KINO Management System

Casper Frost Andersen  
Jonas Olsen  
Patrick Sirich  
Matthias Skou  
Michael Jensen  
Benazeer Turkmani  
Christian Strunge  
Emil Fenger

Indhold

[Introduktion 2](#_Toc528070599)

[Etablering af projektgruppe 3](#_Toc528070600)

[Kontrakt & Retningslinjer 3](#_Toc528070601)

[Opsætning af projekt 3](#_Toc528070602)

[Kommunikationsplatform 3](#_Toc528070603)

[SCRUM-board 3](#_Toc528070604)

[GitHub repository 3](#_Toc528070605)

[Tools & Frameworks 4](#_Toc528070606)

[Rollefordeling uge 1: 4](#_Toc528070607)

[Rollefordeling uge 2: 4](#_Toc528070608)

[Rollefordeling uge 3: 4](#_Toc528070609)

[iteration 1 5](#_Toc528070610)

[Tasks 5](#_Toc528070611)

[Planlagt arbejde i første iteration 6](#_Toc528070612)

[Færdiggjort arbejde i første iteration 6](#_Toc528070613)

[Burndown chart 6](#_Toc528070614)

[Logbog for iteration 1 7](#_Toc528070615)

[Tirsdag 5-10-2018 7](#_Toc528070616)

[Onsdag 10-10-2018 7](#_Toc528070617)

[fredag 12-10-2018 8](#_Toc528070618)

[iteration 2 8](#_Toc528070619)

[Planlagt arbejde for anden iteration 9](#_Toc528070620)

[Færdiggjort arbejde i anden iteration 10](#_Toc528070621)

[Burndown chart 10](#_Toc528070622)

[Logbog for iteration 2 10](#_Toc528070623)

[Iteration 3 12](#_Toc528070624)

[Tasks 12](#_Toc528070625)

[Planlagt arbejde i trejde iteration 12](#_Toc528070626)

[Færdiggjort arbejde efter tredje iteration 12](#_Toc528070627)

[Burndown chart 13](#_Toc528070628)

# Introduktion

Ejeren af KinoXP har henvendt sig til vores virksomhed, med henblik på at få udviklet et nyt IT-system som kan håndtere reservationer og salg af billetter, til de forskellige forestillinger der bliver vist i biografen. KinoXP er en lille biograf med to sale, som pt. håndterer alle reservationer og salg manuelt, og ønsker nu at digitalisere sine arbejdsmetoder.  
Systemet skal udelukkende betjenes af medarbejderne i biografen, som håndteringsværktøj til reservationer og salg af billetter og slik/sodavand, eftersom reservation og salg foregår over telefon.

Vi vil under dette forløb anvende eXtreme programming (herefter XP) som udviklingsmodel, og følge de dertil indrettede retningslinjer for udviklingsforløbet. XP er en agil og test-driven udviklingsmodel som fokuserer på leveringsdygtighed. Under forløbet vil vi have tæt kontakt til kunden for at sikre, at produktet lever op til dennes krav.

Denne rapport har til formål at dokumentere forløbet for vores udviklingsproces, vores arbejdsmetoder, udviklingsværktøjer, udfordringer, iterationer og relevante kodeudsnit og modeller/diagrammer.

# Etablering af projektgruppe

Vi er en gruppe bestående af 8 studerende. Under dette projekt vil vi løbende indgå i mindre grupper ifm. pair-programming. Derudover vil der være en projektmanager som sørger for, at udviklingen forløber som det skal.

Hertil har vi lavet en kontrakt med retningslinjer, som det forventes at gruppen følger:

## Kontrakt & Retningslinjer

1. Projektgruppen har obligatorisk fremmøde i tidsrummet 8:30 - 11:45.
2. Ved sygdom meldes det i Slack-gruppen så hurtigt som muligt - sygemelding efter 8:30 godtages ikke. Ved sygemelding vil de resterende fra gruppen i aftale på dagen, hvordan det videre forløb skal foregå.
3. Der arbejdes i de grupper som er blevet fastsat i starten af ugen. Hvis der løbende skal foretages ændringer, skal projektmanageren samt alle implicerede parter godkende disse.
4. Pauser aftales i fællesskab, evt. i de par der sidder sammen. Fast pause mellem 10:00 - 10:15.
5. Stand-up meeting bliver som udgangspunkt afholdt 8:35 - 8:50 senest.
6. Hvis der er behov for det, vil der blive afholdt et debriefing møde i slutningen af dagen.

## Opsætning af projekt

Inden vi påbegynder projektet har vi nogle generelle opsætninger som skal forudsætte vores arbejdsgang, som bliver redegjort herunder:

### Kommunikationsplatform

Vi vil anvende Slack som primær kommunikationsplatform. Alle sygemeldinger eller forsinkelser vil blive angivet her, samt links til arbejdsværktøjer såsom GitHub, Google Docs ect.

### SCRUM-board

Vi vil benytte os af et SCRUM-board som værktøj under udviklingen. Dette vil bestå af en Product Backlog, en Sprint Backlog, samt tre kolonner der angiver hvilke tasks der er under udvikling, hvilke der er i test-fasen, og hvilke der er færdigudviklede.

### GitHub repository

Vi benytter os af GitHub som synkroniseringsværktøj. Link hertil:  
<https://github.com/michjens/Kino-XP>

### Tools & Frameworks

Herunder en liste af de værktøjer vi vil benytte os af til projektet:

* IntelliJ version 2018.2.4
* Spring
* JS
* HTML
* CSS
* MySQL (Workbench, version 6.3)
* Visual Paradigm version 15.1

### Rollefordeling uge 1:

I den første uge vil Casper tage rollen som projektmanager. Benazeer vil tage notater under kundemøder.

Herefter har vi aftalt følgende pair-programming grupper:

* Patrick & Matthias
* Christian & Michael
* Emil & Benazeer
* Casper & Jonas

### Rollefordeling uge 2:

I den anden uge vil Patrick tage rollen som projektmanager.

Herefter har vi aftalt følgende pair-programming grupper:

* Patrick & Christian
* Casper & Jonas
* Emil & Michael
* Benazeer & Matthias

### Rollefordeling uge 3:

I den tredje uge vil Benazeer tage rollen som projektmanager.

Herefter har vi aftalt følgende pair-programming grupper:

* Patrick & Michael
* Jonas & Christian
* Emil & Benazeer
* Matthias & Casper

# iteration 1

Herunder de user storys vi vil udvikle ud fra pr. 09-10-2018 (i prioriteret rækkefølge):

Som bruger skal jeg kunne logge ind: 6 timer

Som bruger skal jeg kunne se en oversigt over alle film: 5 timer  
Som bruger skal jeg kunne oprette og redigere en film: 5 timer  
Som bruger skal jeg kunne oprette en booking: 18 timer

Som administrator skal jeg kunne se en oversigt over salg af billetter for hver film: 3 timer

## Tasks

Ud fra svarene på disse har vi udviklet disse user storys i prioriteret rækkefølge, med estimerede tasks:

1. Som bruger skal jeg kunne logge ind: 6 timer
   1. Lav bruger klasse (1 time)
   2. Lav bruger tabel i database (1 time)
   3. Fornavn og efternavn skal tilknyttes til hvert login (2 timer)
   4. Ansatte skal kunne logge ind med e-mail og kodeord (1 time)
   5. Bruger skal gemmes i Session (1 time)
2. Som bruger skal jeg kunne se en oversigt over alle film: 5 timer
   1. Opret forside. Tilføj logo og navn på biografen samt navigation til undersider (1 time)
   2. Implementer en oversigt over alle film. Film skal vise:
      1. Navn
      2. Vigtigste skuespillere
      3. Aldersgrænse
      4. Plakat
      5. Kategori
3. Som bruger skal jeg kunne oprette og redigere en film: 5 timer
   1. Lav film klasse og tilhørende HTML-side (1 time)
   2. Lav film tabel i database og ER-diagram (1 time)
   3. Opret fast liste med kategorier i en drop-down menu (1 time)
   4. Tilføj funktion så kun administrator kan ændre i prisen på film (1 time)
   5. Tilføj gem/godkend/annuller funktion (1 time)
4. Som bruger skal jeg kunne oprette en booking: 18 timer
   1. Lav booking klasse (1 time)
   2. Lav booking tabel i database (1 time)
   3. Tilføj checkboks til rabat på slik/sodavand (2 timer)
   4. Tilføj checkboks til pensionistrabat (1 time)
   5. Tilføj grafisk oversigt over sæderne i biografsalen med dertilhørende grafisk interaktion, så man kan vælge x antal sæder (8 timer)
   6. Tilføj status til booking af sæder: ledig (grøn) reserveret (gul) solgt (rød) (2 timer)
   7. Tilknyt en medarbejder til booking (1 time)
   8. Tilføj gem/godkend/annuller funktion (2 timer)
   9. Tilføj slet-funktion (1 time)
5. Som administrator skal jeg kunne se en oversigt over salg af billetter for hver film: 3 timer
   1. Lav separat HTML side for oversigt over billetsalg. (1 time)
   2. I billetsalg-oversigt skal de individuelle bookinger kunne vises(kundeinfo osv.) (2 timer)

### Planlagt arbejde i første iteration

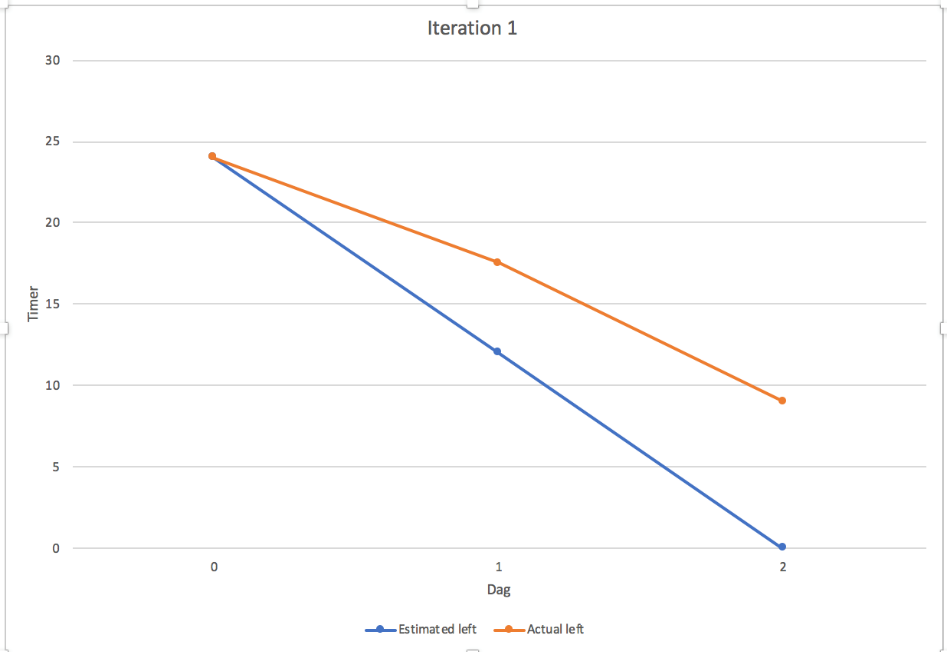
I første iteration planlagde vi i fællesskab at udvikle de 3 højest prioriterede user stories.

Gruppen besluttede at gøre brug af planning poker for at estimere tid pr task. Alle gruppemedlemmer sad i en cirkel og i samarbejde med hinanden estimerede vi vores task. Vi blev enig om tiden ved at smide det antal timer vi mente hver task ville tage, hvis gruppemedlemmer var uenig skulle man argumentere for sit valg og efterfølgende diskuterede gruppen hvorvidt man skulle reestimere eller holde sig til det man kom frem til første gang. Vi gik ud fra at vores estimater var realistiske og håndgribelige i forhold til den arbejdsmængde vi havde til rådighed i hver iteration.

### Færdiggjort arbejde i første iteration

Efter 1. iteration blev afsluttet og vi havde færdigudviklet de 2 højest prioriterede user stories, opdagede gruppen at vi havde fejlestimeret flere task og valgte at holde endnu et møde for at diskutere hvorvidt vores arbejde kunne gennemføres eller om vi skulle reestimere de resterende tasks, så projektet kunne komme på rette spor igen. Vores møde resulterede i at vi både havde fejlestimeret flere task og fejlestimeret den tid gruppen havde til rådighed til hver task. Gruppen blev enig om at det nye mål skulle være at bliver bedre til estimering i 2 iteration, for at sikre at projektet bliver gennemført.

### Burndown chart



### Logbog for iteration 1

Dette afsnit har til formål at dokumentere vores arbejde, tanker og fremgangsmetoder. Vi vil løbende føre individuelle logs over de grupper vi arbejder i, f.eks. i pair-programming, prototyping ect.

### Tirsdag 5-10-2018

Første dag fik vi inddelt roller, opsat vores arbejdsmiljø inkl. Git, slack og Docs. Derefter gennemgik vi opgavebeskrivelsen og delte os op i vores pair programming grupper, og fik udfærdiget nogle spørgsmål til kunden. Spørgsmålene satte vi op mod hinanden, og kom frem til nogle endelige spørgsmål:

1. Hvad er pris-grupperne for billetter? Er der kategorier for børn og senior?
2. Skal film kun have én kategori eller flere? Skal de opdeles i voksen og børnefilm? Måske i aldersgrænser?
3. Hvad er de forskellige aldersgrænser? A / 7 / 11 / 15? Skal systemet afvise i tilfælde af at aldersgrænse ikke er overholdt? Eller er det den enkelte medarbejder der vurderer? Senior / børnebilletter?
4. Er der nogle specielle sæder? Hvis ja, er der tillæg til disse?
5. Hvordan skal en kunde registreres? Telefonnummer? Email? Navn? Hvilke informationer skal en booking indeholde? Tjek af til reklame.
6. Skal film oversigten vises som en film med flere datoer eller en kalender med flere film per dag?
7. Hvordan skal ekstra visninger vises? Skal ekstra visninger promoveres specielt? Hvad er seneste frist? Hvad dækker et ekstra-show over? Er ekstra-shows på bekostning af andre shows?
8. Hvordan måler vi et dårligt salg? Sker fjernelser af film automatisk? Skal der være statistik over billetsalg mm.?
9. Skal salgsappen af billetter og silk/sodavand være sammen eller adskilt?  Evt. med kapagner (20% rabat på en stor menu ved bestilling over telefonen)
10. Er de eneste med adgang de to ansatte der står for salg af billetter? Skal de ansatte have specifikke funktioner?
11. Hvem står for oprettelse af nye film?

Vi har lavet user storys og tasks ud fra disse, som fremgår af User storys uge 1 afsnittet

### Onsdag 10-10-2018

#### Individuelle logbøger:

##### Michael & Strunge:

Vi har fået lavet en klasse der indeholder informationer om en bruger, vi besluttede ikke at lave sub-classes, da “admin” og “ansat” ikke har forskellige fields og lign. Dette forløb uden problemer.

Vi har lavet en funktion hvori man kan differentiere admin fra ansat, den er i øjeblikket i “test”, her snakkede vi om hvordan og hvorledes en metode skulle kunne sørge for dette. mindre bump på vejen, men vi kom i mål (Til test).

Vi har lavet en funktion der kan slette film fra databasen ud fra deres ID.

##### Benazeer & Emil:

Vi har lavet en film klasse, samt en forside med navn og logo i html og css og en oversigt over film i html/css (ingen kode). Udover det har vi lavet fundamentet til navigationen på forsiden.

##### Jonas og Matthias

Jonas har lavet login side og fået det til at fungere med database. Matthias hjalp med database-delen. Til sidst blev der tilføjet et meget svært og krævende Session funktion, så brugere kan gemmes deri.

##### Patrick og Matthias

Vi har lavet et meget flot logo til KinoXP. Vi har oprettet et ER-diagram, lavet tre følgende tabeller: Film, Visninger og Bruger. Disse tabeller har vi oprettet i en database, hvor idFilm er en foreign key i Visninger. Patrick gik kl. 10 fordi han skulle til fysioterapeut. Vi havde fejl-estimeret hvor længe det ville tage, at oprette disse i databasen.

### fredag 12-10-2018

Michael:

MIA - syg

##### Matthias og Strunge

Lavede login css, tilføjede baggrund, logo mm. Derefter lavede vi forside css. Vi ville gerne have et flot baggrundsbillede som går igen på hele hjemmesiden. Det har vi fået tilføjet og kunden var meget tilfreds med designet på siden.

##### Benazeer og Emil

Lavede oversigt over film, samt implementeret databasen.

##### Patrick

Satte server op med brugere og tilladelser og fiksede de problemer der var med databasen.

##### Frost og Jonas

Vi har opdateret rapporten og arbejdet på burndown chart. Jonas (med Casper som partner) har lavet CSS på forsiden, og sørget blandt andet for at film står på række. Vi lavede også opacity på baggrundsbilledet efter login.

# iteration 2

2. iteration start fredag 12-10-18 kl. 11

2. iteration slut fredag 19-10-18 kl. 12

2 iteration startede med et kundemøde, hvor gruppen fik yderligere ønsker og funktioner, kunden godt kunne tænke sig. Kunden tilføjede ikke flere User stories efter mødet.

I følgende afsnit vil alle ændringer til systemet være dokumenteret.

1. Som bruger skal jeg kunne se en oversigt over alle film: 5 timer
   1. Header/menubar skal følge med. (1 time)
   2. Finpuds css på forside-film(tekst) og centrer plakat (1 time)
   3. Manuel funktion der fjerner film fra oversigten og databasen(ikke kun knap) (2 timer)

Efter første iteration kunne vi se at tasksne i user story 3, 4 og 5 var fejlestimeret og vi valgte at lægge nogle tasks sammen og reestimere dem alle.

1. Som bruger skal jeg kunne oprette og redigere en film: 4 timer
2. Lav Film klasse: navn, dato, tid, pris, sal, aldersgrænse, vigtigste skuespillere, kategori. (1 time)
3. Lav Film tabel i database og ER-diagram. (1 time)
4. Lav en Opret Film funktion

Opret en fast liste med kategorier i en drop-down menu. (1 time)

1. Tilføj en redigeringsfunktion.

Tilføj funktion så kun administrator kan ændre i prisen på film.

Tilføj en gem/godkend og annuller funktion. (1 time)

1. Som bruger skal jeg kunne oprette en booking: 24 timer
2. Lav booking oversigt (rediger/slet) (1 time)
3. Opret Booking klasse med følgende informationer: fornavn, efternavn, telefonnr og email på kunden. Email kan være tom. Tilvalg af reklamer via. e-mail.

Tilknyt en medarbejder til en booking.

Tilføj annuller-knap.

Tilføj gem/godkend funktion.

Tilføj en funktion der sletter en booking.(4 timer)

1. Lav Booking tabel i database og ER-diagram.(1 time)
2. Tilføj checkboks til rabat-felt til indtastning af på slik, popcorn og sodavand.

Tilføj check-felt til pensionist-rabat på 10 % (2 timer)

1. Tilføj grafisk interaktion med sæderne, så man kan vælge x antal sæder af gangen. (8 timer)
2. Tilføj grafisk oversigt over sæderne i biografsalen. (4 timer)
3. Tilføj følgende status til booking af sæder: ledig, reserveret, betalt og markerede. (4 timer)
4. Som administrator skal jeg kunne se en oversigt over salg af billetter for hver film: 9 timer
5. Lav separat HTML side for oversigt over billetsalg. (1 time)
6. Vis dagligt billetsalg over alle aktive film. (1 time)
7. Søg i billetsalg-oversigten efter film-navn(sortering) (2 timer)
8. I billetsalg-oversigt skal der være en drop-down med datoer hvor den enkelte film er blevet vist/bliver vist. (2 timer)
9. I billetsalg-oversigt skal de individuelle bookinger kunne vises(kundeinfo osv.) (2 timer)
10. Udregn billetsalg / pris og vis total for disse. (1 time)

### Planlagt arbejde for anden iteration

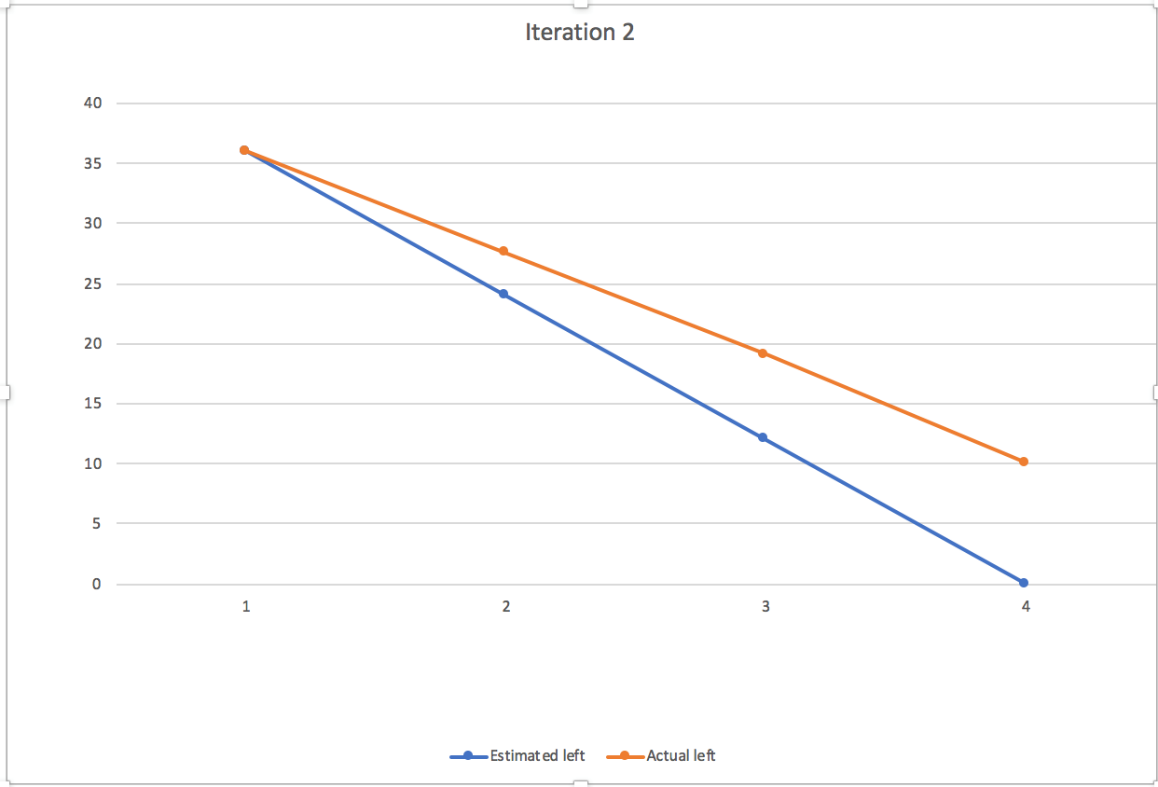
I anden iteration planlagde gruppen at udvikle yderligere 2 user stories. Disse 2 user stories omhandler alle funktioner i bookings og film, hvor man både skal kunne oprette, slette, rediger og vise en oversigt over dem.

Endnu engang lavede gruppen planningpoker for at estimere tiden for hver task og denne gang estimerede gruppen hver task mere realistisk og tog forhold til den mængde arbejdstid vi havde som gruppe.

### Færdiggjort arbejde i anden iteration

Efter anden iteration havde vi lavet begge de user stories vi havde tænkt os at lave. Tirsdag d. 16 blev vi tilbage og arbejdede efter 12, for at få lavet alt det vi skulle nå i denne iteration. Dette extraarbejde gjorde at vi kunne blive færdige med alt det arbejde vi havde planlagt fredagen efter. Dette var en mere successfuld iteration med nogle mere præcise estimater, dog havde vi ikke regnet den extra tid vi brugte tirsdag i vores estimat så vi endte med at bruge mere tid end vi havde planlagt.

### Burndown chart



### Logbog for iteration 2

Tirsdag d. 16-10-2018

##### Patrick & Christian

Vi skulle lave en dropdown menu/funktion til kategorier, dette fik vi overkompliceret lidt, og brugte derfor mere tid end vi først antog. Vi endte med at hardcode menuen i HTML, istedet for at hente kategorierne fra databasen. Efterfølgende har vi påbegyndt en funktion på visninger, hvori der skal kunne oprettes x antal visninger knyttet til en bestemt film, dette fortsætter vi med næste gang, da vi ikke er blevet færdige. Vi stødte ind i et problem mht. datetime i HTML.

##### Frost og Jonas:

Vi skulle lave en grafisk oversigt over to sale, med interaktive og nummererede sæder, hvor man ved at trykke på dem kunne vælge et eller flere sæder, som senere skal kunne reserveres eller sælges.

Vi besluttede at genere sæderne dynamisk vha. nested for-loops for hver sal. Dette gjorde det også meget lettere at give individuelle sæder række og nummer.

Vi benyttede os af et layout inspireret af denne side:

<https://codepen.io/rastogi_saurav/pen/qqROdw>

Derudover har vi tilføjet en skærm øverst på siden, med gulligt lys, og en ind/udgang i bunden, med rødt lys.

##### Matthias og Benazeer:

Vi har oprettet en booking klasse med følgende fields: kundeFornavn, efternavn, telefon, email og tilvalg af reklamer. Man skulle kunne tilknytte medarbejder til denne booking og vi har tilknyttet vedkommende medarbejder der er i den session.

Opret booking er fuld funktionel og smider bookingen over i databasen uden problemer dog mangler vi annullere funktion og redigering.

##### Michael og Emil

Vi lavede en fuldt funktionel opret film funktion samt rediger film funktion. I vores rediger funktion havde vi problemer med at opdatere/overskrive film. Det viste sig at være pga. filmID aldrig blev overført til post’en og derfor ikke kunne findes i databasen. Dette blev løst og virker nu uden problemer.

Fredag d. 19-10-2018 før kl 12

##### Michael og Emil:

Vi har lavet css på opret film, og rediger film funktionerne, samt lavet en separat fil til header’en og lagt denne ind på flere af vores sider i programmet.

##### Matthias og Benazeer:

Vi har lavet redigering og slet funktion på booking. Vi har lavet css til begge html sider og lagt header ind.

##### Frost og Jonas:

Vi researchede hvordan man binder spring og javascript sammen, med henblik på at lave status på sæder.

Casper finpudsede lidt CSS på forsiden

Jonas hjalp Christian og Patrick med visninger, nærmere specifikt dato/tid.

##### Patrick & Christian:

Vi har færdiggjort opret visning funktion, med mindre problemer undervejs, men ikke noget der ikke kunne løses hurtigt.

# Iteration 3

Iteration start fredag 19-10-18 kl 12

Iteration slut tirsdag 23-10-18 kl. 11.15

I tredje iteration fik vi nye user stories, men ingen ændringer til de user stories vi allerede havde. Disse fik også en ny prioritet i forhold til hvad der manglede i prototypen.

De nye user stories i prioriteret rækkefølge, samt de manglende user stories fra forige iterationer:

1. Som bruger skal jeg kunne sælge slik og sodavand over app’en. 7 timer
2. Som bruger skal jeg kunne se en oversigt over vagtplanen. 1 time
3. Som administrator skal jeg kunne se en oversigt over salg af billetter for hver film: 9 timer

## Tasks

Alle nye task tilføjet i tredje iteration.

1. Som bruger skal jeg kunne sælge slik og sodavand over app’en. 7 timer
2. Lav en Slik tabel og klasse, samt tabel til at håndtere silk salg (1 time)
3. Lav en opret/redigér slik funktion (2 timer)
4. Lav salg af slik med dropdown og antal.

Dropdown starter på en og oprettes derefter dynamisk (4 timer)

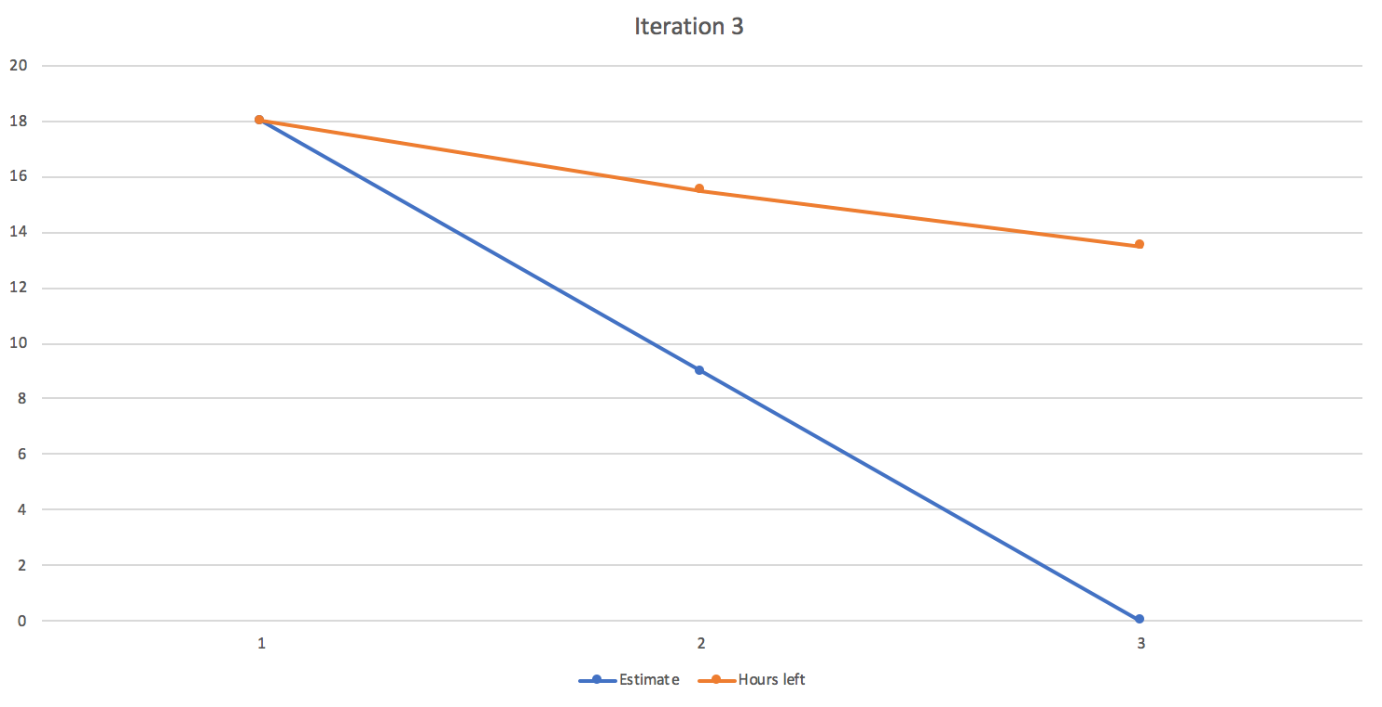
1. Som bruger skal jeg kunne se en oversigt over vagtplanen. 1 time
2. Lav en tabel i HTML der viser en vagtplan.

### Planlagt arbejde i trejde iteration

Efter anden iteration havde vi et møde med kunden der sagde at vi skulle putte nye funktioner ind i programmet. Disse nye funktioner; vagtplan og sliksalg, var prioriteret højere end de manglende funktioner i programmet, så planlægningen til trejde iteration, gik på at estimere de nye user stories da det var disse vi skulle lave i trejde iteration.

### Færdiggjort arbejde efter tredje iteration

### Burndown chart



##### Fredag 19-10-18 efter kl. 12

##### Matthias, benazeer, jonas, frost & christian:

Vi lavede en vagtplan i html. Vi lavede en tabel og hardcodede tider og navne ind. (vi havde ikke tid til at lave funktion der henter fra databasen).

Vi lavede en Slik tabel og klasse, samt tabel til at håndtere slik salg samt funktionalitet til både at opret/redigér slik funktion

##### Tirsdag 23-10-18

##### Emil & benazeer:

Arbejdet på rapport

##### Patrick & michael:

Vi har lavet et klassediagram og ER-diagram. Patrick har forsøgt sig på at lave en funktion som viser den seneste visning, men måtte give op grundet manglende tid.

##### Jonas og christian:

Startede på at lave salg af slik, nåede at lave en enkel dropdown for en slik-type samt udregning ud fra antal. Dynamisk oprettede dropdowns mangler.

##### Matthias & casper:

MIA