

A-Holdet

3. Semester projekt

RO DA 18E 3C

5. Nov - 17. Dec

Joachim Frank

Anders Skov Nielsen

Frederik Sthen Hansen

[**1: Introduktion**](#_rwi7pd1znxo8) **4**

[**2: Problemformulering**](#_j094v1ncq9o) **4**

[**3: Projektetablering**](#_38u0g7hd7qfe) **5**

[3.1: Deltagere](#_r3vbsgccem7d) 5

[3.2: Kontakt information](#_adfcrspn57h3) 5

[3.3: Tidsplan](#_j2gt53452x0k) 6

[3.4: Gruppe kontrakt](#_xzw3odhyood4) 6

[3.4.1: Ambitionsniveau](#_3yair0xbdpws) 6

[3.4.2: Sygdom og fravær](#_du9wq5bcsvc4) 7

[3.4.3: Konflikt](#_ow569ovawmgr) 7

[3.4.4: Mødetid](#_yz51vpy2e560) 7

[3.4.5: Hjemmearbejde](#_7av93ylr64t0) 8

[3.5: Krav til hardware og software](#_b9cr6skp4jg2) 8

[3.5.1: Hardware](#_cckr0fqyoped) 8

[3.5.2: Software](#_ikd2sb9iugzs) 8

[3.6: Risiko](#_g2zu7aw89kvh) 9

[**4 Projekt Manifest**](#_fav0h0xozw9c) **10**

[4.1: SCRUM](#_80r2pz2qoyvn) 11

[4.1.1: Roller](#_qv1af4yxl78s) 11

[4.1.2: Planning Game](#_eo3rjkydkara) 11

[4.1.3: Daily SCRUM](#_gqu6frgr144p) 11

[4.1.4: Sprint review:](#_b3bte03u3xeu) 11

[4.1.5: Sprint Retrospective:](#_osc5cyrw8my) 12

[4.1.6: User Stories](#_qzqyxcc2vvyw) 12

[4.1.7: Kanban Board, ved brug af Trello](#_k8qwfin6k6rx) 12

[4.1.8: Planning Poker](#_4xcdul32rc8p) 12

[4.2: XP](#_ha10nd9fmhp2) 13

[4.2.1: Pair Programming](#_cvrz4hiidxua) 13

[4.2.2: Test Driven Development](#_8zha2z5lotti) 13

[4.2.3: Sustainable Pace](#_q6c3cm95up82) 13

[4.2.4: Collective Code Ownership](#_wl0nlpblv4jx) 14

[4.3: UP](#_jtb2a0oi980b) 14

[4.3.1: UML](#_s9jj0k89dj1u) 14

[**5: Sprint 0 / Inception Deck**](#_z85k67zc8q5t) **15**

[5.1: Idégenerering](#_7jdk1pf0yhgt) 15

[5.2: User Stories:](#_kl7rvfns4nqq) 15

[5.2.1: Acceptance Criteria](#_d19oa31bpdt9) 16

[5.3: Elevator Pitch](#_gymyiwe7iib6) 18

[5.4: Product Box:](#_2echos5g2hhs) 18

[5.5: Product Backlog:  
Ansvarlig: Alle](#_641t36d6qsl2) 20

[5.6: Visual WorkSpace](#_287trrfdw4g1) 21

[5.7: Arkitektur overblik via domænemodel](#_ix4agrvzirv9) 21

[5.8: Sprint retrospective:](#_g21hyu31z2tg) 22

[**6: Sprint 1**](#_ubzi3cm5xlcq) **22**

[6.1: Sprint backlog](#_4ifgw7sdj7nr) 22

[6.2: Daily Scrum](#_p61yvqy9kkr7) 23

[6.2.1: Nov. 26](#_okt8id1ydag2) 23

[5.2.2: Nov. 28](#_ybfqjv60kvci) 23

[6.2.3: Nov. 29](#_9jqs3a6k10f5) 23

[6.2.4: Nov. 30](#_yhczbe4agd6m) 24

[6.3: Sprint Review](#_pgqr8ty3v68r) 24

[6.4: Sprint retrospective](#_3rpot21ltskq) 26

[6.4.1: Test Driven Development](#_ohekdi4x5tc8) 26

[6.4.2: Pair Programming](#_f8kn81lj3a8k) 27

[6.4.3: Review og Retrospective](#_md8sbainw46c) 27

[6.4.4: Daily SCRUM](#_b02b37fjtq2i) 28

[6.4.5: Kanban Board](#_24a4urudfhpp) 28

[6.4.6: Sustainable Pace](#_6lqonwaxgdl3) 29

[**7: Sprint 2**](#_rd97svhkbtzw) **30**

[7.1: Sprint backlog](#_85s44m59ksc1) 30

[7.2: Daily Scrum](#_gschl9uruqth) 30

[7.2.1: Dec. 3](#_lmx296mh7zr3) 30

[7.2.2: Dec. 5](#_3bsvvr8tueib) 30

[7.2.3: Dec. 6](#_ua7h3nr0qqxn) 30

[7.2.4: Dec. 7](#_hdhljwgv2ij7) 30

[7.3: Sprint Review](#_rgbxvcns943t) 31

[7.4: Sprint Retrospective](#_bdefgf9kpy8t) 31

[**8: Sprint 3**](#_5zkz9smp1ys4) **31**

[8.1: Sprint backlog](#_vd4l83judqkr) 31

[8.2: Daily Scrum](#_77we93no03b8) 31

[8.2.1: Dec. 10](#_xr42q3xw04sx) 31

[8.2.2: Dec. 11](#_2bn8nek3m459) 31

[8.2.3: Dec. 12](#_g0dg8e43pwdx) 31

[8.2.4: Dec 13](#_da9fofvvhcpg) 31

[8.2.5: Dec 14](#_782cw2pcr3vv) 32

[8.3: Sprint Review](#_q1cgzw7sid72) 32

[8.4: Sprint Retrospective](#_pn9sgg6p4biw) 32

[**9: Konklusion**](#_w50ao5p8xbpd) **32**

# 1: Introduktion

***Ansvarlig: Anders***

Link til trello <https://trello.com/b/NsBX1Bzi/lms-project>

I denne rapport vil vi beskrive vores arbejde med 3. Semester eksamensprojekt på Datamatiker uddannelsen, hvor vi skal afprøve en arbejdsprocess med SCRUM og XP i modsætning til det tidligere semester hvor vi arbejdede med UP.

I løbet af dette projekt kommer vi også til at arbejde med at opstille et system, der passer til vores givne opgave, men vores rapport vil stærkt fokusere på arbejdsprocessen og vores erfaringer med de metoder og artefakter, som vi har udvalgt fra SCRUM og XP.

Vores Raspberry-Pi kommer vi til at referer til som “gyro-sensor”, den man har på hånden, og sensoren der tænder lyset bliver betegnet som “Bevægelsessensor”, den der sidder i væggen og tænder lyset.

# 2: Problemformulering

***Ansvarlig:***

I lærerværelset på Erhvervsakademi Sjælland (EASJ) i Roskilde har de en bevægelsessensor, der slukker lyset når der er mangel på input. Lærernes arbejde er stillesiddende og giver derfor ikke input til sensoren, hvorved lyset slukker og lærerne skal afbryde deres arbejde for at bevæge sig og dermed give input til sensoren.

Kunden ønsker at kunne arbejde uafbrudt uden at lyset slukkes. Et eksempel på en løsning fra kunden selv består i at programmere en “robot arm” til at lave den nødvendige bevægelse for at sensoren tænder lyset igen.

Ud fra dette problem har vi derfor stillet spørgsmålene:

*“Hvordan kan vi med SCRUM som arbejdsprocess udvikle et system til lærerværelset på EASJ, så lærerne ikke behøver at rejse sig når lyset slukker?”*

*“Hvordan kan vi inkorporere en 3. Parts API i vores system?”*

# 3: Projektetablering

## 3.1: Deltagere

***Ansvarlig: Alle***

|  |  |  |  |
| --- | --- | --- | --- |
| **Navn** | **Rolle** | **Stærke kompetencer** | **Svage kompetencer** |
| Joachim Frank | SCRUM Team | GUI. åben for ændringer. | Kan overse detaljer. Arbejde. |
| Anders Skov Nielsen | SCRUM Team,  Kontaktperson. | Fokus på aktiv opgave.  Velskrivende. | Minimalt arbejde hjemmefra.  Dårlig udmelding af problemer. |
| Frederik Sthen Hansen | Scrum Master SCRUM Team | Kreativitet i design, | Taber overblikket, stresser, minimalt hjemmearbejde |

## 3.2: Kontakt information

***Ansvarlig: Alle***

|  |  |  |
| --- | --- | --- |
| Navn | Telefon | Email |
| Joachim Frank | 53 44 76 00 | Jgf.easj@gmail.com |
| Anders Skov Nielsen | 61 33 39 95 | anderssnielsen@live.com |
| Frederik Sthen Hansen | 23 66 59 02 | Frederik.Sthen.Hansen@gmail.com |

## 3.3: Tidsplan

***Ansvarlig: Anders + Frederik***

|  |  |  |
| --- | --- | --- |
| Fase | Estimeret Tid | Beskrivelse |
| Projekt Etablering | til 9-11-2018. Mandag. | Projekt-etablering/manifest |
| Inception Deck / Sprint 0 | Til d. 22 november | Product Box, Elevator Pitch, User Stories, Product Backlog |
| Sprint 1 – 3 | 3 uger (d. 25 nov - 13 dec) | TDD, Pair-Programming, Rapportskrivning |
| Rapportfinalisering | 1 dag til og med 17/12 | Endelig gennemgang af Rapport, korrektur, omskrivning af afsnit |

## 3.4: Gruppe kontrakt

### 3.4.1: Ambitionsniveau

***Ansvarlig: Alle***

Vi er alle generelt tilfredse med at vi kan bestå vores eksamen, men vil arbejde på at vi i det mindste kan få et 7-tal, mens højere karakterer selvfølgelig stadig er værd at stræbe efter, men vil ellers primært være en bonus.

### 3.4.2: Sygdom og fravær

***Ansvarlig: Alle***

I tilfælde af sygdom eller generelt fravær skal der sendes en besked til holdet via en fælles gruppe på Discord chat appen. Denne besked behøver ikke umiddelbart nogen grund, medmindre man forventer at være fraværende i længere tid.

Når man er fraværende forventes det at man læser rapporten igennem for ændringer dagligt for at holder sig opdateret.

Hvis man er fraværende i længere tid bedes man deltage i et par timer om dagen via Discord chatten, og at man holder sig opdateret på teorien bag arbejdet så man ikke falder bagud og glemmer hvad vi arbejder på.

Skulle det ske at kontaktpersonen selv er fraværende, skal vedkommende skrive til dagens vejleder, eller de resterende team deltagere, især hvis der ikke er en vejleder til stede på den givne dag.

### 3.4.3: Konflikt

***Ansvarlig: Alle***

I tilfælde af indbyrdes konflikt vil det givne emne først og fremmest blive taget op til diskussion i håb om at der kan findes frem til en løsning. Hvis dette viser sig ikke at være tilfældet, så vil der afholdes en afstemning hvor flertal udvælges.

### 3.4.4: Mødetid

***Ansvarlig: Alle***

Når der er timer på skemaet, så forventes det at man møder op, så mest muligt arbejde kan finde sted, samt at der er bedst mulighed for møder og fælles diskussion.

### 3.4.5: Hjemmearbejde

***Ansvarlig: Alle***

Hvis der er behov for læsning inden for et vist emne, kan dette blive allokeret til hjemmearbejde.

Derudover er hjemmearbejde ikke obligatorisk, men anmodes i tilfælde af længerevarende fravær. Hjemmearbejde kan også blive nødvendigt, hvis der ikke bliver lavet nok fremskridt i løbet af den almene mødetid ved skemaet.

## 3.5: Krav til hardware og software

### 3.5.1: Hardware

***Ansvarlig: Frederik***

Hver deltagere i teamet skal have deres egen computer at arbejde på.

En Raspberry PI med SenseHat bruges til at simulere en gyroskopisk sensor man bærer på håndryg eller håndled, der skal registrere lærernes bevægelser når de arbejder normalt. Denne Pi vil fremover være benævnt “Gyrosensor”. Dette er for at differentiere den fra ordet “bevægelsessensor”, der refererer til den indbyggede bevægelsessensor i lærerværelset.

### 3.5.2: Software

***Ansvarlig: Alle***

Rapporten vil primært blive skrevet i et fælles Google Docs dokument og Github vil bruges til at opbevare vores program og andre nødvendige filer.

For at kunne organisere vores opgaver benytter vi os af et digitalt Kanban Board på hjemmesiden Trello.com.

Visual Studio skal være opdateret til version 15.9.2 og i løbet af projektet må der ikke udføres opdateringer, for at reducere problemer med forskelligt software.

Visual Studio Code skal være opdateret til version 1.28.2, som anvendes til Typescript applikation og i løbet af projektet må der ikke udføres opdateringer, for at reducere problemer med forskelligt software.

I tilfælde af fildeling vil vi gøre brug af Github Desktop, som for alle skal være opdateret til version 1.5.0, for at undgå eventuelle konflikter mellem differentierende software.

Når der skal laves test af REST API uden for Unit Testing, vil vi gøre brug af Postman da flertallet har bedre erfaring med denne software i forhold til Fiddler, et lignende program.

Til opbevaring af vores databaser og web apps, som fx REST API og hjemmeside gennem HTML og TypeScript, vil vi gøre brug af Microsofts Azure, som er en cloud service, der gør det muligt at anvende vores programmer over nettet.

I tilfælde af fjern-møder vil der gøres brug af Discord, en gratis chat app. DIscord behøves ikke opdateres til en standardiseret version.

Som 3. parts API benytter vi <http://worldclockapi.com/>. Herfra vil vi hente klokkeslæt, dato, ugedag og tidszone.

## 3.6: Risiko

***Ansvarlig: Alle***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Niveau** | **Beskrivelse** | **Betydning** | **Indikator** | **Beredskabsplan** |
| 2 | Tab af link til docs. | H | Mistet adgang til Docs | Kontakt andre medlemmer for nyt link. |
| 7 | Længere sygdom. | H | Tilstedeværelse | Kontakt og reorganisation. |
| 4 | Aftagende energiniveau i mødetid | M | Lav fokus og træthed | Lad vedkommende tage en pause eller hjem for ikke at risikere fravær i følgende dage |
| 6 | Ukendt emne (Teknologi) | H | Kodning kører fast | Led efter løsning i C#-note. Konsultér andre gruppemedlemmer, og derefter læreren hvis problemet stadig ikke har løst sig. |
| 5 | Ukendt emne (Python Raspberry pi sensorer) | H | Vi har kun basis erfaring inden for emnet. | Diskuter designløsning i gruppen for at sikre at alle har samme forståelse af emnet. |
| 4 | Tekniske problemer med Visual Studio | H | Uforståelige fejl i kode | Kontakt vejleder |

# 4 Projekt Manifest

***Ansvarlig: Anders***

Vi vil her gøre rede for hvilke praktiker og artefakter fra SCRUM og XP som vi vil tage i brug i løbet af vores projektforløb og de individuelle sprint. Dertil vil vi også tage enkelte værktøjer fra UP i brug.

## 4.1: SCRUM

### 4.1.1: Roller

***Ansvarlig: Frederik + Anders***

Vores Product Owner består i vores vejledere, lærerne på EASJ.

Efter et frivilligt valg har Frederik meldt sig som vores SCRUM Master i starten af vores Sprint 0, mens at resten af holdet er det generelle SCRUM Team.

Dog har vi op til 1. Sprint ændret SCRUM Master til at være Anders, da Frederik har meget at se til uden for projektet i forb. med flytning til Roskilde. Han vil muligvis genoptage rollen hen imod slutningen af projektet, såfremt der bliver overskud til dette. Denne ændring er også blevet taget på delvis basis af at rotere rollen, så alle kunne prøve at være ansvarlig for opdateringer og kundekontakt med vores Product Owner.

### 4.1.2: Planning Game

***Ansvarlig: Anders***

For at få et overblik over arbejdet inden for de individuelle sprint vil vi gøre brug af et overordnet Planning Game, hvor vi ser på vores givne user stories og den udregnede velocity og derfra udpeger hvilke user stories vi vil tage fat på per sprint.

Dette vil vi primært gøre fredagen inden næste sprint, så vi ikke står med denne opgave når vi møder op om mandagen og derfor kan komme i gang hurtigst muligt.

### 4.1.3: Daily SCRUM

***Ansvarlig: Alle***

Dertil vil vi også have dagligt SCRUM møde for at skabe et hurtigt overblik over, hvor vi står i processen og hvad vi skal arbejde på i løbet af den givne dag.

### 4.1.4: Sprint review:

***Ansvarlig: Alle***

I hvert sprint review vil vi tage et kig op hvor stort fremskridt vi har gjort med udviklingen af produktet. Fokus vil især lægges på i, hvilket omfang vores velocity ligger på og om vi følger den ideelle velocity vi fastsatte i starten af det tidligere sprint. Til dette vil vi anvende vores Kanban Board til at måle projektet og give os den info, der skal til for at tegne projektets kurs ind på et burndown chart

### 4.1.5: Sprint Retrospective:

***Ansvarlig: Alle***

Efter sprint review afholder vi sprint retrospective hvor vi finder forklaringen på afvigelser i velocity fra det forventede, ved at diskutere hvad der fungerer godt i vores arbejdsmetoder, og hvad der skal strammes op på.

### 4.1.6: User Stories

***Ansvarlig: Alle***

Vi vil i Sprint 0 opstille diverse user stories som vi kan arbejde ud fra vigtigheden via Business Value (forkortet BV), som er afsat af vores kunde, også kendt som vores Product Owner.

### 4.1.7: Kanban Board, ved brug af Trello

***Ansvarlig: Frederik***

Vi har valgt at organisere vores arbejdsopgaver ved brug af et Kanban board, frem for et SCRUMboard, da Kanban boards er mere gennemtestede og formaliserede end SCRUM boardet, der stadig har flere experimentelle træk.

### 4.1.8: Planning Poker

***Ansvarlig: Alle***

Når vi skal til at estimere vores User Stories i Sprint 0 vil vi gøre brug af metoden Planning Poker, da vi føler at denne fungerer bedst for os da alle får mulighed for at komme med deres mening og argumentere for deres valg af score points.

## 4.2: XP

### 4.2.1: Pair Programming

***Ansvarlig: Alle***

For at overkomme de mere komplekse dele af vores user stories, og for at undgå synkroniseringsfejl i vores kode, vil vi gøre brug af Pair Programming til at samarbejde om at få løst et problem. Disse tilfælde vil blive angivet i løbet af rapporten, når de finder sted.

Bruger i xp løbende lelr vil i benytte metoden i f.eks sprint 2. Måske i stedet skrive at i vil benytte Xp løbende i alle sprint når det findes naturligt.

### 4.2.2: Test Driven Development

***Ansvarlig: Alle***

Når vi skal i gang med at programmere i 1. Sprint vil vi afprøve Test Driven Development, da vi ikke har gjort os megen erfaring inden for denne arbejdsmetode og mener at det bedste tidspunkt at foretage et sådant arbejds eksperiment vil være i starten af programmeringen, hvor vi ikke har opbygget megen kode endnu.

### 4.2.3: Sustainable Pace

***Ansvarlig: Frederik***

Sustainable pace bruges i forløbet af projektet for at kunne holde den nødvendige mængde arbejde i projektet. Ved hjælp af sustainable pace kan vi sikre stabile sprint, hvor vi ikke springer noget arbejde over. Vi vælger at henvende til sustainable pace, om det lykkedes og resultaterne ved det i hvert enkelte retrospekt sprint.

Derudover vil vi også kort nævne at tirsdagene d. 27/11 og d. 4/12 har vi ikke skemalagte timer og derfor, på nuværende tidspunkt, vælger at holde fri, medmindre vi senere tænker at vi har brug for den ekstra tid.

Da vores projekt adskiller sig fra et virkeligt udviklingsprojekt ved at vi skal skrive en rapport og at det er denne, der er karaktergivende, har vi besluttet at afvige fra enkelte sustainable pace ideer, men på måder vi mener beholder kernen i metoden. Vi vil til kodning have faste arbejdstider som ikke må overskrides, for at simulere metodens krav om aldrig at overarbejde, men i stedet udelukkende bruge evt overarbejdstid på rapportskrivning.

Vores kodningstid er i starten sat til 4 timer om dagen, i modsætning til 6-8 man burde se i erhvervslivet af samme årsag. For os er det rapporten, og kun i mindre omfang vores produkt, der er udslagsgivende for dette projekts succes. For at vurdere om dette er nok arbejdstid til kodning, vil på hvert dagligt scrum møde spørge folk om de mener at vi skal øge kodningstiden. Ved projektets start vil vi kode fra 09:00 til 13:00, og skrive stikord til rapporten ud fra vores kode. Efter kl 13 vil vi udelukkende skrive på rapporten og derefter er der ingen faste tider for fyraften fra rapportskrivning.

### 4.2.4: Collective Code Ownership

***Ansvarlig: Alle***

Da dette er et fælles projekt, som skal gavne os alle, vil vi gå ud fra at ingen ejer specifikke dele af koden og alle har rettighed til at komme med ændringer og rettelser, selv hvis de ikke er den originale forfatter af koden.

## 4.3: UP

### 4.3.1: UML

***Ansvarlig: Frederik***

I dette projekt ønsker vi også at benytte UP’s UML-diagrammer til at visualisere vores design ideer og give os en arbejdstegning til første udkast når vi begynder at kode i 1. Sprint.

I skal måske allerede her have styr på, hvilke metoder I vil benytte I de enkelte sprint

Vi bryder med vilje med UML syntaksregler i første omgang da vores UML diagrammer i dette projekt ikke er ment som dokumentation, men som ide-værktøj til at skyde kodningen i gang med.

# 5: Sprint 0 / Inception Deck

## 5.1: Idégenerering

***Ansvarlig: Frederik + Anders***

Som allerede beskrevet i starten af rapporten blev vi præsenteret med en potentiel løsning af kundens problem i form af en robotarm. Vi har dog efterfølgende prøvet at komme frem til andre potentielle løsningsforslag for at se om en bedre løsning kunne forekomme.

1. Bevægelsessensor på hånden via gyroskop, som sender besked til sensoren om at lyset skal holdes tændt
2. Tryksensor i stolene, der ville virke som en override på sensoren ved at lyset altid holdes tændt hvis nogen sidder ned i rummet.
3. Varmesensor, igen på hånden eller i lommer, som kunne fornemme ændring af varme fra brugere frem for hvis den bare lå på bordet.

Efter en diskussion i gruppen kom vi frem til at gyrosensoren var den mest realistiske plan.

## 5.2: User Stories:

***Ansvarlig: Alle***

Vi har valgt user stories fordi det er en overskuelig og letforståelig måde at holde overblik over programmets funktionalitet på og hvem der skulle ville have brug for hvert enkelte funktion.

1. Som bevægelsessensor, vil jeg have en override switch, så jeg ikke slukker lyset når der er folk til stede. (bv 3) sp: 4
2. Som TS-applikation, vil jeg kunne sende data til væggens bevægelsessensor, så vi kan kommunikere. (bv 3) sp: 4
3. Som lærer, vil jeg kunne sende input til gyro-sensor ved at arbejde normalt på computeren(bv 5) Sp: 2
4. Som gyro-sensor vil jeg kunne videresende input til TypeScript Application. (bv 2) sp: 5
5. Som lærer vil jeg med TS-application kunne justere Bevægelse Sensorens refresh-rate (bv 4) sp: 7
6. Som lærer vil jeg kunne logge mig ind på TS-applicationen for at kunne begrænse, hvem der styrer lyset på kontoret. (bv 4) sp:3
7. Som admin/udvikler vil jeg kunne tilføje og fjerne brugere fra TS-applicationen (bv 5) (start her- Vibeke) sp: 3

7.1 Som admin vil jeg have en liste af brugere af denne service.(senere tilføjet: bv ?)

sp: 2

1. Som TS-application vil jeg kunne modtage tidspunkt fra 3rd part webservice (world clock API, kalender) (bv 1) sp:5
2. Som TS-application vil jeg have at de ændringer er gemt permanent selv når URL adressen ikke besøges (bv 2) sp: 5
3. Som Lærer vil jeg med TS-App kunne bestemme, hvornår bevægelsessensoren skal holde op med at modtage input fra TS-App’en (bv 5) (start her -Vibeke) sp:5
4. Som Lærer vil jeg kunne indstille Pi’ens sensitivitet i TS-App'en, så den registrerer mine bevægelser. (bv 5) (Start her - Vibeke) sp:5 umiddelbart ved I ikke endnu at det er disse jeg vil vælge så det skal nok angives I starten af sprint 1
5. Som REST Service med CORS vil jeg kun kunne modtage requests fra IP-adresser på Skolens adresse. (bv 1) sp: 5

Herfra kan vi se at vi har en score point sum på 55, som ville være ideelt at nå igennem inden projektets slutning. For at dette skulle lykkedes ville det være nødvendigt for os at opnå en velocity på 18-19 score points per sprint.

### 5.2.1: Acceptance Criteria

***Ansvarlig: Joachim + Anders***

1. Override
   1. (Success) Aktivering af override stopper sensor fra at slukke lyset.
   2. (Failure) override switchen virker ikke og lyset slukker.
2. Data til væg sensor
3. (success) Data bliver sent fra app’en, og registreres af væg sensor.
4. (Failure) Data sendes ikke fra app’en.
5. (Failure) Væg-sensor modtager ikke de sendte Data.
6. Input til Pi
   1. (Success) Gyroskop registrerer bevægelse.
7. Input fra raspberry pi til TS-app
8. (success) Input der sendes fra Raspberry pi, sendes til, og modtages af TS-applikationen
9. (Failure) Input modtages ikke af Raspberry pi.
10. (Failure) Input sendes ikke af raspberry pi.
11. (Failure) Inputtet modtages ikke af TS-applikationen
12. Sensor Refresh
    1. (Success) Tid til lys slukker er ændret efter nye indstillinger.
13. Login
    1. (Success) Bruger info står i hjørnet af hjemmeside.
    2. (Success) Knapper på hovedmenu kan tilgås.
    3. (Failure) Knapper kan tilgås uden login.
14. Tilføj/Fjern Bruger - Brugerliste
    1. (Success) Tilføjet bruger kan hentes via API.
    2. (Success) Længde af brugerliste er faldet med 1 efter at fjerne bruger.
    3. (Success) Brugerliste kan hentes via API.
    4. (Failure) Tilføjet bruger findes ikke i brugerliste.
15. Clock/Kalender API
    1. (Success) Ur kan ses på hjemmeside.
    2. (Success) Data kan ses på hjemmeside.
16. Permanent Indstilling
    1. (Success) Ændringer i indstillinger ses stadig efter hjemmeside er lukket ned.
17. Sluk Pi
    1. (Success) Pi deaktiveres ved tryk af knap på hjemmeside.
    2. (Failure) Pi er stadig aktiv efter tryk af knap på hjemmeside.
18. Pi Sensitivitet
19. (success) Ændring på sensitivity slideren i programmet, ændre hvor store eller små bevægelser pi’en registrere efter behov.
20. (Failure) ændring på slideren gør intet, og pi’en registrere uønskede bevægelser.
21. (Failure) Ændring på slideren gør intet og pi’en registrere ikke de ønskede bevægelser.
22. Rest modtager request fra skolens ip-adresser
23. (success) Rest-servicen modtager request fra skolen ip-adresser som ønsket.
24. (Failure) Rest-Service modtager ikke request sent med skolen ip-adresser
25. (Failure) Rest-Service modtager request fra andre ip-adresser, der ikke kommer fra skolen af.

## 5.3: Elevator Pitch

***Ansvarlig: Frederik***

Som et ekstra punkt er vi blevet bedt om at fremstille en Elevator Pitch, som et eksempel på, hvordan man ville kunne sælge ideen om vores aktuelle opgave, og denne ses nedenfor.

**Til:** Lærerne på EASJ Roskilde

**Som:** Skal kunne arbejde uforstyrret

**The**: LMS (Lights Management System)

**Er et/en**: Hjemmeside der gør brug af Raspberry Pi, egen og 3rd party World Clock API

**Som**: Gør det muligt for lærerne at indstille og afsende input til lys-sensoren (holde lyset tændt)

**I modsætning til**: At skulle afbryde arbejdet for at vinke til sensoren

**Vores produkt**: Er digitalt, automatisk og fleksibelt.

## 5.4: Product Box:

***Ansvarlig: Frederik + Anders***

På samme måde som vores Elevator Pitch blev vi også bedt om en Product Box som eksempel på en visuel og hurtig fortælling om, hvad vores endelig system ville involvere.

**LIGHT MANAGEMENT SYSTEM**

****

**Wave goodbye to obstinate sensors!**

**Automatic**

**Adjustable online**

**Secure**

**Jeg ved ikke om man kan sige at pæren er god måske sætte robotarmen ind her også**

## 5.5: Product Backlog: Det er ikke en rigtig produktacklog I har lavet ***Ansvarlig: Alle***

Nu, hvor vi har fået styr på vores User Stories og de aspekter involveret deri, kan vi sætte vores Product Backlog op for et let overblik over, hvad vi har at arbejde med og begynde at opstille en rækkefølge som vi vil arbejde ud fra i vores opkommende Sprint.

Her har vi fx. valgt at vi hurtigst muligt skal have sat en liste af brugere op, da denne er kritisk for at få løst User Story #7, som er blandt de højeste Business Values vi har at arbejde med. Samtidig har vi også rykket User Story #8 højere op end egentlig nødvendigt i en sand arbejdsprocess, da denne er vores involvering af 3rd Part API, som vi er krævet at skulle arbejde med i dette eksamensprojekt.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **ID** | **Item** | **Estimering** |
| 1 | 4 | Kommunikation | **5** |
| 1 | 7.1 | Liste af brugere | **2** |
| 1 | 7 | tilføj/fjern brugere | **3** |
| 1 | 11 | Pi sensitivitet | **5** |
| 1 | 3 | Pi registrerer input fra brugeren | **2** |
| 2 | 10 | On/Off switch til TS-app(til gyro-sensor) | **5** |
| 2 | 8 | World Clock API | **5** |
| 2 | 5 | Refresh Rate | **7** |
| 2 | 6 | Styre begrænsning | **3** |
| 3 | 1 | Override Switch | **4** |
| 3 | 2 | Data (TS-Sensor) | **4** |
| 3 | 9 | Permanent ændring | **5** |
| 3 | 12 | Rest/CORS get/put | **5** |

## 5.6: Visual WorkSpace

***Ansvarlig:***

• A story wall

• A release wall

• A velocity and burn-down graph

• An inception deck, if they have the room

## 5.7: Arkitektur overblik via domænemodel

***Ansvarlig: Alle***

For at få et hurtigt, umiddelbart overblik over arkitekturen af det arbejde vi skal foretage os i de næste par uger, har vi gjort brug af programmet Visio, hvori vi har opsat de elementer, der typisk bruges i opsætningen af en domænemodel fra UP.

Dette er dog ikke ment som en formelt korrekt domænemodel og er kun taget i brug som en visualisering af de mange elementer vi skal arbejde med, og hvad disse kan komme til at indeholde i fremtiden, både inden for den givne arbejdstid vi har at gøre godt med og i tilfælde af at vi havde mere tid til at tilføje ekstra funktionalitet. Bilag(Arkitektur domæne model)

## 5.8: Sprint retrospective:

***Ansvarlig: Alle***

I starten af november blev Joachim ramt af influenza, og Anders i midt på måneden. Dette gjorde i begyndelsen SCRUM som metode er svær at opretholde når gruppen er ramt af sygdom. Vores diagnoser gør nogle gange vores vilje til til kommunikation mangelfuld, hvilket gør det svært at opretholde de i vores gruppekontrakt specificerede retningslinjer.

Det lykkedes os at indhente det tabte arbejde og vi fik samlet op og re-grupperet tæt på slutningen af sprint 0.

Burndownchart 1 udkast

# 6: Sprint 1

## 6.1: Sprint backlog

***Ansvarlig: Alle igen her i er inde på noget af det rigtige men jeres backlog er ikke rigtig***

Blandt de vigtigste User Stories fra vores kunde vibeke står vi med at skulle begynde på tilføjelse og fjernelse af brugere, som vil gøre brug af systemet, samt at få vores gyro sensor til at registrere bevægelse fra brugerens hånd, og at man kan indstille sensitiviteten for denne sensor.

Vi står dog i en situation ,hvor vi ikke har et nuværende system at bygge op på, så vi bliver derfor nødt til først at sætte vores arkitektur op og skabe forbindelse mellem de individuelle elementer før at vi kan i sandhed komme i gang med vores arbejde. Dertil kræver vi også en database til opbevaring af listen af vores brugere, så disse kan gemmes i det at de bliver tilføjet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sprint** | **ID** | **Item** | **Estimering** | **Sprint** |
| 1 | 4 | Kommunikation | **5** | 1 |
| 1 | 7.1 | Liste af brugere | **2** | 1 |
| 1 | 7 | tilføj/fjern brugere | **3** | 1 |
| 1 | 11 | Pi sensitivitet | **5** | 1 |
| 1 | 3 | Pi registrerer input fra brugeren | **2** | 1 |

## 6.2: Daily Scrum

### 6.2.1: Nov. 26

***Ansvarlig: Anders***

Efter en langsom opstart pga. forsinkelser og fravær blev det besluttet at det umiddelbare startpunkt ville være at få start på vores arkitektur i form af at sætte en REST service op, så vi kunne gå i gang med at få funktionaliteten klar til tilføjelse og fjernelse af bruger.

Vi ville også gerne være kommet i gang med vores gyro sensor, men pga. Netproblemer var dette ikke muligt på nuværende tidspunkt.

### 5.2.2: Nov. 28

***Ansvarlig: Anders***

Eftersom at vi ikke kom i gang med UDP relateret arbejde var dette en af punkterne vi gerne ville tackle i løbet af dagen men pga usikkerhed om dele af funktionaliteten dertil, så har vi valgt at vente med dette til senere på dagen, hvor vores vejleder inden for teknik er til stede.

Dernæst skal vi også helt klart i gang med TypeScript/HTML delen, så vi kan fuldføre vores arbejde med tilføjelse og fjernelse af brugere via vores komplette REST API.

### 6.2.3: Nov. 29

***Ansvarlig: Anders***

Vores arbejde med brugere er overstået og mangler kun potentiel finpudsning i fremtiden.

Vi vil nu komme i gang med at arbejde med vores Raspberry Pi, men dette er dog ikke muligt fra vores standard arbejdslokale, da forbindelsen til det krævede netværk ikke er kraftig nok eller ikke kan tilgås overhoved.

Derfor vil vi til at starte med arbejde via en emulator hvor vi kan skrive nok kode til at få styr på vores registrering af bevægelses input og justering af indstillinger for sensitiviteten af input. Dog kan vi ikke arbejde med at sende beskeder fra vores Pi til vores UDPProxy, medmindre vi rykker op i klasselokalet.

### 6.2.4: Nov. 30

***Ansvarlig: Anders***

Inden at holdet dukkede op til dagens arbejde og SCRUM møde, blev der lavet test på vores forbindelse mellem Gyrosensoren og vores UDPProxy, da denne forbindelse ikke blev lavet i løbet af gårsdagen.

Dette viste sig så at være pga vi havde gjort brug af det forkerte netværk og blev derfor hurtigt bragt i orden. Derfor kunne vi nu vælge at gå i gang med kommunikationen mellem vores UDPProxy og REST, så vi kunne få forbindelse mellem alle elementerne af vores system.

Kan i på nogen måde dokumenterer hvad i har lavet ved hjælp af skærmdumps en prototype, noget kode ect.

## 6.3: Sprint Review

***Ansvarlig: Frederik + Anders***

Nu hvor vi er kommet til vejs ende for vores første sprint ser vores Kanban Board således ud.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Product Backlog** | **Sprint backlog** | **In progress** | **Testing** | **Done** |
| Story 10: On-off switch til TS-app |  | Story 4: kommunikation | Story 7: tilføj/fjern Brugere | Story 11: Pi Sensitivity |
| Story 8: world clock API |  | Story 6: Login |  | Story 3: Gyro sensor får input fra bruger |
| Story 5: refresh-rate |  | Story 9: Permanente ændringer |  |  |
| Story 1: Override Switch |  |  |  |  |
| Story 2: Data fra TS-APP til Vægs Bevægelsessensor |  |  |  |  |
| Story 12: CORS til REST |  |  |  |  |

Den største succeshistorie, på et rent fremskridts niveau, inden for dette sprint var vores arbejde med at få vores Gyrosensor til at registrere bevægelse, som er kommet fint på plads efter en opfriskning af kode (user story #3).

Derudover har vi fået fuldendt #11 til et funktionelt stadie, dog efter at vi har ændret i kravene til denne user story uden at have konsulteret vores kunde. Dette vil vi komme mere ind på i vores retrospective.

Derefter har vi dog diverse user stories, som stadig ikke er kommet fuldt på plads.

#7, som består af vores administration af brugere er komplet men vi har dog glemt at få lavet en unit test for at vise at der sker en ændring i længden af en brugerliste, når vi fjerner en bruger. Derfor står denne stadig under Testing men forventes at være simpel at løse senere.

Vores kommunikation i #4 har fået samtlige elementer sat op, så der kan kommunikeres den ene vej mellem alle delsystemer. Dog går denne kommunikation ikke hele vejen igennem men i stedet foregår halvvejs for to forskellige elementer: Bruger og Sensor. Da denne user story kræver at kommunikation foregår hele vejen fra vores Gyrosensor til vores HTML side, er denne stadig In Progress og er muligvis blevet fejlestimeret da denne user story har vist sig at være den mest besværlige for os i løbet af 1. Sprint.

Undervejs i vores arbejde indså vi at vi allerede har taget delvis hånd om #9, da vi har sat database tabeller op, hvori vores indstillinger kan gemmes. Derfor ser vi denne som værende In Progress, selvom denne ikke skulle være igangsat på nuværende tidspunkt.

Til sidst har vi også taget enkelte skridt ind i #6 ved at oprette en HTML side for login. Denne er dog stadig ikke udført betydeligt, men er stadig blevet sat igang og derfor In Progress.

Hvad betyder det at i ikke er blevet færdige for det næste sprint.

## 6.4: Sprint retrospective

***Ansvarlig: Frederik + Anders***

Men vigtigst af alt er et indblik i vores arbejdsprocess i løbet af vores første sprint, som kunne være gået betydeligt bedre.

### 6.4.1: Test Driven Development

***Ansvarlig: Frederik + Anders***

Tilbage i sprint 0 havde vi tænkt os at prøve at arbejde med test driven development i løbet af sprint 1, da vi tænkte at det bedste tidspunkt for dette ville være i starten, hvor vi ikke havde noget kode at arbejde med. Men pga sygefravær og tekniske problemer med at forbinde til vores Raspberry Pi, og derfor kunne arbejde på vores Gyrosensor, måtte vi revurdere vores prioritet af test driven development. Denne beslutning var også for at vi kunne gøre os nogen nævneværdige fremskridt i fremstillingen af vores software produkt, så test driven development blev ikke taget i brug på noget tidspunkt.

På nuværende tidspunkt ser vi ikke denne arbejdsmetode komme på banen igen, da det ideelle tidspunkt at gøre brug af den er passeret i vores øjne og vi foretrækker stærkt at kunne arbejde med test efter at vi har fået opsat kode på forhånd.

Hm hvad er det egentlig der kendetegnertestdriven development.

### 6.4.2: Pair Programming

***Ansvarlig: Frederik + Anders***

På den anden side fik vi dog taget Pair Programming i brug da vi begyndte at arbejde på systemer og kode, som vi ikke har megen erfaring med i forhold til hvad vi lærte i vores første studieår.

Disse tilfælde var arbejde med TypeScript og Python.

Vores første oplevelse med at tage Pair Programming i brug fremkom efter nogle timers arbejde om onsdagen på vores TypeScript applikation, som viste sig at være besværlig at arbejde med i forhold til C# kode, da vi var nødt til at arbejde på den samme fil, og versionsstyring blev et mindre mareridt, når alle tre i holdet prøvede at lave ændringer de samme steder.

Processen blev dog forbedret i løbet af dagen og vi kom ind i en mere optimal rytme, dog med enighed om at når vi arbejdede med denne type kodning, så burde vi holde os til at en enkelt person stod for at skrive mens resten hjalp til fra sidelinjen.

Dette blev så tilfældet om torsdagen, hvor vi endelig gik i gang med vores Gyrosensor, og nu holdt vi os til primært til en enkelt mand ved tasterne, som gjorde processen betydeligt lettere at håndtere.

### 6.4.3: Review og Retrospective

***Ansvarlig: Frederik + Anders***

Da vi endelig nåede til fredag og skulle i gang med at evaluere vores arbejde i løbet af sprint 1 kom vi dog frem til at vi havde en meget kaotisk struktur og at vi nu, efter en længere dag, havde rigtig mange punkter at skulle komme igennem og skrive om for at officielt kunne blive færdige med sprint 1.

I forhold til vores review og retrospective, som begge var planlagt kun at bruge tid på i slutningen om fredagen, så kom vi frem til at vi burde prøve at sprede disse diskussioner ud over ugens strabadser. Dette tænker vi at gøre ved at diskutere individuelle user stories, når vi mener at disse er i deres endelige stadier. På denne måde bliver arbejdet bedre fordelt i løbet af ugen, samt at vi både får snakket om arbejdet, når det er mest friskt i hukommelse, og at vi har mere rapportskrivning på dagen udover fredag, hvor vi førhen kun havde vores Daily SCRUM.

### 6.4.4: Daily SCRUM

***Ansvarlig: Frederik + Anders***

Angående Daily SCRUM så har disse været et betydeligt element i vores kaotiske struktur, som ikke kan ses som en umiddelbar struktur overhoved da vores mangel på erfaring gjorde at disse møder primært blev til en enkelt person, som kom med de fleste ideer på varierende tidspunkter om morgenen, og pga dårlige perioder for andre medlemmer af holdet var vi aldrig fuldtallige til SCRUM møder.

Derfor har vi nu, efter en opfølgning på den typiske procedure at Daily SCRUM, besluttet af stramme op, så vi kan prøve at gøre korrekt brug af dette værktøj. Dette har vi tænkt os at gøre ved først og fremmest at fastsætte tidspunktet på vores møder til kl. 10:00. Herfra vil vi enkeltvis, i en fastsat rækkefølge, nævne:

* Hvad vi sidst har arbejdet på.
* Hvad vi vil gå i gang med på den givende dag.
* Hvad kan forekomme som problematisk i løbet af dagens arbejde.

Dette skal ikke tage længere end 10 minutter og med vores gruppe størrelse vil det give 3 minutter hver, hvor hvert emne får et enkelt minut.

### 6.4.5: Kanban Board

***Ansvarlig: Frederik + Anders***

Som tidligere nævnt, så valgte vi i 1. Sprint at arbejde på review om fredagen. Dette betød også at vi blev disorganiseret i forhold til vores Kanban Board, som blev opdateret løbende men ikke under de korrekte omstændigheder.

I den kaotiske process glemte vi at tjekke vores acceptance criteria inden at vi rykkede rundt på vores user stories. Dette gjorde at da vi kom til fredagen, fandt vi frem til at en del som vi mente var færdige, ikke var så langt pga anden formulering da vi satte vores user stories op i forhold til den fremgangsmåde vi senere kom frem til da vi begyndte at kode.

Derfor har vi gået imod den akkurate metode ved at fx sige at vores user story #11 er Done ud fra det resultat vi har opnået, men dette ville ikke være acceptabelt ellers når man arbejder med SCRUM. Den korrekte fremgangsmåde ville have været at vi diskuterede den givne ændring med kunden og argumenterede for, hvorfor dette ville være en bedre løsning på nuværende tidspunkt. Med denne erfaring gjort vil vi samt også stræbe på at være bedre til at opholde vores user stories som de er ud over #11 og i tilfælde af ændringer, tage dem op med kunden.

Med disse erfaringer blev vi derfor enige om at inden man rykker en user story, skal man huske at tjekke deres acceptance criteria og teste derudfra. Hvis alle disse er på plads kan man derefter flytte den givne user story til Done efter fælles vurdering, og når en user story er nået til Done skal der skrives i vores Sprint Review, så dette bliver gjort i forhold til vores nye fremgangsmåde per 2. Sprint.

### 6.4.6: Sustainable Pace

***Ansvarlig: Frederik + Anders***

For at vi bedre kan gøre brug af Sustainable Pace, så i 1. Sprint ikke udmundede sig til meget, bl.a fordi vi ikke havde meget at skrive om i løbet af ugen, pga vores ide om at review og retrospective skulle holdes til fredagen, har vi besluttet at fastlægge rapportskrivning til kl. 13 og derefter.

På denne måde kan vi garantere os at der altid bliver skrevet på rapporten og at der er tid sat af til gennemløbende review, når dette er tilfældet. Samtidig, i tilfælde af at der ikke er noget nyt at tilføje til rapporten, vil denne tid blive brugt på gennemlæsning og omskrivning, således at vi alle holder os opdateret på rapportens stadie.

Denne tid med fokus på rapportskrivning vil altid være på en lektions varighed af 45 minutter og dette, samt tidsplanen på Daily SCRUM og opdatering af Kanban Board, vil blive bedre opholdt af vores SCRUM Master, så vi kan få en betydelig bedre oplevelse af SCRUM i vores 2. Sprint.

Jeg vil egentlig også gerne have et burn down chart ind I rapporten.

# 7: Sprint 2

## 7.1: Kanban Board

***Ansvarlig: Frederik***

Ved starten af 2. Sprint er status for vores Kanban Board således:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Product Backlog** | **Sprint backlog** | **In progress** | **Testing** | **Done** |
| Story 1: Override Switch | Story 10: On-off switch til TS-app | Story 4: kommunikation | Story 7: tilføj/fjern Brugere | Story 11: Pi Sensitivity |
| Story 2: Data fra TS-APP til Vægs Bevægelsessensor | Story 8: world clock API | Story 6: Login |  | Story 3: Gyro sensor får input fra bruger |
| Story 12: CORS til REST | Story 5: refresh-rate | Story 9: Permanente ændringer |  |  |

Vi var på den ene side bagud med sprint 1’s stories, grundet komplikationerne nævnt i slutten af sprint 1, men samtidig var vi allerede gået i gang med Login siden til sprint 2.

## 7.2: Daily Scrum

### 7.2.1: Dec. 3

***Ansvarlig: Anders***

Desværre starter vores nye struktur af Daily SCRUM dårligt ud, da vi denne dag kun er en enkelt til stede mens resten af holdet er utilpas.

Derfor kan vi kun dele at Anders har arbejdet på rapport, som ikke blev færdiggjort for 1. Sprint og vil fortsætte med dette, så gruppen kan komme godt i gang med 2. Sprint, når folk er tilbage på benene. Det eneste umiddelbare problem med dette er enkelte dele af rapporten, der kræver nogle spørgsmål til en vejleder for at vi kan fastlægge hvor et par af vores user stories ligger i forhold til vores arbejdsprocess.

### 7.2.2: Dec. 5

***Ansvarlig: Frederik + Anders***

Vi har nu haft vores første, strukturerede Daily SCRUM og er kommet frem til følgende for hver enkelt medlem af gruppen:

* Frederik har sidste fredag arbejdet med Python og sikret at denne kode blev gemt andetsteds for garantiens skyld, men derudover ikke gjort andet pga flytning af lejlighed. Vil i dag kigge rapporten igennem og derefter assistere Joachim med TypeScript via Pair Programming og forudser ingen umiddelbare problemer udover dem stillet af Joachim selv.
* Anders har i mandags skrevet på rapport og i dag haft kommunikation med kunden, hvoraf mere af rapporten er blevet skrevet på. Vil arbejde på en unit test af vores user story #7 og rydde op i dele af koden fra 1. Sprint.
* Joachim har i fredags arbejdet på konstruktion af en login side i vores HTML og vil fortsætte med dette i dag. Dog forudses nogle problemer med hvordan værdier bliver henter fra vores database via TypeScript, som Frederik vil prøve at hjælpe med.

### 7.2.3: Dec. 6

***Ansvarlig:***

### 7.2.4: Dec. 7

***Ansvarlig:***

## 7.3: Sprint Review

***Ansvarlig:***

## 7.4: Sprint Retrospective

***Ansvarlig:***

# 8: Sprint 3

## 8.1: Sprint backlog

***Ansvarlig:***

## 8.2: Daily Scrum

### 8.2.1: Dec. 10

***Ansvarlig:***

### 8.2.2: Dec. 11

***Ansvarlig:***

### 8.2.3: Dec. 12

***Ansvarlig:***

### 8.2.4: Dec 13

***Ansvarlig:***

### 8.2.5: Dec 14

***Ansvarlig:***

## 8.3: Sprint Review

***Ansvarlig:***

## 8.4: Sprint Retrospective

***Ansvarlig:***

# 9: Konklusion

***Ansvarlig:***