



BEUTH HOCHSCHULE FÜR TECHNIK BERLIN  
University of Applied Sciences



Beuth Hochschule für Technik Berlin  
Fachbereich VI - Informatik und Medien  
Luxemburger Str. 10, 13353 Berlin

# Bachelorarbeit

## Entwicklung einer Datenbank-Applikation für das "acaLoan-Raspi"-Projekt des PSE-Labors

Development of a database application for the "acaLoan-Raspi"  
project of the PSE laboratory

Oleksandra Baga

Beuth Hochschule für Technik Berlin  
Matrikelnummer 849852  
Bachelor of Engineering (B.Eng.)  
Computer Engineering - Embedded Systems  
E-Mail: oleksandra.baga@gmail.com

*Supervisor* Prof. Dr. Christian Forler  
Fachbereich VI - Informatik und Medien  
Beuth Hochschule für Technik Berlin

31.10.2020

# Contents

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation und Aufgabestellung . . . . .	1
1.2	Technische Basis und Themengebiet . . . . .	2
<b>2</b>	<b>Theoretische Hintergrund</b>	<b>3</b>
2.1	Über Raspberry Pi Board und OS . . . . .	3
2.2	Kontaktlose Chipkartentechnik MIFARE . . . . .	3
2.3	Sender-Empfänger-System mit RFID . . . . .	3
2.4	Datenbanken mit Python und SQLite . . . . .	3
2.5	HTTP für Design der verteilten Systeme . . . . .	3
2.6	Django Framework . . . . .	3
2.7	Endliche Zustandsmaschine . . . . .	3
2.8	Clientseitiges JavaScript . . . . .	3
<b>3</b>	<b>Systemdesign</b>	<b>4</b>
3.1	User Stories . . . . .	4
3.2	UML . . . . .	4
3.3	Endliche Zustandsmaschine . . . . .	4
<b>4</b>	<b>Implementation</b>	<b>5</b>
4.1	Register-Client . . . . .	5
4.2	Server . . . . .	5
4.3	Display-Client . . . . .	5
<b>5</b>	<b>Application</b>	<b>6</b>
5.1	Usage in laboratory . . . . .	6
<b>6</b>	<b>Results and conclusion</b>	<b>7</b>
6.1	Results . . . . .	7
6.2	Conclusion . . . . .	7
<b>A</b>	<b>Appendix</b>	<b>8</b>
A.1	mikroBUS . . . . .	8
<b>B</b>	<b>List of abbreviations</b>	<b>9</b>



# Einleitung

- **Röst-Mahl-Koch-Kaffeemaschine:** Integration von Connectivity und Entwicklung von IoT-Services für Röst-Mahl-Koch-Kaffeemaschine für Bonaverde<sup>1</sup>.

## 1.1 Motivation und Aufgabestellung

Die vorliegende Arbeit beschäftigt sich mit der Entwicklung einer Datenbank-Applikation für das PSE-Labor (Labor für Pervasive Systems Engineering), das sich an der Beuth Hochschule für Technik Berlin befindet und seit fast 5 Jahre ein wichtiger Teil des Studiums im Studiengänge Technische Informatik und Medieninformatik ist. Während im PSE-Labor stattfindenden Übungsveranstaltungen im Studiengang Technische Informatik werden vorhandene im PSE-Labor die Raspberry Pi Minicomputers (kurz: Raspi) an die Studierenden verliehen. Zu Beginn einer Laborübung werden die Raspi Boards den Studierenden vom Lehrkraft, der die Übung betreut, übergeben und am Ende der Laborübung zurückgezogen. Nachweislich ist das Vorgehen oft mit Reihe von Problemen verknüpft, die sich jedes Semester und fast jedes Mal wiederholen. Die folgenden Problemen wurden von Mitarbeitern des Labors bereits festgestellt und verlangsamten den Prozess der Verleihung und Übungsführung:

- Studierende kennen ihre am Semesteranfang zugewiesene Gruppennummer auch nach mehreren Wochen nicht und geben den Lehrkraft einen Board mit einer falschen Registriernummer, der einer anderen Gruppe früher zugewiesen wurde und nur von der zugewiesenen Gruppe benutzt werden darf.
- Studierende versuchen einen Board nach Hause auszuleihen, der zu den Lab-Boards gehört und nur im Labor während der Übungszeit verliehen werden darf. Außerplanmäßig von Studierenden darf Lab-Board nicht ausgeliehen und auch mit nach Hause (home-loan) nicht genommen werden.
- Es gibt ein Verwaltungsaufwand für die ausleihbaren Home-Boards, die von den Studierenden für jeweils eine Woche mit nach Hause genommen werden können. Die Mitarbeiter müssen handlich die Studentennamen, Matrikelnum-

---

<sup>1</sup><https://www.bonaverde.com/>

mer, Board und Zeit am Zettel registrieren und in einer Woche überprüfen, ob alle ausgeliehenen Boards pünktlich ins Labor zurückgekommen sind.

- Erfahrungsgemäß können Studierende nach Ablauf der Frist ein Ausleihgerät in einem sehr üblen Zustand der Verschmutzung oder Zerstörung zurückgeben, dass es besteht eine Notwendigkeit den Zustand des Gerätes stets zu kontrollieren, damit es immer bekannt wird, zum welchen Zeitraum Raspi Board zum letzten Mal funktionsfähig war und von wem ausgeliehen wurde.
- Falls gilt ein Raspi Board als verloren, es sollte eine Möglichkeit geben, alle vorherigen Ausleihen anzuschauen und festzustellen, von welchem Studierende es ausgeliehen und nicht zurückgegeben wurde. Mit den Zettelchen, auf denen den Name von Studierende und Board Nummer gemerkt werden, ist es zu aufwändig nachvollziehen.

## 1.2 Technische Basis und Themengebiet

Im PSE-Labor laufen mittlerweile sehr viele von den beiden Mitarbeitern, Andreas und Brian, in eigener Regie geführte und nicht. Da diese Projekte formal von den offiziellen Labor-Tätigkeiten und -Schwerpunkten des Labors unterschieden werden müssen und diesen auch nicht zugerechnet werden dürfen.

## Theoretische Hintergrund

- 2.1 Über Raspberry Pi Board und OS
- 2.2 Kontaktlose Chipkartentechnik MIFARE
- 2.3 Sender-Empfänger-System mit RFID
- 2.4 Datenbanken mit Python und SQLite
- 2.5 HTTP für Design der verteilten Systeme
- 2.6 Django Framework
- 2.7 Endliche Zustandsmaschine
- 2.8 Clientseitiges JavaScript

# Systemdesign

## 3.1 User Stories

## 3.2 UML

## 3.3 Endliche Zustandsmaschine

# Implementation

## 4.1 Register-Client

Der Standard legt das physikalische Layout der mikroBus-Pinout-Verbindung, die verwendeten Kommunikations- und Stromversorgungspins auf dem Mainboard fest.

## 4.2 Server

Die *Cloud* oder *Cloud Computing* Begriff kommt offensichtlich aus dem Englischen und heißt auf Deutsch "Wolke". Der Begriff beschreibt einen oder mehrere entfernte Server, auf die man seine Daten von einem Gerät über das Internet hochladen kann. Dann übernimmt die Cloud die Aufgaben wie die Datenverarbeitung oder komplizierte Programmabläufe. Während der Datenverarbeitung weißt der Nutzer nicht, wie viele Server hinter der Cloud stecken und welche komplizierte Hardware für die Berechnungen benötigt werden.

## 4.3 Display-Client

Die *Cloud* oder *Cloud Computing* Begriff kommt offensichtlich aus dem Englischen und heißt auf Deutsch "Wolke". Der Begriff beschreibt einen oder mehrere entfernte Server, auf die man seine Daten von einem Gerät über das Internet hochladen kann. Dann übernimmt die Cloud die Aufgaben wie die Datenverarbeitung oder komplizierte Programmabläufe. Während der Datenverarbeitung weißt der Nutzer nicht, wie viele Server hinter der Cloud stecken und welche komplizierte Hardware für die Berechnungen benötigt werden.



# Application

## 5.1 Usage in laboratory

Der Standard legt das physikalische Layout der mikroBus-Pinout-Verbindung, die verwendeten Kommunikations- und Stromversorgungspins auf dem Mainboard fest. Der Zweck von mikroBUS ist es, eine einfache Erweiterbarkeit der Hardware mit einer großen Anzahl von standardisierten kompakten Zusatzboards zu ermöglichen, von denen jede einen einzelnen Sensor, Display, Encoder oder Motortreiber, eine integrierte Schaltung hat. Der von MikroElektronika entwickelte mikroBUS ist ein offener Standard - jeder kann mikroBUS in seinem Hardwaredesign implementieren. Die Abbildung<sup>1</sup> ?? zeigt die Pinout Spezifikation des Herstellers, die man entsprechend ändern kann und die neue Verbindungen für den eigenen Projekt feststellen. Wenn ein Modul eine Schnittstelle verwendet, die bereits auf mikroBUS vorhanden ist, benutzt man diese exakten Pins und markiert diese entsprechend. Wenn ein Pin nicht verwendet wird, sollte er als NC (für "Not Connected") markiert sein.

---

<sup>1</sup><https://download.mikroe.com/documents/standards/mikrobus/mikrobus-standard-specification-v200.pdf>

## Results and conclusion

### 6.1 Results

Der Standard legt das physikalische Layout der mikroBus-Pinout-Verbindung, die verwendeten Kommunikations- und Stromversorgungspins auf dem Mainboard fest. Der Zweck von mikroBUS ist es, eine einfache Erweiterbarkeit der Hardware mit einer großen Anzahl von standardisierten kompakten Zusatzboards zu ermöglichen, von denen jede einen einzelnen Sensor, Display, Encoder oder Motortreiber, eine integrierte Schaltung hat.

### 6.2 Conclusion

Der Standard legt das physikalische Layout der mikroBus-Pinout-Verbindung, die verwendeten Kommunikations- und Stromversorgungspins auf dem Mainboard fest. Der Zweck von mikroBUS ist es, eine einfache Erweiterbarkeit der Hardware mit einer großen Anzahl von standardisierten kompakten Zusatzboards zu ermöglichen, von denen jede einen einzelnen Sensor, Display, Encoder oder Motortreiber, eine integrierte Schaltung hat.

# Appendix

## A.1 mikroBUS

Der Standard legt das physikalische Layout der mikroBus-Pinout-Verbindung, die verwendeten Kommunikations- und Stromversorgungspins auf dem Mainboard fest.

## List of abbreviations

<b>BOM</b>	Bill of Materials
<b>DEVKIT</b>	Development Kit
<b>Cloud</b>	Cloud Computing
<b>EKG</b>	Elektrokardiogramm
<b>GmbH</b>	Gesellschaft mit beschränkter Haftung
<b>I<sup>2</sup>S</b>	Inter-IC Sound
<b>IoT</b>	Internet of Things, Internet der Dinge
<b>IT</b>	Informationstechnik, Bereich der Informations- und Datenverarbeitung
<b>JSON</b>	JavaScript Object Notation
<b>LED</b>	Light-emitting diode, Leuchtdiode
<b>MQTT</b>	Message Queuing Telemetry Transport
<b>NVS</b>	Non-volatile storage
<b>PCB</b>	Printed circuit board
<b>PWM</b>	Pulse Width Modulation
<b>SPI</b>	Serial Peripheral Interface Bus
<b>UV</b>	Ultraviolettstrahlung

List of images

C