Gesture Control for Philips Hue

Projekt in Mobile und ubiquitäre Systeme (MUS2)

Software Engineering (Master)

FH Hagenberg

Sommersemester 2015

Florentina Grebe (S1410454005)

Sabine Winkler (S1410454017)

**Inhaltsverzeichnis**

[1. Problembeschreibung 3](#_Toc423900516)

[2. Lösungsansatz 4](#_Toc423900517)

[2.1 Lösungsidee 4](#_Toc423900518)

[2.2 Verwendete Hardware 4](#_Toc423900519)

[2.3 Verwendete Technologien und Frameworks 4](#_Toc423900520)

[3. Ergebnisse 6](#_Toc423900521)

# Problembeschreibung

# Lösungsansatz

## Lösungsidee

## Verwendete Hardware

Es wurden folgende Hardware-Komponenten verwendet:

* StarterKit von Philips Hue.  
  Dieses inkludiert
  + 3 x 9W A60 E27 Lampen
  + 1 x hue-Bridge
  + 1 x Netzadapter
  + 1 x Ethernet-Kabel.

Siehe <http://www.hue.philips.de/>, <http://www2.meethue.com/de-at/produkte/hue/> und <http://www.amazon.de/Philips-hue-personal-wireless-lighting/dp/B00IG18C9G/?_encoding=UTF8&camp=1638&creative=6742&keywords=B00IG18C9G&linkCode=ur2&qid=1415618007&site-redirect=de&sr=8-1&tag=httphiiprodef-21&linkId=JHE4RGYA7EF6N27H>.

* Microsoft Kinect 1 mit Xbox 360 Sensor.  
  Diese konnte mit freundlicher Genehmigung von der FH Hagenberg entliehen werden.
* 3 x Kabel mit Lampenfassung E27  
  Siehe <http://www.amazon.de/Kabel-Lampenfassung-wei%C3%9F-L%C3%A4nge-Schalter/dp/B008NAFGU8/ref=sr_1_6?ie=UTF8&qid=1430079083&sr=8-6&keywords=lampenfassung>.

## Verwendete Technologien und Frameworks

Es wurden folgende Technologien und Frameworks eingesetzt:

* .NET-Framework 4.5
* Microsoft C# und WPF für die GUI
* Kinect for Windows SDK 1.8  
  <https://www.microsoft.com/en-us/download/details.aspx?id=40278>
* Kinect for Windows Developer Toolkit v1.8.0  
  <https://www.microsoft.com/en-us/download/details.aspx?id=40276>
* Bibliotheken *GestureFabric* und *KinectUtils* für die Gestenerkennung mittels Microsoft Kinect.  
  Diese wurden mit freundlicher Genehmigung von der FH Hagenberg zur Verfügung gestellt.
* Microsoft Speech API (SAPI) 5.4 für die Spracherkennung, insbesondere der Namespace System.Speech.Recognition.
* Philips Hue API.  
  Siehe <http://www.developers.meethue.com/philips-hue-api>.  
  Dieses API bietet einen Zugriff auf das hue-System mittels RESTful Webservices over HTTP und JSON.  
  Für einen Zugriff auf die vollständige Dokumentation dieser API ist eine Registrierung bei Philips Hue notwendig.
* Q42.HueApi.  
  Siehe <https://github.com/Q42/Q42.HueApi>

Diese Bibliothek bietet einen einfachen Zugriff auf die Philips Hue API in C# und kapselt die RESTful-Webservice-Aufrufe.  
Q42.HueApi kann mittels NuGet-Package-Manager installiert werden.

* Microsoft Visual Studio 2013

Zu beachten ist, dass zuerst *Kinect for Windows SDK* und danach erst *Kinect for Windows Developer Toolkit* installiert werden darf. Zudem ist wichtig zu wissen, dass für die Microsoft Kinect 1 ein *Kinect for Windows SDK 1.8* erforderlich ist. Das SDK in Version 2 funktioniert für Microsoft Kinect 1 nicht.

# Ergebnisse