

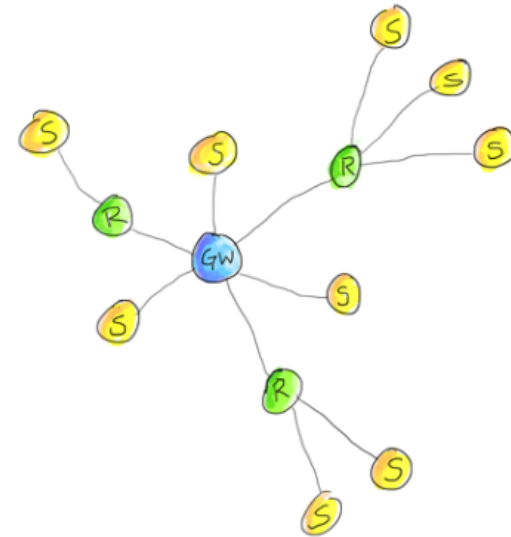
Bachelor Projekt Systementwicklung

Smart Home Energy Management

Peter Altenbernd, Sommersemester 2023

Technologische Vorgaben

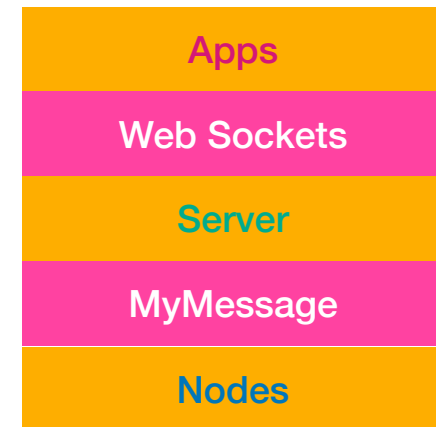
- Qt mit C++ für Nodes, Server, Apps
- Nodes: **Arduino Nano**
 - Sensoren+Schalter (S)
 - Repeater (R)
- Server: **Raspberry Pi**
 - Gateway (GW)
 - **Smart Home Server**: Kern des Projekts!
 - Webserver
- **App(s)**: Plattformen: Desktop, iOS, Android, Webassembly auf Basis von Qt
- Basis: **MySensors** <https://www.mysensors.org/>



Projekt-Struktur

3 Teams, je ungefähr gleich groß

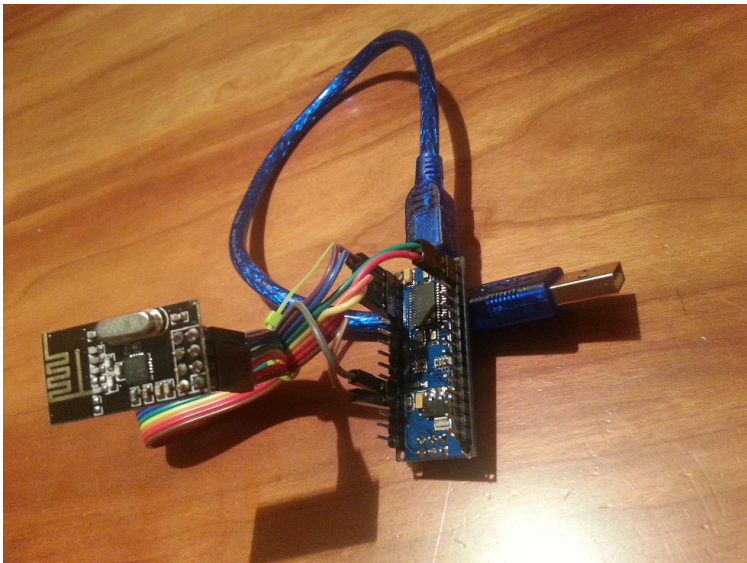
- **Apps**: Endgeräte
 - Display
 - Interaktion
 - Administration
 - Schnittstelle zum Smart Home Server
- **Server**: auf Raspberry Pi
 - Betrieb MySensors Gateway
 - Smart Home Server
 - * Power Management (Kern des Projekts)
 - * Schnittstellen zu allen Beteiligten
 - Webserver
 - Projektleitung und System-Integration
- **Nodes**
 - Hardware
 - Betrieb MySensors Basis SW
 - Erweiterungen der Basis SW
 - MyMessage Schnittstelle zum Server
 - Lokales WLAN



Team Nodes

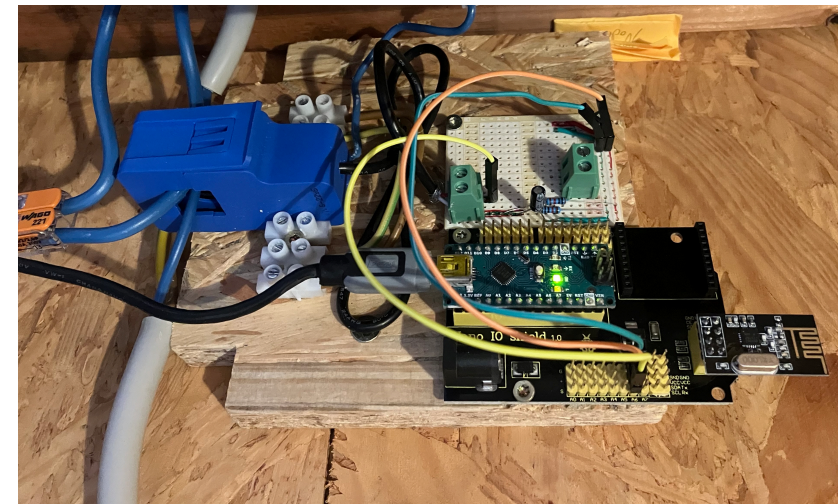
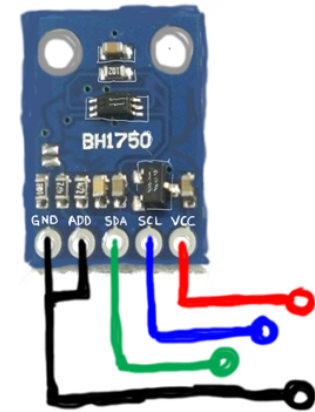
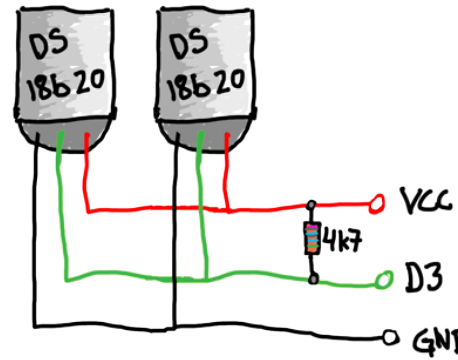
Die Mitglieder sollten *Spaß am Basteln* haben.

- **Repeater** bzw. Grundausstattung
 - Das Funknetz ist nie stabil: Repeater machen nichts anderes als Signale weiterleiten
 - Jeder weitere Knoten ist zugleich Repeater
 - Stromversorgung: integriert oder extern



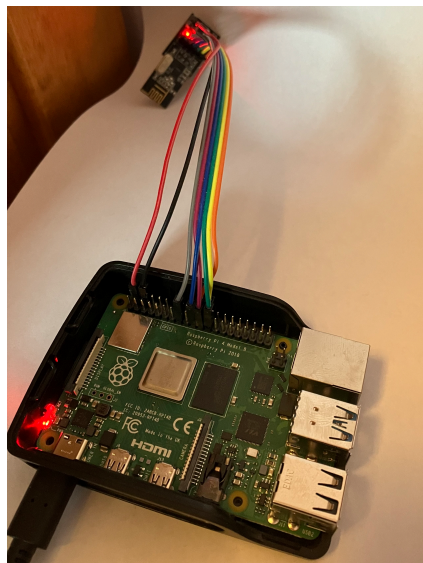
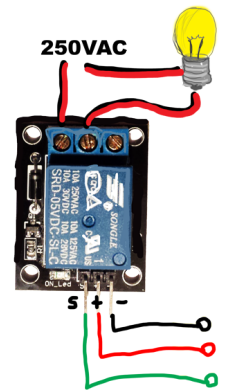
Team Nodes

- Sensoren
 - Licht: Simuliert die PV-Anlage: LUX wird konvertiert in KW
 - Temperatur: Signalisiert den Bedarf
 - Power Meter: misst den Strom von Verbrauchern



Team Nodes

- **Switch (Actuator, Schalter)**
 - Schaltet Verbraucher
 - Achtung: Der Strom kann sehr stark sein!
 - Manche Heizungen haben mehrere Stufen
 - Anzeige Status
 - 2 Varianten:
 - * Heizung: temperaturabhängig via Server
 - * Heißwasser: bedarfsabhängig via Server
- **Raspberry Pi (HW)**



Team Nodes

- **Software**
 - Die MySensors Basis Software muss erweitert werden
 - Erweiterungen des Netzwerks müssen einfach möglich sein
 - Alle Sensoren und Schalter müssen auf einem Knoten miteinander kombinierbar sein (eindeutige Sensor IDs, zugeordnet zu Pins)
 - Zu jedem Sensor und zu jedem Knoten braucht es zugehörigen Code auf dem Server
 - Schnittstelle zum **Server**: MyMessage
- **Aussehen**
 - Die Knoten müssen Wohnzimmer-tauglich sein
 - Seit Kurzem gibt es am Fachbereich einen Makerspace - seien Sie kreativ!

Team Server

Die Mitglieder sollten *Sinn fürs große Ganze* haben und gut *programmieren* können.

- **Installationen**
 - MySensors Gateway
 - Webserver
- **Projekt-Leitung und System-Integration**
- **Schnittstellen**
 - Gemeinsam mit **Team Nodes**:
 - * MyMessage inkl. Erweiterungen
 - * Läuft via Gateway unter TCP
 - Gemeinsam mit **Team Apps**:
 - * Web-Sockets in selbst definierten Format
 - * JSON

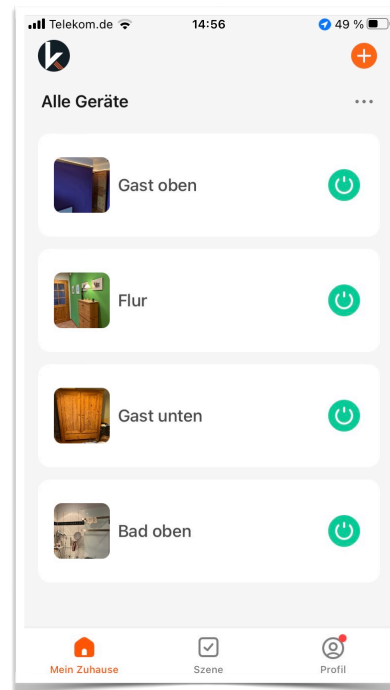
Team Server

- **Software-Entwicklung: Smart Home Server**
 - Abfragen oder Entgegennehmen der aktuellen Sensor-Werte
 - Umsetzung der App-Funktionalität
 - Spiegelung der Netz-Infrastruktur
 - Kopplung von Temperatur-Sensoren mit Schaltern
 - Power-Management: Bin-Packing Algorithmus
 - Fehlerkontrolle und -toleranz
 - Statistik
 - ...

Team Apps

Die Mitglieder sollten *kreativ sein*.

- **App-Entwicklung**
 - für alle Plattformen (s.o.)
 - Spiegelung der Netz-Infrastruktur
 - Schnittstelle zum **Server**: Web-Sockets + JSON
 - Wichtig: **Usability** und **Design**



Team Apps

- **Funktionalität**
 - **Anzeigen** von Sensor-Werten (inkl. Schaltern), aktuellem Verbrauch, Statistik
 - **Interaktion** mit Schaltern
 - **Modi**:
 - * **OFF**: im Sommer oder bei Abwesenheit
 - * **ON**: Dauer-an: Wein vorm Kamin
 - * **Zeitplan** mit Temperaturen (*MinTemp*)
 - * **Auto**: geht nur an, wenn genug Energie vorhanden bin *MaxTemp*
 - * **Smart**: Kombination aus *Zeitplan* und *Auto*
 - * ...
 - **Einstellung** von Zeiten, Temperaturen (*MinTemp*, *MaxTemp*) usw.
 - Dynamisches Hinzufügen / Entfernen von Knoten (wenn noch Zeit)

Organisation

- **Aufwand:** 7,5 CP = 10h/Woche
 - der Termin im Stundenplan ist i.W. nur zur Organisation
 - Fertigen Sie sich Stundenzettel an: Sie brauchen sie für den Abschluss-Bericht
- **Ablauf der Treffen**
 - Berichte der 3 Gruppen
 - Klärung von Fragen
 - Ausblick auf die nächste Woche

Organisation

- **Ablauf des Semesters**
 - Woche 1 (heute): **Kick-off**
 - Woche 2: *siehe nächste Seite*
 - Wochen 3-5: Arbeit
 - Woche 6: **Mid-Term Workshop** mit Prototypen
 - Wochen 7-12: Arbeit
 - Woche 13: **Abschluss-Workshop** mit Präsentation
 - Prüfungswochen: Abgabe der **Berichte** (je Gruppe): danach Notenvergabe
- **Benotung:** Auf Basis der Präsentationen und der Berichte

Organisation

- **Heute**
 - Gruppen-Bildung
 - Projektleitung (ggf. bis nächste Woche)
 - Namensfindung
 - Terminplanung: Wann wollen Sie sich außerhalb vom Stundenplan regelmäßig treffen?
 - Ziel-Definition für die kommende Woche
- **Voraussichtliche Ziele zur kommenden Woche:**
 - **Team Nodes:** MySensor, HW-Auswahl und -Bestellung, Arduino IDE
 - **Team Server:** Grobe SW-Architektur, Git, Qt Creator
 - **Team Apps:** Recherche existierender Apps, Grob-Konzept, Qt Creator