University College Nordjylland

4. Semester Projekt

UVU-Bot Projekt

*Forfatter:*

Dennis Rasmussen

*Vejledere:*

Mogens Holm Iversen

Ib Helmer Nielsen

Antal anslag:  
xxxxx

17. december 2018

**University College Nordjylland  
Teknologi og Business**Datamatiker  
Dmaa0217  
UVU-Bot projekt

**Projektdeltager:**Dennis Rasmussen

**Vejledere:**Ib Helmer Nielsen  
Mogens Holm Iversen

**Afleveringsdato:**17. december 2018

**Repository placering:**<https://github.com/Derasm/4Semester>

# Abstract:

Denne rapport tager udgangspunkt I problemformuleringen  
*Hvordan kan en robot designes og programmeres, med fokus på valg af teknologi og robusthed i arkitekturen, til at kunne styres fra en telefon.*Rapporten undersøger processen i valg af teknologi og udarbejdelse af arkitektur, de tankemæssige begrundelser der er gjort for valgene samt hvordan den potentielle robot der bliver udarbejdet kan bruges. Løsningen på problemformuleringen blev en undersøgelse af mulighederne indenfor Internet of Things apparater, og deres programmeringsmuligheder(programmeringssprog), herunder applikationsmuligheder og programmeringssprog.   
Der blev i første omgang valgt at fortsætte med Raspberry Pi da denne var den kraftigste af mulighederne. Ud fra dette valg blev udviklingsmulighederne undersøgt, herunder arkitektur, fysiske opkoblingsmuligheder og udviklingssprog. Undersøgelsen af Raspberrien viste at udviklingsprocessen ville blive for besværlig med for mange nye teknologier der skulle tages højde for, hvilket ville tage mere end den mulige tid i projektet. Der blev derefter valgt at skifte til en Arduino Uno da udviklingen af denne platform var markant simplere, med næsten de samme fysiske muligheder. Arkitekturen på Uno’en viste sig at være anderledes, da standard arkitektur ikke ville passe på et program hvor et User Interface ikke er til stede. En software arkitektur blev valgt hvor den tredelte arkitektur stadig kunne bruges, dog med ændrede navne og funktioner af de enkelte lag. Dette sikrede en lav kobling mellem klasserne samt en høj samhørighed på både metode- og klasseniveau.

Konklusionen på projektet blev en robot der kan styres fra telefonen, designet med god kodeskik i fokus som kan skaleres og eventuelt overføres til større og kraftigere kontrolenheder.

Contents

[Abstract: 3](#_Toc530665582)

[Hovedafsnit 5](#_Toc530665583)

[Problemanalyse 5](#_Toc530665584)

[Problemfelt 5](#_Toc530665585)

[Problemformulering 5](#_Toc530665586)

[Proces 5](#_Toc530665587)

[Teknologi 5](#_Toc530665588)

[Arkitektur 5](#_Toc530665589)

[Implementering 5](#_Toc530665590)

[Refleksion 5](#_Toc530665591)

[Konklusion 5](#_Toc530665592)

[Litteraturliste (inkl. Alle kilder der er lavet henvisninger til) 5](#_Toc530665593)

[Bilag 5](#_Toc530665594)

# Hovedafsnit

## Problemanalyse

Indledning

Robotter er i dag at finde overalt omkring os. Alt fra selvkørende biler til samlebånd og legetøj. Omkostningerne og processen at lave en robot var før utroligt høje. De er dog faldet drastisk efter hardware som Aruino og Raspberry er blevet mere udbredt. Denne rapport vil undersøge hvordan en robot kan planlægges, designes og implementeres til at udføre en handling bestemt af brugeren via telefon.

## Problemfelt

Emnet for projektet er en robot der skal bruges som proof of concept til at vise mulighederne for programmering af robotten både hardwaremæssigt, samt sammenkobling af software til hardwaren.

## Problemformulering

*Hvordan kan en robot designes og programmeres, med fokus på valg af teknologi og robusthed i arkitekturen, til at kunne styres fra en telefon.*

* Hvad hardware komponent ville være mest realistisk at bruge i den tilladte tid?
* Hvordan kan arkitekturen i robotten designes til at holde kobling lav og samhørighed høj.
* Hvordan kan man forbinde robotten til en telefon, og hvilken teknologi vil være mest passende ( Wifi, bluetooth osv).

Hvordan kan

## Proces

Det første skridt i projektet var at vurdere hvilken onboard computer der skulle bruges som kontrolenhed. Dette blev gjort ved at sammenligne de to muligheder; Raspberry Pi 3B og Arduino Uno.

|  |  |  |
| --- | --- | --- |
|  | Raspberry Pi 3 | Arduino Uno |
| Programmeringssprog | C++, Python, Java, med flere. | C++, C, Python. |
| Kompleksitet | * *Styresystem*   Linux   * *Opsætning*   Middel besværlighed   * *Overførsel af filer*   USB kabel og pind   * *Test og debugging*   Test og debugging gøres seperat | * *Styresystem*   unødvendigt   * *Opsætning*   Er gjort   * *Overførsel af filer*   Automatisk upload af filer   * *Test og debugging*   Indbygget test og debugging. |
| udvidelsesmuligheder | I/O pin’s til komponenter, USB, HDMI og andre forbindelsesmuligheder | I/O pins til komponenter. Mulighed for påsættelse af Shields for forsimplet udvidelse. |

## Teknologi

## Arkitektur

## Implementering

## Refleksion

# Konklusion

# Litteraturliste (inkl. Alle kilder der er lavet henvisninger til)

https://www.merriam-webster.com/dictionary/robot

# Bilag

Angivelse af sti til repo.