



Technische
Universität
Braunschweig



SEP 2020

Organisatorisches

Prof. Dr.-Ing. Ina Schaefer, 06.02.2020

Allgemeine Informationen

- Gesamtorganisation:
Prof. Dr.-Ing Ina Schaefer
Institut für Softwaretechnik und Fahrzeuginformatik (ISF)
Mühlenpfordtstraße 23, Zimmer 347
Email: i.schaefer@tu-bs.de
- Ansprechpartner:
M.Sc. Tobias Runge
Email: tobias.runge@tu-bs.de
- Betreuung der Projektgruppen: Institute in der Informatik,
Wirtschaftsinformatik und Elektrotechnik
- Zentrale Webseite zum SEP:
<https://www.tu-braunschweig.de/isf/teaching/2020s/sep>

Lernziele

Laut Modulhandbuch

Nach Abschluss dieses Moduls besitzen die Studierenden ein grundlegendes Verständnis zur Entwicklung komplexer Softwaresysteme.

Sie sind prinzipiell in der Lage die Aufgabenstellung mit Modellen zu erfassen, in ein Design umzusetzen und zu implementieren.

Inhalte

- Erstellung, Dokumentation und Präsentation von Software
- Fokus nicht rein auf Programmierung
- Es soll der grundlegende Prozess gelernt werden, d.h.:
 - Softskills wie Teamwork, Selbstorganisation
 - Selbstständige Einarbeitung in neue Aufgaben und Themen
 - Überarbeitung von Dokumentation und Software anhand von externen Reviews
- Bearbeitung einer Aufgabenstellung in einer Kleingruppe, in der Regel 5 – 6 Teilnehmer
- Externe Dokumentenreviews durch das ISF zur Sicherung der Dokumentationsqualität

Empfohlene Voraussetzungen

- Die Programmierkenntnisse aus den Veranstaltungen Programmieren 1 und 2, sowie Kenntnisse des Stoffs aus Software Engineering 1 werden vorausgesetzt.
- Zusätzlich können für die einzelnen Projekte weitere Kenntnisse benötigt werden. Informieren Sie sich vorab auf den jeweiligen Projektseiten.
- Es ist nicht Aufgabe der Betreuer diese zu vermitteln, sondern Sie bei der Einarbeitung und Entwicklung lediglich zu unterstützen.

Bedingungen

- Jedes Gruppenmitglied muss zu allen Projektphasen inhaltlich beitragen.
- Jedes Gruppenmitglied muss sowohl zum Code als auch zur Dokumentation beitragen.
- Jedes Dokument muss vollständig bearbeitet abgegeben werden.
- Jedes Gruppenmitglied ist für die Vollständigkeit der Dokumente und deren pünktliche Abgabe verantwortlich.
- Jedes Dokument, das nach der Deadline eingereicht wird, gilt als nicht eingereicht.
- Nicht eingereichte Dokumente oder unzureichend bearbeitete Dokumente führen zu einer Verwarnung der Gruppe. Das Dokument muss nachbearbeitet werden.
- Ab der zweiten Verwarnung werden im Einzelfall mit Hinzunahme des Betreuers passende Maßnahmen ermittelt. Dies könnten eine Nachbearbeitung des Dokumentes, Disqualifikation der Gruppe oder andere Maßnahmen sein, die für den Fall passen.
- Die Teilnahme an allen Veranstaltungen ist verpflichtend.
- Alle SEP relevanten Dokumente sowie der Code werden ausschließlich im ISF Redmine/SVN versioniert.

Zulassungsvoraussetzungen

- Für Bachelorstudenten der Informatik und Wirtschaftsinformatik in den *neueren Prüfungsordnungen* (Info: PO \geq SoSe 2014, WInfo: PO \geq SoSe 2015) gilt: Das Bestehen *des Moduls SE1 (Studienleistung + Klausur)* ist Zulassungsvoraussetzung.
- Für Bachelorstudenten der Informatik und Wirtschaftsinformatik in den *älteren Prüfungsordnungen* (Info: PO < SoSe 2014, WInfo: PO < SoSe 2015) gilt: Das Bestehen *der Klausur SE1* ist Zulassungsvoraussetzung.
- IST-Studenten in älteren PO können ohne Zulassungsvoraussetzung am SEP teilnehmen.
- Seit der neuesten PO gilt für IST-Studenten: Das Bestehen *der Klausur SE1* ist Zulassungsvoraussetzung.
- **Im Zweifel beim jeweiligen Prüfungsamt erkundigen ob alle Zulassungsvoraussetzungen erfüllt sind!**

Anmeldung

- Anmeldezeitraum vom 07.02. – 07.03.2020 (**strikt!**)
- Link zur Anmeldewebsite auf der zentralen SEP-Webseite des ISF
- Anmeldeformular muss ausgefüllt in den Briefkasten des ISF (Foyer des Informatikzentrums) geworfen werden. **Anmeldung erst nach Eingang in Papierform abgeschlossen!**
- Bekanntgabe der Gruppeneinteilung nach der SE1-Wiederholungsklausur am 24.03.2020

Ablauf und Termine

- Kick-Off in den Projektgruppen: in KW 16 (14.04.-17.04.) nach Absprache mit den Betreuern
- Abgabe Angebot: 29.04. (bei den Betreuern)
- Abgabe Pflichtenheft & Abnahmetestspezifikation: 20.05.
- Zwischenpräsentation (inkl. Vorstellung des Prototypen): **29.05., Details und Raum werden noch bekannt gegeben**
- Abgabe Fachentwurf: 10.06.
- Abgabe Technischer Entwurf: 01.07.
- Abgabe Testdokumentation: 15.07.
- Tag der jungen Software Entwickler (TDSE): **Donnerstag, 23.07. nachmittags**

Projektvorstellung

- | | |
|--------------|-------------|
| 1) CIO | 2) IAS 4x |
| 3) IBR (ALG) | 4) IBR (CM) |
| 5) IBR (DS) | 6) IFIS |
| 7) IFN | 8) ISF |
| 9) IRP | 10) ISS |
| 11) TCS | 12) CG |
| 13) PLRI | |



Thema für das SEP Topic for the SDT

Chief Information Officer (CIO); H. Hess, H. Eggers

Lending 3.0

- Background
 - A flexible tool to manage the lending of equipment and goods is missing
 - The requirements are manifold. The solutions available at the market are somewhat ...
- Challenge
 - Build a web-based tool to manage the lending of goods.
 - Create a usable interface for hirer and customer.
 - Don't forget security and communication.
 - Create a truly digital process (except of the physical lending and return).
- Environmental conditions
 - I18n web-based GUI at least in English and German
 - Use of [Grails](#) (and maybe [Vue.js](#))
 - Publish your result as Open Source project



Contact

- Hendrik Eggers, Chief Information Officer (CIO)
- Preferable via E-Mail: h.eggers@tu-braunschweig.de
- English or German

- Consultation about details for the topic welcome

- Presence appointment scheduled for **Wednesdays 16:45 – 18:15 every week**

Vending Machine

Softwareentwicklungspraktikum Projektvorstellung

David Klein

Institute of Application Security
Technical University of Braunschweig

`david.klein@tu-braunschweig.de`

Vending Machine



Vending Machine – It's 2020!



Project Scope

- ▶ Getränkeautomat, der:
 - Getränke mischen kann (etwa Spezi aus Cola und Fanta)
 - über eine Grafische Benutzeroberfläche verfügt
 - ▶ Benutzerführung beim Einkauf
 - ▶ Informationen über Füllstand, Einnahmen, etc.
 - Bezahlen via einer Cryptocurrency erlaubt

Details

- ▶ Hardware:
 - Raspberry Pi
 - Sensoren und Aktuatoren
 - Von uns gestellt!
- ▶ Software:
 - Python
 - Web Technologien (HTML/CSS)
- ▶ Kann alles im Rahmen des Projekts erlernt werden!
- ▶ Organisatorisches:
 - 1 Gruppe mit ca. 6 Personen

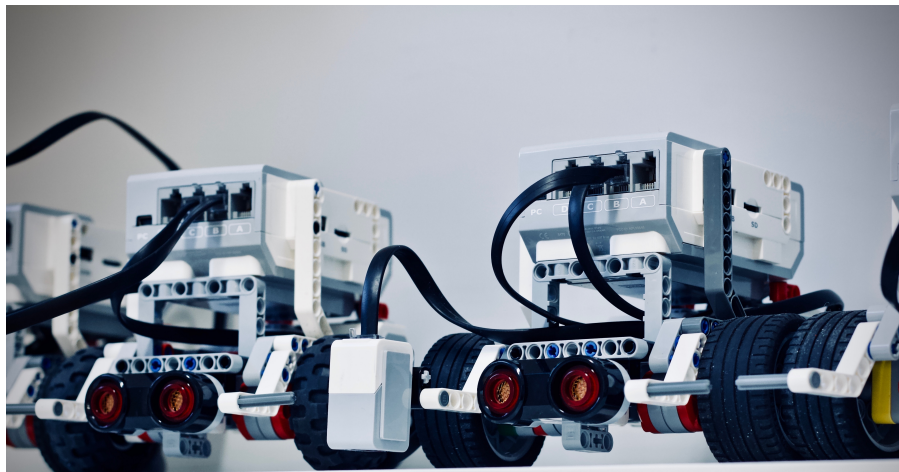
Lego Rumba

Manuel Karl

Institute of Application Security
Technical University of Braunschweig

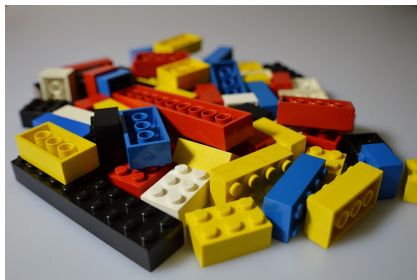
`m.karl@tu-braunschweig.de`

LEGO Mindstorms



Projektziel

- ▶ Ziel ist es mittels Lego Mindstorms einen Roboter zu bauen, um Lego-Bausteine aufzuräumen
- ▶ Hierzu soll der Roboter in der Lage sein Lego-Bausteine:
 - einzusammeln
 - anhand der Farbe zu klassifizieren
 - und entsprechend der Klassifizierung in vorbereitete Behältnisse einzusortieren



Technologien

- ▶ Hardware:
 - Lego Mindstorms
 - Raspberry PI
 - Sensoren und Aktuatoren
- ▶ Software:
 - Entweder Java oder Python
- ▶ Kann alles im Rahmen des Projekts erlernt werden!

Telegram as a Database (TaaD)

(Projektname: Telegrambot)



- Telegram erlaubt Nachrichten an sich selbst und andere mit Gesprächsverlauf
- Telegram erlaubt das Versenden von Daten (e.g., Bilder, Text, uvm.)
- Dies könnte man doch kombinieren



Aufgabe & Technologien



- Evaluation der Telegram automatisierungs API zur Nutzung von Telegram als (geteilte?) Datenbankanwendung
- Umsetzung der Evaluation in einen Prototypen
- Telegram API
- beliebige Programmiersprache
- 1 Team
- 5 Personen



CHIPS^{extended}

Softwareentwicklungspraktikum Projektvorstellung

Alexandra Dirksen

Institute of Application Security
Technical University of Braunschweig

`a.dirksen@tu-bs.de`

Content Hidden In Plain Sight

- ▶ publicly visible blog/website

Content Hidden In Plain Sight

- ▶ publicly visible blog/website
- ▶ publishing encrypted content

Content Hidden In Plain Sight

- ▶ publicly visible blog/website
- ▶ publishing encrypted content
- ▶ Android app vor decryption

Content Hidden In Plain Sight

- ▶ publicly visible blog/website
- ▶ publishing encrypted content
- ▶ Android app vor decryption
- ▶ encryption for privileged users
- ▶ hiding ciphers from others

Text to post

Public ▾

Post Reset

public message: Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam

public message: At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren.

(a) Browser

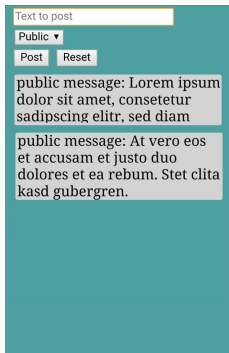
CHIPS

public message: Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam

secret for user 1: Meeting at 5 pm. Don't tell anyone.

public message: At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren.

(b) Android



Text to post

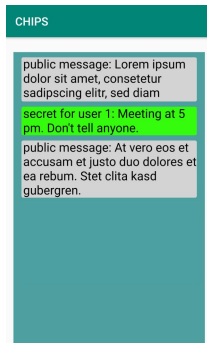
Public ▾

Post Reset

public message: Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam

public message: At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren.

(a) Browser



CHIPS

public message: Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam

secret for user 1: Meeting at 5 pm. Don't tell anyone.

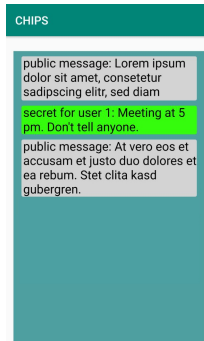
public message: At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren.

(b) Android

- 👉 using public key crypto
- 👉 hiding ciphertext in html tag



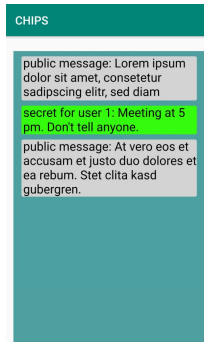
(a) Browser



(b) Android



(a) Browser

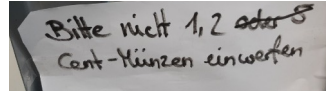
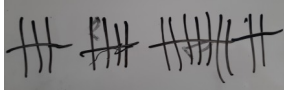


(b) Android

- 👍 using attribute-based encryption
- 👍 hiding cipher in adorable cat pictures

Project Scope

- ▶ (opt.) reusing the CHIPS project
- ▶ ABE implementation (by JT)
- ▶ using different encryption policies
- ▶ using Steganography for cipher hiding



SEP'20 — IBR Algorithmik Kitty — Smarte Kaffeekasse

Phillip Keldenich, Dominik Krupke

03.02.2020

Problem 1



Name	Alex				
Startkapital	0			5,20	
Einzahlung		10,60	15,00		
Mate	HTI	HTI	HTI	HTI	HTI
Kaffee Bismade				HTI	
Schorle			HTI	HTI	
Wasser 0,7 l	II		I		
Wasser 1,0					
Sonstiges			0,50		
(Zwischen-)summe	0		5,20		

48,5 ct

DKK

56,60 €

HTI HTI

III

Bismade

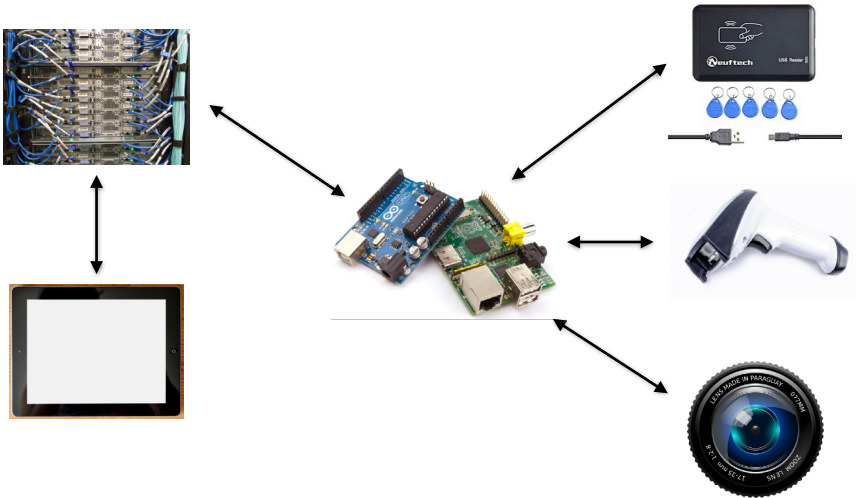
III

HTI HTI HTI HTI

Problem 2

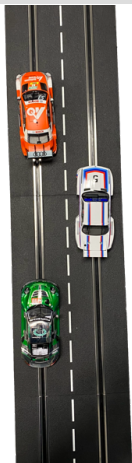


Idee

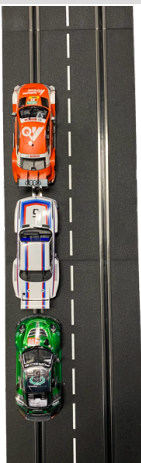


Autonomes Fahren (IBR CM)

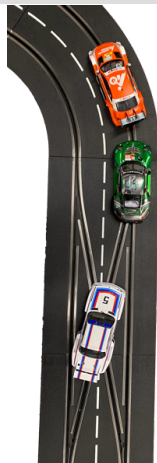
Platooning: Anreihen und Ausfahren



Menschlicher
Fahrer



Autonomes
Platoon



Einreihen
in Platoon

Autonomes Fahren (IBR CM)

Platooning: Anreihen und Ausfahren

Aufgabe:

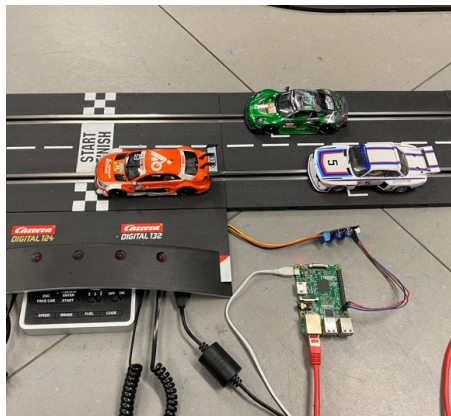
- Erzeugen & Auflösen des Platoons
- Abstandsregelung
- Geschwindigkeitsregelung

Eingesetzte Hardware:

- Kamera zur Positionsbestimmung
- Per Pi steuerbare Carrera-Bahn

Anforderungen:

- C/C++, Python
- OpenCV, ARToolKit



Autonomes Fahren (IBR CM)

KI vs Mensch

Aufgabe:

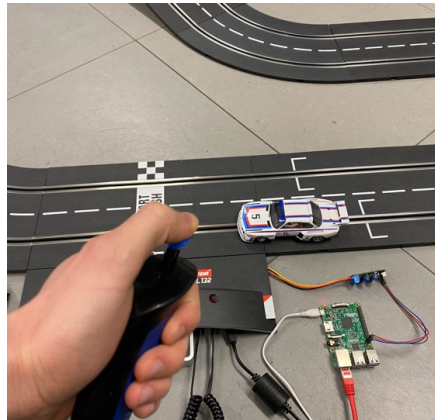
- Schneller fahren als ein Mensch ohne aus der Spur zu fliegen
- Streckenerkennung
- Spurwechsel
- Geschwindigkeitsregelung

Eingesetzte Hardware:

- Kamera zur Positionsbestimmung
- Per Pi steuerbare Carrera-Bahn

Anforderungen:

- C/C++, Python
- OpenCV, ARToolKit



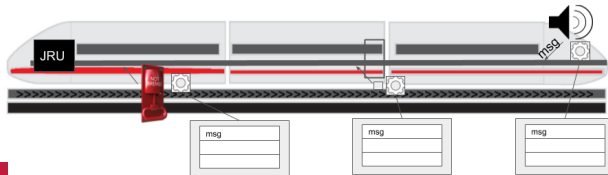
Data Logging on Trains Using Blockchain

- Train needs to be monitored for repairs, accidents
- **Trains safety procedure**
 - Central **event recorder** resilient to crash
 - Record various signals (Door signals, Brake controls..)
- **But..**
 - Risk of data manipulation
 - Risk of damage of the recorder
- **Goal**
 - A Blockchain-based solution
 - Decentralized and secure components logging



A Blockchain Train?

- Real project idea: **RailChain**
 - **Log and monitor status changes**
 - Speed
 - Diagnostic messages (Train stops, Time)
- Recorded and stored reliably in the Blockchain
- **General idea**
 1. Implement a Blockchain on a moving train
 2. Integration with Rust-based consensus
 3. Offer data as a cloud-server for Maintenance



Team Organization

- 1 Group
- 3-8 participants
- **Devices:** Sensors and Train Marklin
- Things of interest
 - C/C++ Programming
 - **Rust** Familiarity
 - Blockchain knowledge
 - **Hardware & Linux** basic knowledge
- **Kick-Off** in the first week of the semester
- Weekly meetings & code reviews
- **Presentation** in the middle and at the end of the semester





ifis

Institut für Informationssysteme
Technische Universität Braunschweig

SherloQL



Hermann Kroll

Institut für Informationssysteme
Technische Universität Braunschweig
<http://www.ifis.cs.tu-bs.de>



Technische
Universität
Braunschweig



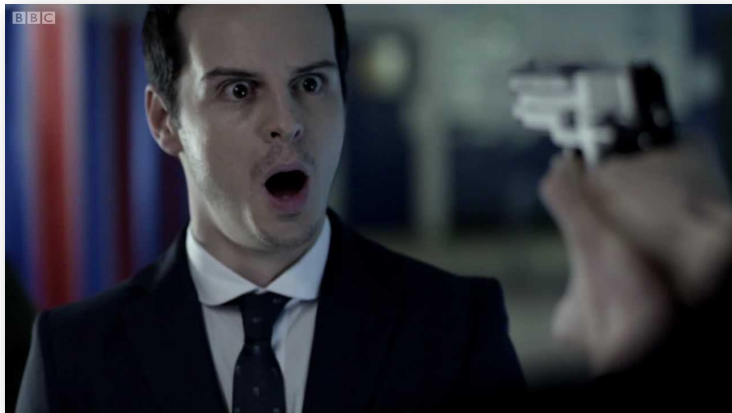
SherloQL

Wer ist Professor Moriarty?





SherloQL





SherloQL meets Alternative Reality





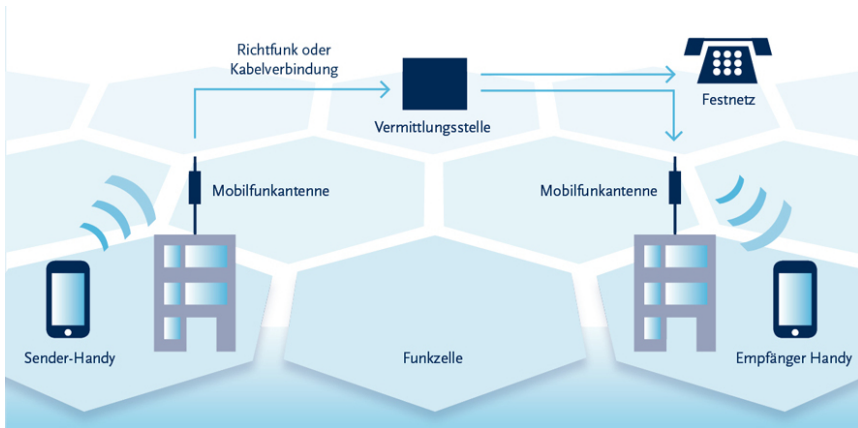
Technische
Universität
Braunschweig



Entwicklung eines Mobilfunkdemonstrators

Lennart Thielecke, Institut für Nachrichtentechnik, 06.02.2020

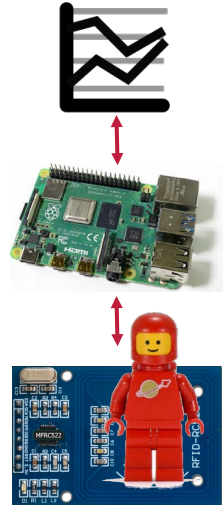
Wie funktioniert eigentlich ein Mobilfunknetz?



Quelle: informationszentrum-mobilfunk.de

Was bieten wir?

- **Ihr erhaltet Einblicke in folgende Bereiche:**
 - Funktionsweise von Mobilfunknetzen
 - Hardwarenahe Entwicklung auf embedded Systemen
 - Linux Grundkenntnisse
 - Programmieren in Python
- **Welche Hardware wird eingesetzt?**
 - Raspberry Pi zur Steuerung
 - RFID Reader zur Lokalisierung der Figuren
 - 3D-Drucker zur Gestaltung von eigenen Figuren?
- **Was solltet ihr mitbringen?**
 - Etwas handwerkliches Geschick
 - Kreativität zur Gestaltung des Spielfeldes



Quellen: heise.de und bricklink.com



Technische
Universität
Braunschweig



Softwareentwicklungspraktikum

Escape Room VR

Kamil Rosiak
07.01.07.2020

ESCAPE

R ROOM VR

Create a unique VR experience



Escape Room VR

Task:

- Develop a VR Escape Room with Unity
- Use the Oculus Quest as input device

Requirement:

- Learn how to create a game with Unity
- Implementation in C#
- Creativity, Team player

Organization:

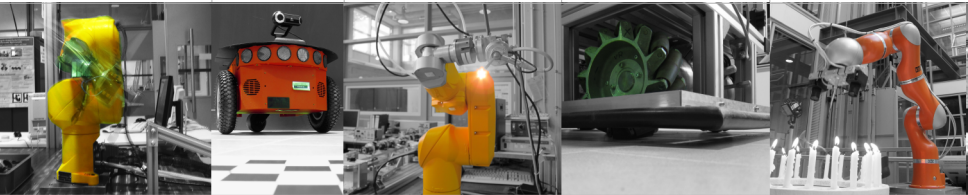
- Two teams with 6-8 students





Technische
Universität
Braunschweig

Institut für Robotik und
Prozessinformatik



Rescue Robots

Heiko Donat und Sinan Barut

Rescue Robots - Motivation



Szenario:

Rettung von Überlebenden nach einer Katastrophe (z. B. Erdbeben o. Hochwasser)

Problem:

- Bergung von Verschütteten ist zeitaufwändig und Versorgung stark eingeschränkt
- zeitnahe Erstversorgung überlebenswichtig
- gefährliche Umgebungen können Retter selbst gefährden oder schnellen Zugang unmöglich machen

Rescue Robots - Motivation



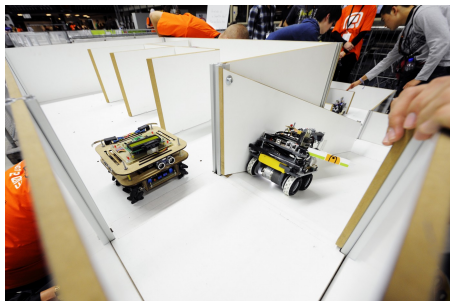
Szenario:

Rettung von Verschütteten nach einer Katastrophe (z.B. Erdbeben o. Hochwasser)

Problem:

- Bergung von Verschütteten ist zeitaufwändig und Versorgung stark eingeschränkt
- zeitnahe Erstversorgung überlebenswichtig
- gefährliche Umgebungen können Retter selbst gefährden oder schnellen Zugang unmöglich machen

Rescue Robots - Aufgabenstellung



Entwickle eine künstliche Intelligenz, die

- sich in einem Labyrinth orientiert.
- verrückbare Hindernisse verschieben kann.
- „Überlebende“ sucht und findet.

Und alles so schnell wie möglich (In einer Katastrophe zählt jede Sekunde!)

*Inspiriert von der RoboCup Rescue League

Rescue Robots - Details

Anzahl Gruppen:

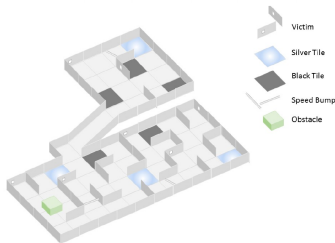
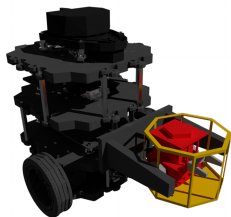
- 2 Gruppen mit 5 - 7 TeilnehmerInnen

Entwickelt wird:

- eine künstliche Intelligenz (jede Gruppe für sich)
- auf der Hardwareplattform: TurtleBot 3 + Raspberry Pi 3
- mit dem Robot Operating System (ROS)

Anforderungen:

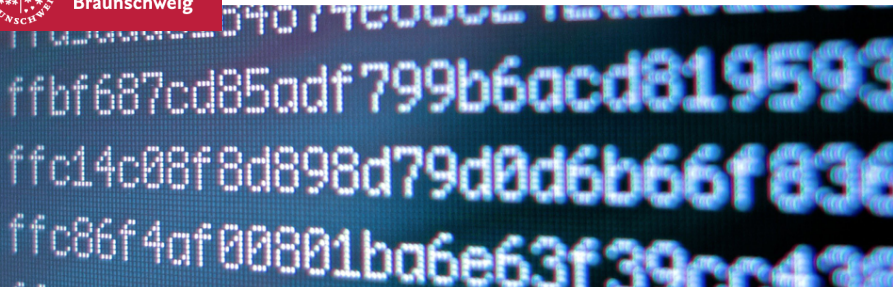
- Programmiersprachen: C++ und/oder Python
- Grundkenntnisse in Linux (Bash/Zsh)
- Teamfähigkeit und Diskussionsbereitschaft





Technische
Universität
Braunschweig

Institute of
System Security



ParkDrone

Alexander Warnecke

Problemstellung

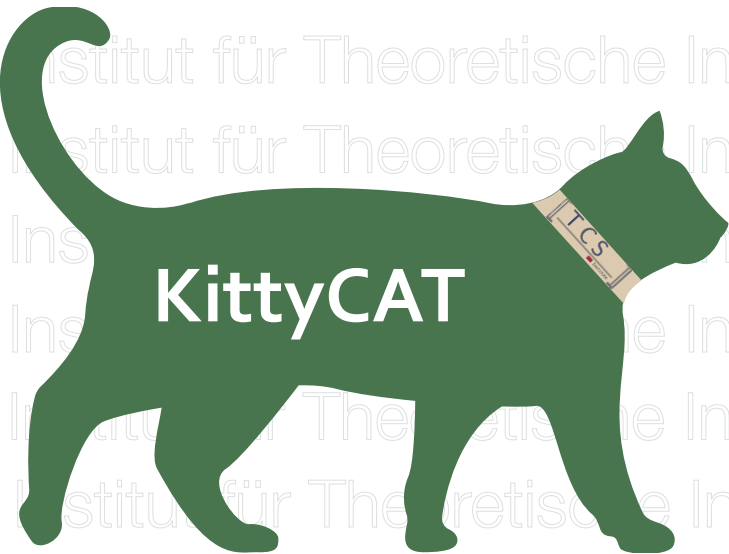
- **Wo sind wie viele Parkplätze frei?**
- **Ziel: Dynamische Auskunft durch Luftaufnahmen**
 - Parkplätze abfliegen und Foto aufnehmen
 - Automatische Auswertung der Bilder
 - Bereitstellung der Infos in App / Web-Service
 - Vorhersage mit größerer Datenbasis
- **Mögliche Erweiterungen: Fahrradparkplätze, Falschparker**



Technische Details

- **Aufnahme der Bilder**
 - Drohne hat Software für Kamera
 - Regelmäßiger Abflug der Route
- **Verarbeitung der Bilder**
 - Computer Vision Algorithmen (Kantenextraktion, Interpolation, ...)
 - Datenstrukturen zur Speicherung
- **Bereitstellung der Informationen**
 - App- / Web-Entwicklung
 - Algorithmen zur Zeitreihenanalyse

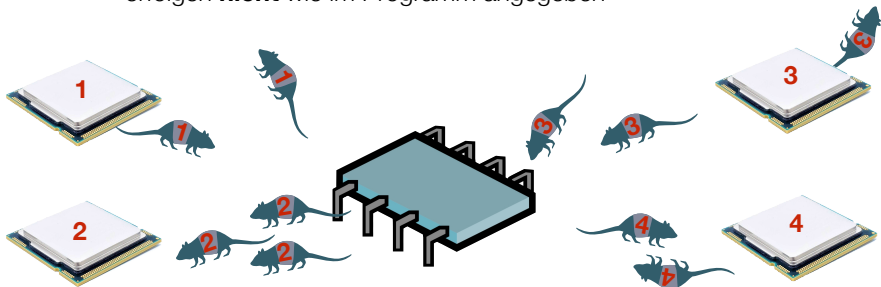








Roland Meyer
Peter Chini
Sebastian Wolff

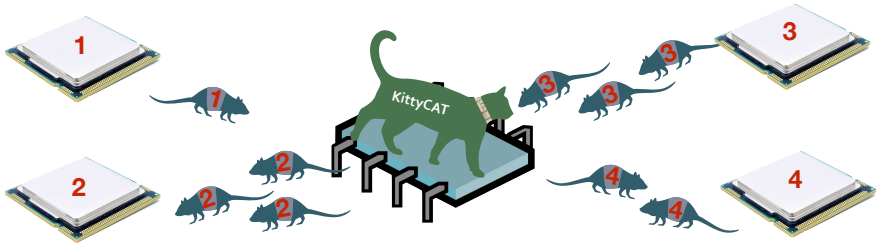
Problemstellung

- Parallele Programme \Rightarrow Synchronisation über Speicher (z.B. Locks)
- Aber: Speicheroperationen Architektur-abhängig und Performance-optimiert
 \Rightarrow erfolgen **nicht** wie im Programm angegeben



Aufgabe

- Entwicklung CPU-Simulator für beliebige --Modelle
- --Modelle sind als Formale Sprache gegeben (CAT)
- Verifikation von Assembler Programmen \Rightarrow Simulation aller Ausführungen



Institut für
Computergraphik



Massively Distributed Collaborative Crowd Input System for Dome Environments



Massively Distributed Collaborative Crowd Input System for Dome Environments

Institut für
Computergraphik



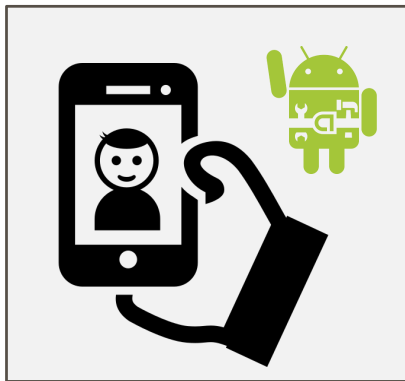
Mobile Controller Frontend

Dome Input Backend

Demo

SmartRed

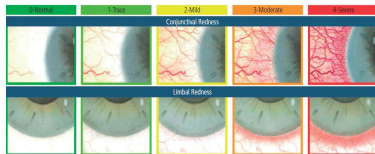
App zur Segmentierung und
Augenrötungsbestimmung



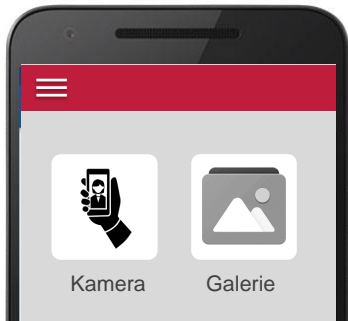
WeLineation Project
automatische Segmentierung
(Deep Learning)



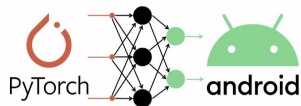
**Automatische
Augenrötungserkennung**



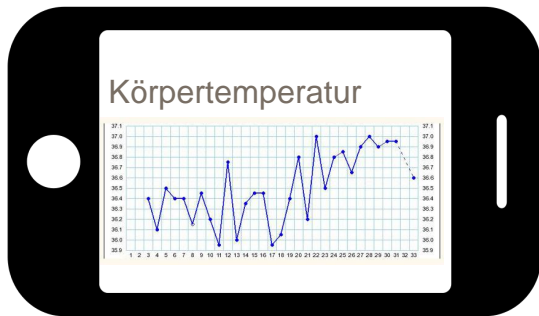
→ **Android App zur Auswertung
der Augenrötung von
Smartphone Bildern**



- **Smartphone App**
 - Android, pyTorch-Bibliothek
 - Integration vorhandener Segmentierungsalgorithmen
 - Integration vorhandener Algorithmen zur Rötungsbestimmung
 - UI/UX Design
- **Selbstständige Verlaufskontrolle der Rötung durch den „Patienten“**



Health App with FHIR Server Communication



- Entwicklung einer Android App zum Anzeigen von unterschiedlichen Sensordaten (Körpertemperatur, ...)
- Datenübermittlung von einem Bluetooth-Sensor an das Smartphone
- Alarmfunktion in der App
- Kommunikation über HL7 FHIR um die Daten auf einem Server zu speichern

