnis für holzkeile |  |
| **3x Möbelfüße** | Seitenlänge Anbringung: 6cm, Höhe: ca. 15cm |
| **3x Stahlrohrfüße**  Bildergebnis für stahl rohr |  |
| **Kamerastativ (flexibel)** |  |
| **Hilfsmaterialen** | Schrauben, Klebeband, Schleifpapier |
| **Werkzeuge** | Stichsäge, Schleifmaschine, Akkuschrauber |

# Bedienungsoberfläche

## Einrichten der Entwicklungsumgebung

### Java installieren

Folgende Java-Tools müssen installiert werden:

* **JDK 1.8** zur Ausführung von Eclipse
* **JDK 1.7** zur Programmierung des Bricks (dieser unterstützt derzeit nur  
   Java 7)

### Installation der leJOS Tools

Damit ausführbare Programme für den EV3 Brick entwickelt werden können, sind die leJOS Tools notwendig. Durch Installation der Tools (C:\Program Files\leJOS EV3) werden

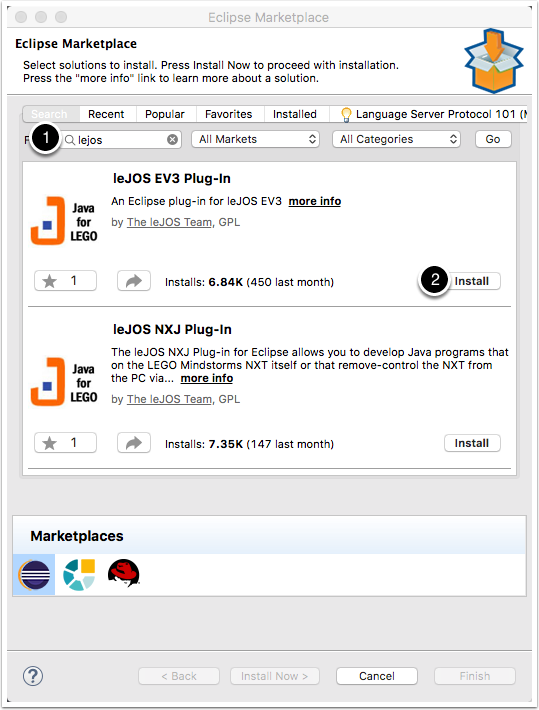
* Java Libraries
* sowie das leJOS Betriebssystem für den Brick

bereitgestellt.

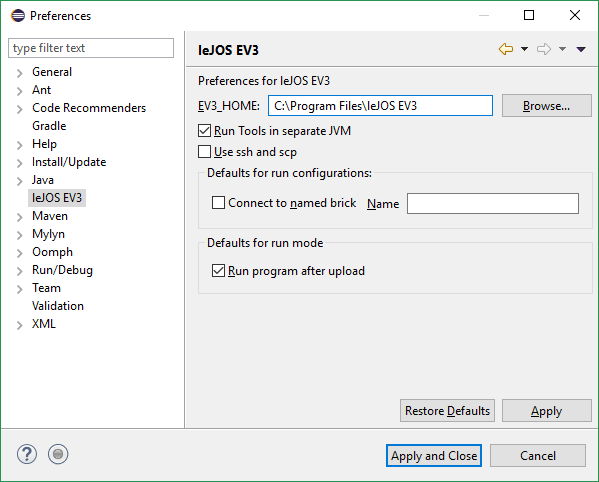
### Eclipse aufsetzen

Entsprechend der obigen Auflistung wird Eclipse für Java Developers auf dem Entwicklungsrechner installiert.

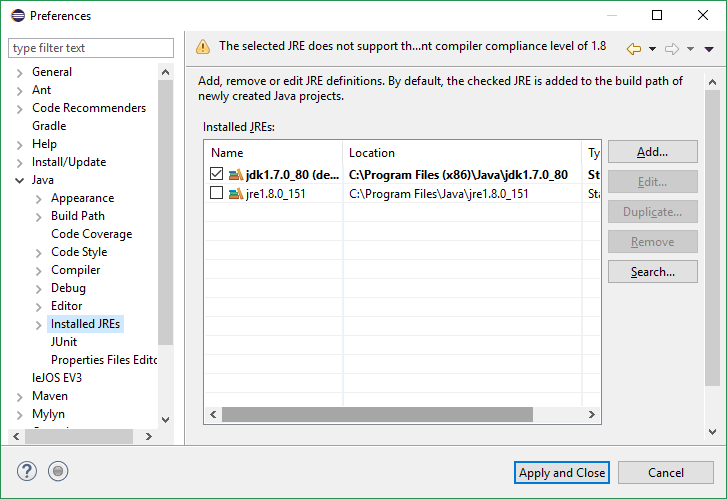
Weiters dient das leJOS Eclipse Plugin später zur Installation von leJOS Projekten auf dem Brick. Dazu wird folgendes Paket für Eclipse installiert:



Unter Window>Preferences können die Settings für leJOS nochmals überprüft werden.



Unter Window>Preferences>Java>Installed JREs wird das JDK 1.7 hinzugefügt.

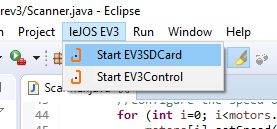


## Vorbereitung des EV3 Bricks

### Formatieren der MicroSD Karte

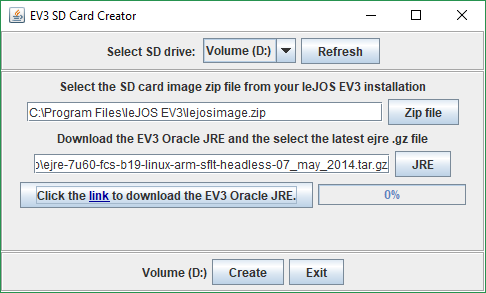
Zu Beginn wird die MicroSD Karte mit passendem Adapter mit dem Entwicklungsrechner verbunden.

Von Eclipse aus wird der leJOS SD-Card Formatter gestartet.



Drei Dinge werden dabei festgelegt:

1. Auswahl des Laufwerkbuchstabens der MicroSD Karte
2. Pfad zum leJOS Image (standardmäßig: C:\Program Files\leJOS EV3\lejosimage.zip)
3. Pfad zum JRE für Mindstorms (zB C:\...\Downloads\ ejre-7u60-fcs-b19-linux-arm-sflt-headless-07\_may\_2014.tar.gz)



Mit „Create“ wird die Vorbereitung der MicroSD Karte abgeschlossen.

### MicroSD Karte an EV3 Brick anschließen

Die mit leJOS ausgestattete MicroSD Karte wird an den Lego Mindstorms Brick angeschlossen.

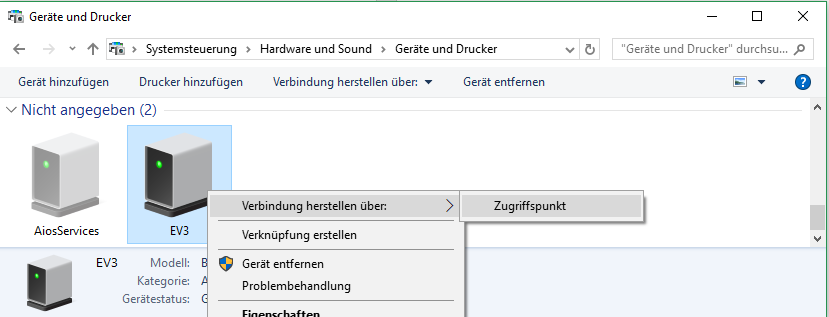


Wird der Brick anschließend gestartet (mittlerer Knopf), wird von MicroSD Karte gebootet und leJOS gestartet.

## Herstellen einer Bluetooth-Verbindung von Rechner zu EV3 Brick

Als ersten Schritt muss vom PC ein Pairing mit dem Brick als Bluetooth-Gerät hergestellt werden.

1. Am EV3 Brick wird im Bluetooth Menü die Sichtbarkeit eingestellt.
2. In Windows 10 kann nun der Brick unter „Bluetooth- und andere Geräte“ gefunden werden (Bluetooth- oder ein anderes Gerät hinzufügen).
3. Sobald der Brick gefunden ist (Name: „EV3“) wird die Kopplung mit dem Pin des Bricks bestätigt. Dieser lautet standardmäßig „1234“.
4. Als letzten Schritt muss ein PAN mit dem Brick hergestellt werden (ähnlich wie ein Wireless Netzwerk), damit eine TCP/IP Kommunikation zwischen Rechner und Brick erfolgen kann. Die Vorgehensweise ist wie folgt im Screenshot:

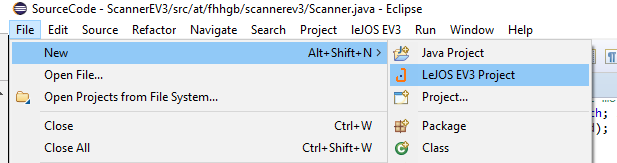


## Entwicklung eines leJOS Programms

### Erstellen eines Eclipse Projekts

In Eclipse steht durch das leJOS Plugin ein neues Projekt-Template zur Verfügung.

Ein neues „LeJOS EV3 Project“ wird demnach definiert.



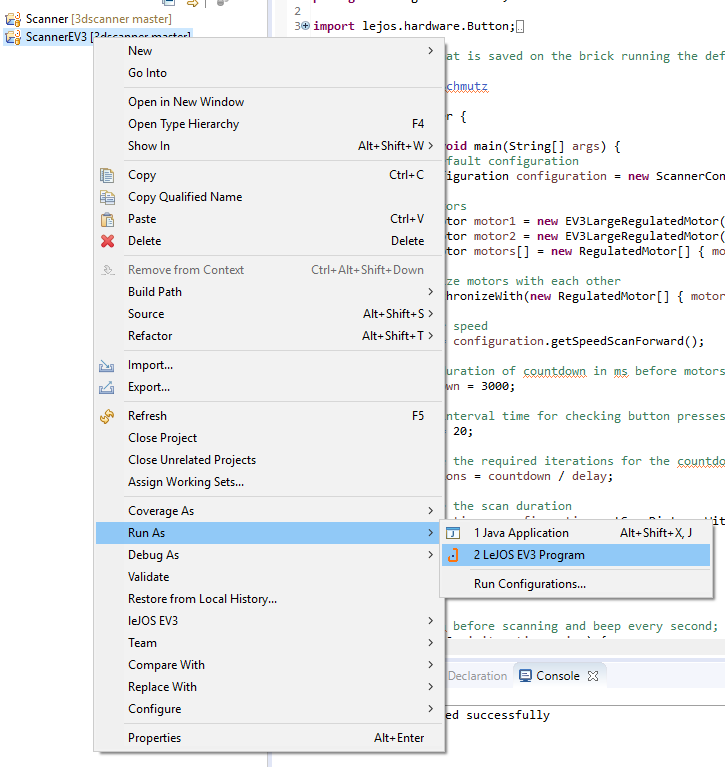
### Ausführen eines leJOS Programms

Für die Entwicklung von leJOS Projekten bestehen zwei Varianten:

1. Installation & Ausführung des Java Programms auf dem EV3 Brick
2. Ausführung des Java Programms auf dem Entwicklungsrechner und Steuerung des EV3 Bricks mittels RMI

**Variante a)**

Zur Installation und Ausführung des Programms auf dem Brick, wird das leJOS Plugin für Eclipse genutzt und beim Starten des Programs (Run As) ausgewählt.



Anmerkung: Dabei muss eine aktive Verbindung von Rechner zu Brick bestehen (Bluetooth, USB oder WiFi; in diesem Dokument wird lediglich die Variante Bluetooth betrachtet).

Beispielprogramm:

public class HelloWorld **{**

public static void main**(**String**[]** args**)** **{**

System**.**out**.**println**(**"Hello World!"**);**

Sound.beep();

Button**.**waitForAnyPress**();**

**}**

**}**

**Variante b)**

Hierbei wird das Java-Programm lokal auf dem Rechner ausgeführt. Mithilfe der entsprechenden Klassen kann mittels RMI der Brick gesteuert werden, wenn eine Verbindung von dem Rechner zum Brick besteht.

Beispielprogramm:

public class HelloWorld **{**

public static void main**(**String**[]** args**)** **{**

Brick brick **=** **new** RemoteEV3**(**"10.0.1.1"**);**

TextLCD lcd **=** brick**.**getTextLCD**();**

lcd**.**drawString**(**"Hello World!"**,** 0**,** 0**);**

Audio audio **=** brick**.**getAudio**();**

audio**.**systemSound**(**0**);**

**}**

**}**

**Anmerkung**: Es ist zu beachten, dass sich die verwendeten Klassen für Variante a) und b) unterscheiden!

Begründung:

Variante a) wird direkt auf dem Brick ausgeführt.

Variante b) steuert den Brick vom Rechner aus mittels RMI.

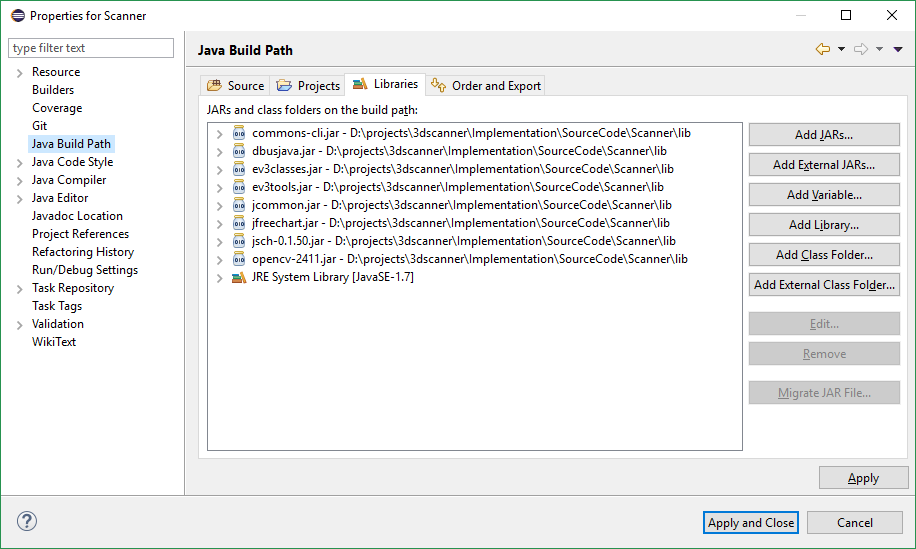
## Entwicklung der Kontrolloberfläche

Das Gesamtpaket besteht aus zwei Projekten:

* **Scanner** Java-Programm für PC mit grafischer Oberfläche zur  
   Steuerung des Bricks per Bluetooth und RMI
* **ScannerEV3** Java-Programm für den Brick, das als Ausfallmaßnahme  
   des ersten Programms dienen soll und beim Start einen  
   Scan-Vorgang durchführt.

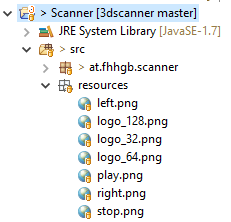
### Scanner

Für das Hauptprojekt am Rechner werden dazu die benötigten JARs für leJOS nochmal manuelle hinzugefügt. Project>Properties>Java Build Path



Grund für diese Maßnahme ist, dass anschließend eine einzelne ausführbare JAR-Datei exportiert wird, die all die leJOS JARs beinhalten soll.

Weiters werden Bilder in ein eigenes Package im Java Projekt eingefügt:



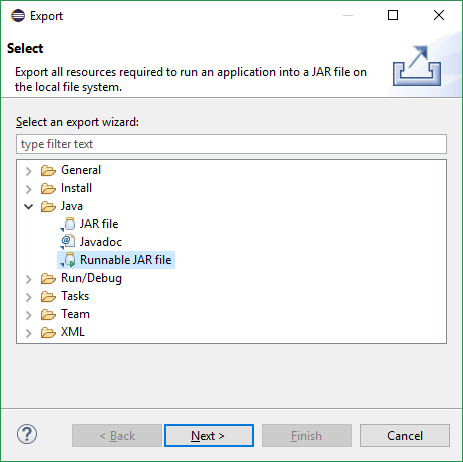
In Java werden die Bilder wie folgt ausgelesen:

Image img **=**

ImageIO**.**read**(**getClass**().**getResource**(**"/resources/left.png"**));**

Das hat den Vorteil, dass anschließend auch die Bilder in die endgültige JAR Datei gepackt werden können.

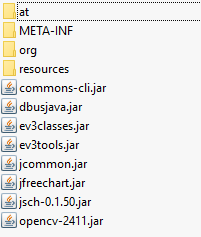
Als letzten Schritt wird in Eclipse das Projekt als JAR exportiert.



Dabei kann eine dieser beiden Varianten gewählt werden:



Bei Wahl der im Screenshot abgebildeten Variante hat die resultierende JAR-Datei folgenden Inhalt:



### ScannerEV3

Das Backup-Programm wird lediglich über Eclipse auf den Brick installiert, wie es in Punkt 2.4.2 Ausführen eines leJOS Programms Variante a) beschrieben ist.

# Schienensystem