

Entwicklungsprojekt  
Interaktive Systeme  
Installationsdokumentation

vorgelegt an der

Technische Hochschule Köln – Fakultät für  
Informatik und Ingenieurwissenschaften

ausgearbeitet von

Leonid Vilents

betreut durch

Prof. Dr. Kristian Fischer

Prof. Dr. Gerhard Hartmann

B. Sc. Sheree Saßmannshausen

B. Sc. Daniela Reschke

## Inhaltsverzeichnis

Systemvoraussetzungen.....	3
Voraussetzungen Serverlogik .....	3
Voraussetzungen Client.....	3
Installation.....	4
Installation und Inbetriebnahme des Servers .....	4
Installation und Inbetriebnahme des Clients .....	4

## Systemvoraussetzungen

### Voraussetzungen Serverlogik

Für das Betreiben des Systems wird ein Computersystem benötigt, auf dessen Betriebssystem das JavaScript-Framework Node.js lauffähig ist. Die offizielle Webseite von NodeJS bietet Installationspakete für folgende Betriebssysteme nativ:

- Microsoft Windows (32bit, 64bit) als Installer (.msi) oder Binary (.zip)
- Apple macOS (64bit) als Installer (.msi) oder Binary (.tar.gz)
- Linux-Distributionen (32bit, 64bit) als Binary oder ARM (6, 7, 8)
- SunOS (32bit, 64bit) als Binary
- Docker-Image
- AIX (64bit)

Ferner wird auch der Sourcecode als .tar.gz zum selber Kompilieren angeboten.

Zusätzlich zu Node.js, wird ein MySQL-Server benötigt. Neben der Enterprise-Edition für Unternehmen, bietet MySQL auch unter der GPL-Lizenz einen Community-Server an.

### Voraussetzungen Client

Für das Betreiben der Client-Anwendung für Endbenutzer, wird die Version 8 des Java Runtime Environments benötigt. Java wird von folgenden Betriebssystemen unterstützt:

- Microsoft Windows ab Vista SP2
- Microsoft Server 2009 R2 SP1 (64bit)
- Microsoft Server 2012, 2012 R2 (64bit)
- Apple macOSX ab Version 10.7.3
- Linux / RPM-Linux (32bit, 64bit)
- Solaris (32bit, 64bit)

## Installation

### Installation und Inbetriebnahme des Servers

Für die Installation des Servers wird davon ausgegangen, dass MySQL und Node.js bereits auf dem Zielsystem installiert sind. Es folgen nun die Installationsanweisungen für Windows und für unixoide Systeme.

1. Laden Sie den Server-Teil der Repository herunter ( <https://github.com/scarcloud/EISSS17Vilents/tree/master/MS3/Implementation/Server> ).
2. Die Schemadatei für die Datenbank befindet sich unter „data“ als „schema.sql“. Starten Sie ihren MySQL-Server und führen Sie diese Datei in dessen Benutzer-Interface (Kommandozeile oder Dritt-Programm) aus. Falls Sie an der Schemadatei Änderungen vornehmen möchten (z.B. Benutzernamen ändern), ändern Sie auch die Zugangsdaten in der „dbam.json“-Konfigurationsdatei (zu finden unter „util/config“).
3. Falls Sie das System mit Testdaten befüllen wollen, existiert unter <https://github.com/scarcloud/EISSS17Vilents/tree/master/MS3/Implementation> eine Rubrik „Daten“, unter der man eine Fixtures-Datei findet.
4. Öffnen Sie die Kommandozeile im Ordner mit der Hauptdatei „server.js“, und führen Sie diese mit Node.js („node server.js“) aus. Der Server wird jetzt versuchen, sich mit dem MySQL-Server zu verbinden. Bei Erfolg akzeptiert der Server jetzt Http-Requests

### Installation und Inbetriebnahme des Clients

Für die Installation des Clients wird davon ausgegangen, dass Java bereits auf dem Zielsystem installiert ist.

1. Laden Sie unter <https://github.com/scarcloud/EISSS17Vilents/blob/master/MS3/Implementation/Client/client.zip> den kompilierten Client als ZIP-Datei herunter.
2. Entpacken Sie den ZIP-Ordner und führen sie client.jar aus.
3. Der Client wird jetzt laden, und Sie können sich entweder einloggen oder registrieren.