

KØBENHAVNS UNIVERSITET

PROJEKTKURSUS SYSTEMUDVIKLING 2014

Open Source Bannerreklamesystem
EasyAd

Projektgruppe:

Signar Nielsen
060585

Instruktor:

Aske Mottelson Clausen

Aslak Niclasen
060693

Amanda Bergqvist
171090

24. april 2014

Indhold

1	Indledning	2
2	Abstract	2
3	IT-projektets formål og rammer	3
3.1	FACTOR-kriteriet	3
4	Kravspecifikation for IT-løsningen	4
4.1	Funktionelle og ikke-funktionelle krav	4
4.2	Højniveau use case model	7
4.3	Tre use case modeller	7
4.4	Klassediagram	10
4.5	Sekvensdiagram	11
5	Systemdesign sammenfatning	12
6	Program- og systemtesting	13
6.1	Interne test	13
6.2	Eksterne test	13
7	Brugergrænseflade og interaktionsdesign	14
8	Versionstyring	17
9	Projektsamarbejdet	17
9.1	Hvad går godt?	17
9.2	Hvad går mindre godt?	18
9.3	Hvilke optimeringer i forhold til projektarbejdet kan vi foretage?	19
10	Reviews	20
10.1	Review: A rational design process	20
10.2	Review: Designing for usability	21
11	Litteraturliste	24

1 Indledning

Ifølge en rapport fra Danske Medier Research¹ om danskernes internetforbrug er adgang til internettet fra hjemmet steget fra 83% til 92% på blot 4 år - fra 2007 til 2011. I takt med den stigende grad af internetforbrug anvender annoncører de digitale medier til at ramme deres målgruppe.

Flere og flere virksomheder, interesseorganisationer og foreninger bliver derfor synlige på internettet med en hjemmeside eller profil på de sociale medier. De står alle overfor samme udfordring, nemlig hvordan man tjener penge på nettet. Den mest udbredte forretningsmodel er salg af annoncer på nettet. Der er to udfordringer ved at annoncere på nettet; den ene er den tekniske udfordring ved at selv skulle vedligeholde et bannersystem, og den anden er den økonomiske udfordring ved at skulle købe sig adgang til et bannersystem eller den nødvendige tekniske viden. Vores system er et forsøg på at løse disse to problemer.

2 Abstract

Projektet Easy Ad er et open source bannerreklamesystem. Formålet med Easy Ad er at udvikle et systemet der er nemt og gratis at bruge. Systemet skal kunne downloades af mindre ikke-kommercielle foreninger og interesseorganisationer, der hverken har de fornødne midler til at købe dyre systemer eller den tekniske viden til at vedligeholde et sådan system.

Easy Ad skal udvikles til et firma ved navn Treu Media. Treu Media er specialiceret indenfor annoncesalg og en lang række andre medieudgivelser. Easy Ad skal gøre det nemt for Treu Media at administrere annoncer til deres mange kunder. Systemet skal derudover løse en lang række tekniske udfordringer, som f.eks. hvordan man sikrer med hvilken vægtning et banner

¹http://www.fdim.dk/sites/default/files/mediearkiv/rapporter/danskernes_brug_af_internettet_2012_rapport.pdf

bliver vist. Eller hvordan man undgår at samme banner bliver vist flere gang på samme hjemmeside.

Simon Shine er den tekniske kontaktperson og det er ham der sætter de tekniske krav til systemet. Jan Treu-Nielsen, Treu Media's ejer, vil være kontaktperson ved aftesting og opsætning af systemet. Projektet vil løbe fra marts og indtil ultimo juni. Systemet er en prototype og vil muligvis ikke være færdig udviklet ved projektets udgang.

3 IT-projektets formål og rammer

Projektet har til formål at udvikle et open source bannerreklamesystem. Nedenfor ses projektets overordnede rammer og formål beskrevet i faktor-kriteriet.

3.1 FACTOR-kriteriet

Functionality

- Visning af bannere med forskellig vægtning
- Tidsbestemt/antalsbestemt bannervisning
- Oprettelse af kunder
- Oprettelse af grupper
- Oprettelse af bannere

Application domain

- Medarbejdere tilknyttet Treu media

Conditions

- Skal kunne bruges på tværs af browsere
- Skjult for brugere af hjemmesider
- Open source

Technology

- Web-teknologi: html, css, php, javascript.
- Databaser: MySQL
- Webserver: Apache

Objects

- Bannere/Reklamer
- Administratorer
- Kunder
- Grupper

Responsibility

- Holder styr på banner visning

4 Kravspecifikation for IT-løsningen

4.1 Funktionelle og ikke-funktionelle krav

Funktionelle krav

Funktionelle krav beskriver interaktionen mellem brugeren og systemet - uden at systemet fysisk, eller i praksis, er implementeret. Funktionelle krav siger noget om, hvordan systemet skal fungere, forventet input til systemet og forventet output fra systemet. De praktiske, eller fysiske, krav til systemet er beskrevet i ikke-funktionelle krav nedenunder.

EasyAd er et system, som gør det muligt at vedligeholde reklamer på en hjemmeside. Systemet er opbygget sådan, at det kan være ejeren der administrerer systemet, eller det er en tredje part, der administrerer systemet. Dette gør, at man altid skal kunne logge på for at få adgang til systemet og for at kunne administrere reklamerne. Når man er logget på systemet, skal man oprette en hjemmeside (eller en kunde), hvor reklamen skal vises, og derefter skal man oprette en gruppe (eller en zone), som reklamen hører til.

Til sidst skal man oprette selve reklamen og tilføje den til en gruppe.

Hjemmesiden (eller kunden), der oprettes kan være ens egen hjemmeside, eller det kan være kundens hjemmeside. I begge tilfælde skal hjemmesiden have adgang til systemet via en API, for at få vist en hjemmeside.

Grupperne bestemmer, hvor på siden, reklamen skal vises. Hvis man f.eks. kun ønsker at få en reklame vist på en underside, kan man lave en gruppe til undersiderne, mens man har en anden gruppe til forsiden.

Ikke-funktionelle krav

Ikke-funktionelle krav er direkte krav til systemet, hvor og hvordan det skal fungere. Ikke-funktionelle krav bliver også kaldt kvalitets krav, det vil sige krav til kvaliteten af det system, eller program, man udvikler. Man kan f.eks. stille krav til performance, pris, sikkerhed og mange andre ting.

Da ikke-funktionelle krav er et vidt begreb, har vi valgt at bruge FURPS+ modellen, som bogen også bruger. FURPS står for Functionality, Usability, Reliability, Performance and Supportability. Plusset står for underkategorier, som også bliver kaldt constraints.

Her er de ikke-funktionelle krav:

- **Functionality:** Systemet skal kunne håndtere (oprette, ændre og vise) reklamer på en hjemmeside. Dette kræver selvfølgelig, at alle involverede parter har adgang til internettet. Det vil sige, systemet, hjemmesiden og brugeren af hjemmesiden.
- **Usability:** Formålet med systemet er, at det skal være nemt at bruge for alle involverede parter. Både administratorer, hjemmesider og brugere. Da det er meget svært at definere, hvad er "nemt" at bruge, laver vi nogle user-tests, for at sikre os, at det faktisk er nemt at bruge. Da tiden og ressourcerne er begrænsede, vil vi ikke teste, om systemet er nemt at sætte op. Vi er nødt til at prioritere, og derfor vil vi i stedet teste, om systemet er nemt at bruge, efter det er blevet sat op. Dette vil vi teste sammen med Treu Media, hvor vi vil lave et tænkhøjt eks-

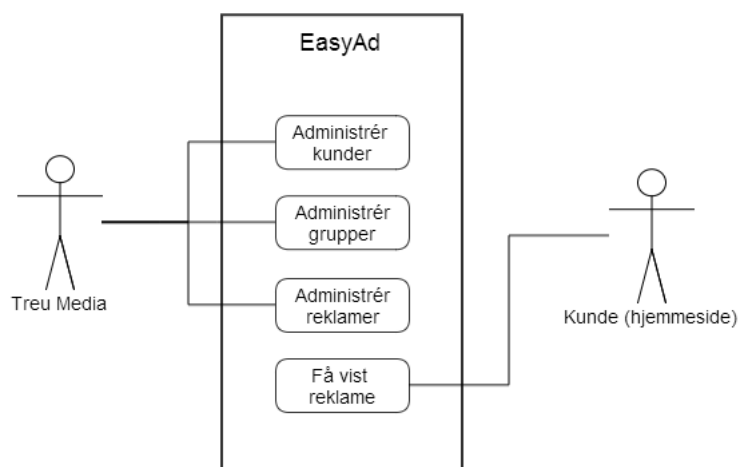
periment. Det er også vigtigt, at systemet fungerer på alle moderne browsere (IE8+, Safari, Chrome og Firefox). Dette kan nemt testes i f.eks. browserstack.com - hvis der er tid til det, vil vi udføre sådan en test.

- **Reliability:** Denne del er lidt udenfor vores kontrol, da vi ikke kan styre, hvor og i hvilket miljø systemet vil blive sat op. Der skal ikke være nogen software-forhindring for, at systemet skal kunne være tilgængeligt 24 timer i døgnet 365 dage om året. Hvis vi antager, at serveren altid har forbindelse til internettet, så sætter dette først og fremmest krav til serveren, både hardware og software. Serveren skal være dimensioneret til at kunne håndtere trafikken og softwaren skal være designet på en hensigtsmæssig måde, så den ikke bruger unødvendige ressourcer.
- **Performance:** Det er et krav til systemet, at svartiden er så lav som overhoved muligt. Der er flere faktorer, der spiller ind, men først og fremmest er det internetforbindelsen, serveren og selve softwaren. Brugeren skal ikke bemærke, at han venter på, at reklamerne loader. Dette kan gøres på forskellige måder, f.eks. ved at lade reklamerne asynkront ved hjælp af XHR-objektet i JavaScript, så siden bliver mere brugervenlig.
- **Supportability:** Systemet kan understøttes af enhver, da det er open-source. Systemet bliver udviklet i PHP, hvilket også er et open-source sprog.

Constraints: Systemet er webbaseret og vil blive udviklet i PHP. Derudover benytter vi andre webteknologier, som f.eks. HTML, CSS, JavaScript (jQuery). Projektet er open-source, men hvis man vil udvikle, eller udvide, systemet, kan man gøre det i PHP. Vi har valgt at benytte en MySQL-database. I forhold til reklamerne skal man kunne uploade almindelige billeder, video-sekvenser og flash-filer. Vi har ikke taget en endelig stilling til, hvilke filtyper man skal have lov til at uploade.

4.2 Højniveau use case model

Vi har lavet et højniveau use case diagram, som viser aktørerne og systemets funktionalitet. Treu Media har mulighed for at administrere kunder, grupper og reklamer, mens kunden (som også kan være en besøgende på en hjemmeside), får vist en reklame.



Figur 1: Højniveau use case model

4.3 Tre use case modeller

Vi har en interessent, som er Treu Media, som har fået en ny kunde og har solgt et banner til kunden. Nu skal Treu Media oprette kunden i systemet, oprette en gruppe og oprette en reklame.

Vi starter med den første use case, hvor administrator skal oprette en kunde:

Use case navn: Opret kunde

Aktør: Treu Media

Flow:

1. Tryk på "Customers"
2. Tryk på "Add customer"
3. Udfyld felterne "Name" og "Url". Begge felter skal udfyldes
4. Tryk på "Submit"

Startbetingelser: Treu Media skal være logget på systemet

Slutresultat: En kunde er oprettet i systemet

Derefter tager vi en use case, hvor administrator skal oprette en gruppe, som tilhører en kunde:

Use case navn: Opret gruppe

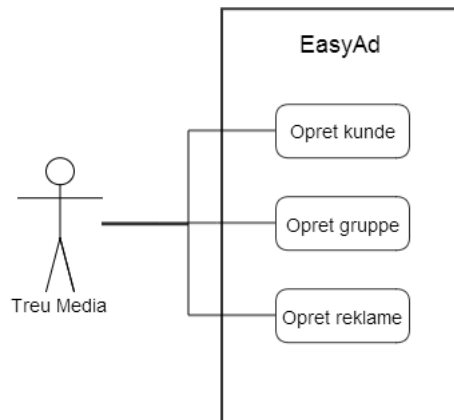
Aktør: Treu Media

Flow:

1. Opret en kunde. Se tidligere use case
2. Tryk på "Groups"
3. Tryk på "Add group"
4. Udfyld feltet "Group name" og vælg en kunde fra select-boksen
5. Tryk på "Create"

Startbetingelser: Treu Media skal være logget på systemet. Der skal være oprettet en kunde

Slutresultat: En gruppe er oprettet i systemet



Figur 2: Administrator skal oprette en reklame

Til sidst tager vi en use case, hvor administrator skal oprette en reklame, der tilhører en gruppe.

Use case navn: Opret reklame

Aktør: Treu Media

Flow:

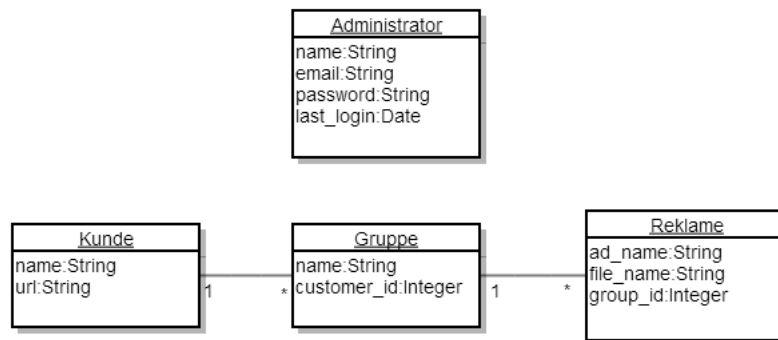
1. Opret en kunde. Se tidligere use case
 2. Opret en gruppe. Se tidligere use case
 3. Tryk på "Ads"
 4. Tryk på "Add ad"
 5. Udfyld feltet "Ad name" og vælg en kunde fra select-boksen og vælg siden, hvilken gruppe reklamen skal høre til fra select-boksen
 6. Vælg reklamen fra din computer ved at trykke på "Vælg fil"
 7. Tryk på "Submit"
-

Startbetingelser: Treu Media skal være logget på systemet. Der skal være oprettet en kunde og en tilhørende gruppe.

Slutresultat: En reklame er oprettet i systemet

4.4 Klassediagram

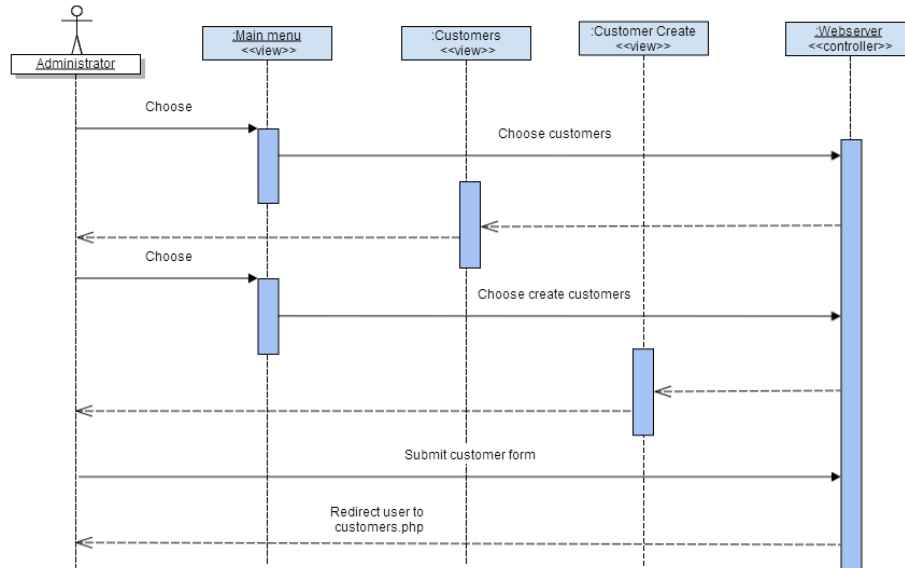
Vores klasser er følgende: administrator, kunde, gruppe og reklame. Forholdet mellem klasserne er, at flere grupper kan tilhøre én kunde og flere reklamer kan tilhøre én gruppe. Derved kan flere reklamer også tilhøre én kunde.



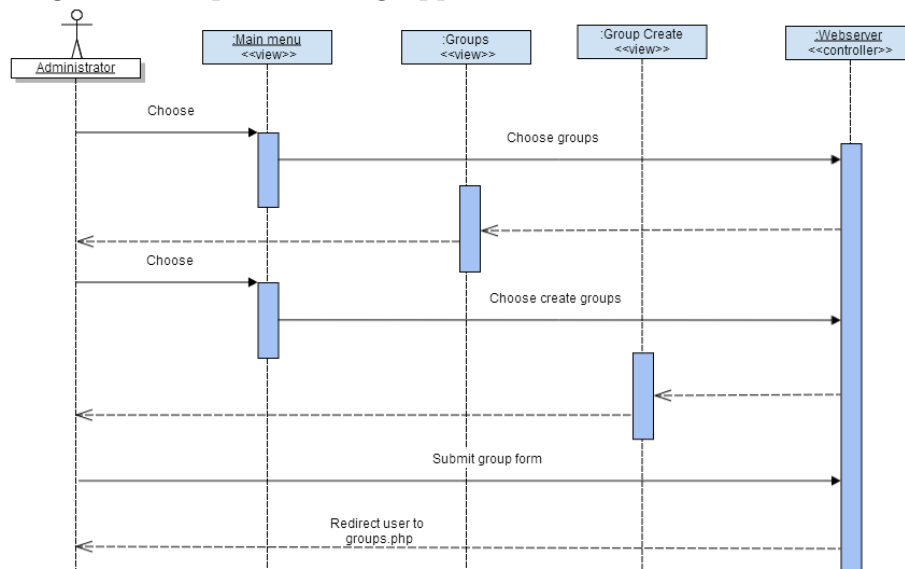
Figur 3: *Klassediagram*

4.5 Sekvensdiagram

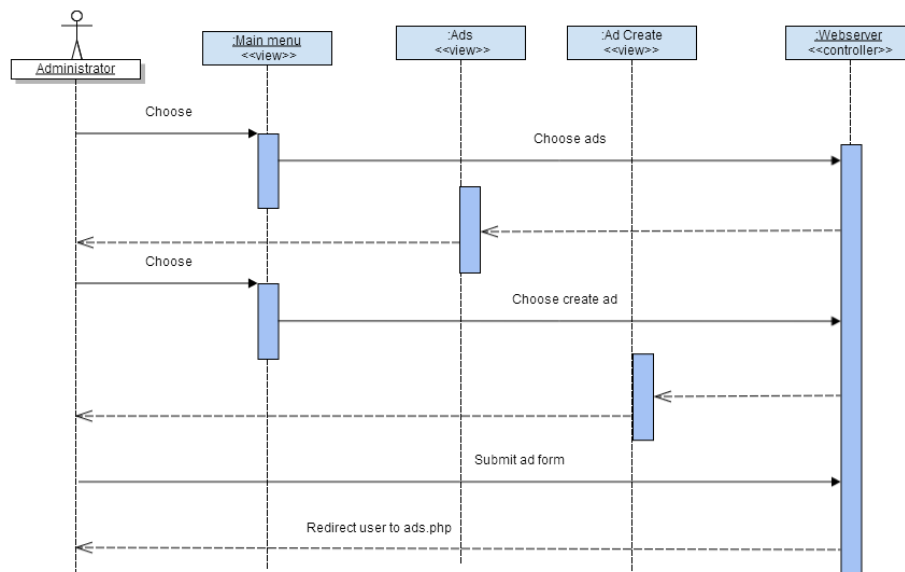
Sekvensdiagram over oprettelse af kunde.



Sekvensdiagram over oprettelse af gruppe



Sekvensdiagram over oprettelse af reklame



5 Systemdesign sammenfatning

Vi er nået dertil, at Treu Media kan oprette kunder, grupper og reklamer. Vi har lavet en grov skitse af systemet, men mangler at implementere nogle funktioner, som f.eks. at give Treu Media mulighed for at ændre en kunde, gruppe og reklame. Vi skal også sørge for at beskytte imod SQL-injections og sikre os, at de krævede felter faktisk bliver udfyldt, når Treu Media f.eks. opretter en reklame. Den vigtigste funktion vi mangler, er at vise en reklame på en hjemmeside. Der findes flere muligheder for, hvordan vi kan lave denne del af systemet. Vi har diskuteret fordele og ulemper ved de forskellige muligheder, og har ikke truffet en endelig beslutning. Højest sandsynligt vælger vi en løsning, hvor kunden skal tilføje en javascript-fil til hjemmesiden, som henter og placerer reklamerne på hjemmesiden.

6 Program- og systemtesting

Projektgruppen ønsker at teste systemet i flere omfang. Disse omhandler både interne og eksterne kravspecifikationer.

6.1 Interne test

Ved de interne test skal vi benytte os af unit testing. Denne testing skal finde sit ståsted i matematikken, og vil omhandle en form for tællemekanisme. Gruppen ønsker længere fremme i projektet at teste antallet af visninger af bannere på hjemmesider. Vi skal altså forsøge at skabe en bannervisning internt i programmet et kendt antal gange. Vores tællemekanisme skal således beregne et resultat udfra vores banneres visning. Dette resultat skal herefter sammenlignes med det forventede antal visninger fra start.

En konkret måde at løse gruppens problem her, kunne være på følgende måde: uploade et banner, sende fiktive requests til en hjemmeside med dette banner. Herved kunne gruppens tællemekanisme aflæse om requesten bliver behandlet korrekt og dermed registrere en visning. Dette ville resultere i at man kan tælle antallet af requests som der er blevet behandlet i processen og gruppen ville hurtigt kunne se om det forventede resultat stemte overens med det relle resultat fra vores tællemekanisme.

6.2 Eksterne test

Projektgruppen regner desuden med testing af systemets nuværende GUI. Gruppen forsøger at skabe et ”tænke højt forsøg” som indebærer at interessenten benytter sig af systemet. Vi vil på forhånd give interessenten systemet i hænderne, og guide denne i en retning med henblik på testing af brugervenligheden og korrekthed af vores GUI. Der vil på forhånd være stillet en række opgaver, som i projektgruppens øjne er set som de væsentligste faktorer for systemets kunnen (use cases). Disse små opgaver skal interessenten

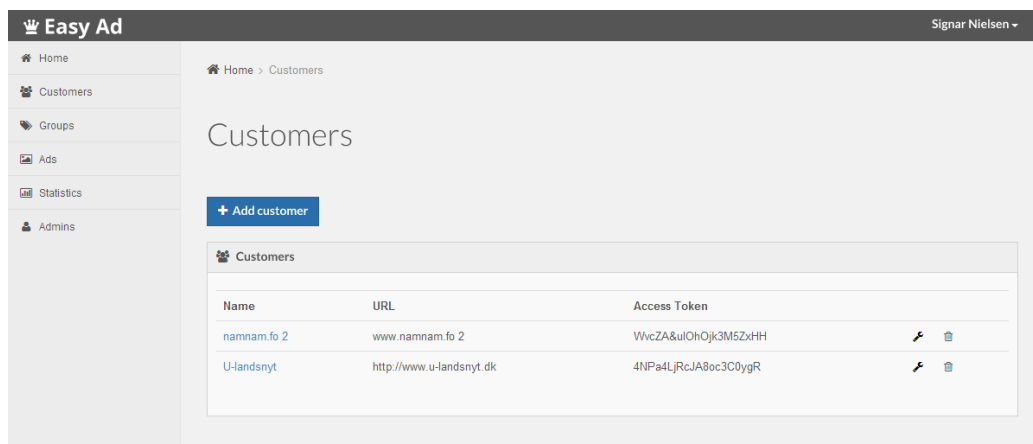
dernæst udføre, eller prøve at udføre på egen hånd. En form for dokumentering behøves.

Da projektgruppen ikke burde være til stede under testingen af programmet, ville man benytte sig af f.eks. videooptagelse så interessenten kunne afprøve systemet uden vores indflydelse eller kommentarer om systemet. Her ved ville vi få et bedre overblik og en viden om hvordan mennesker håndterer systemets kunnen. Under dette forsøg ville de eksterne kravspecifikationer uden tvivl komme i spil. F.eks. brugervenlighed ift. overblik fra interessentens side.

7 Brugergrænseflade og interaktionsdesign

Skærmbilleder

Oversigt og oprettelse af kunde



Figur 4: *Oversigt over kunder*

Oversigt og oprettelse af reklamer

Statistik for reklamer

Easy Ad Signar Nielsen ▾

Home Customers Groups Ads Statistics Admins

Home > Customers > Add customer

Customers

Customers

Name

URL (Customer's website)

[Create customer now](#)

Figur 5: Oprettelse af kunde

Easy Ad Signar Nielsen ▾

Home Customers Groups Ads Statistics Admins

Home > Ads

Ads

[+ Add Ad](#)

Ads

Name	Customer	Group	Impressions	Deadline	
Card tournament	namnam fo 2	frontpage	100 / 6	26-05-2014	<input checked="" type="checkbox"/>
Holstee Manifesto	namnam fo 2	subpage left	6	22-05-2014	<input checked="" type="checkbox"/>
Jeans	U-landsnyt	forsiden	31	09-05-2014	<input checked="" type="checkbox"/>
Kort turnering	U-landsnyt	forsiden	20 / 31	01-08-2014	<input checked="" type="checkbox"/>

Figur 6: Oversigt over reklamer

Audio visuel præsentation

Vi har lavet en demonstration af vores brugergrænseflade, som vi har uplo-
 adet til youtube:

www.youtube.com/watch?v=RXtx08uCdH4&feature=youtu.be&hd=1

Easy Ad Signar Nielsen

Home Customers Groups Ads Statistics Admins

Home > Ads > Add ad

Ad

Add ad

Ad name *

Max impressions

Deadline for ad

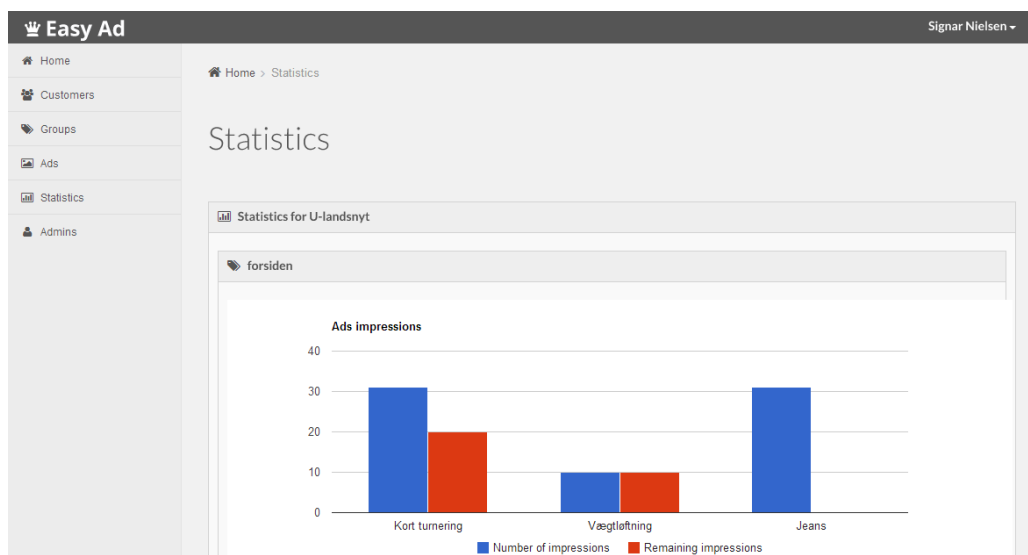
Customer name *

Group name *

Choose file to upload *
 Der er ikke valgt nogen fil

* Required field

Figur 7: Oprettelse af reklame



Figur 8: Statistik for reklamer

Tænke-højt forsøg

Vi har ikke udført et tænke-højt forsøg endnu, men det har vi planer om at nå til næste rapport.

8 Versionstyring

Vi benytter GitHub til versionstyring. Vores projekt ligger på <http://github.com/AslakNiclasen/ProjektOpgave>.

Da vi committer flere gange i løbet af en programmerings-session, giver det ikke rigtig nogen mening at vedhæfte vores commit-log fordi den er så lang. Den er også offentligt tilgængelig, så alle kan gå ind og se den. Vi har heller ikke valgt at vedhæfte vores kode, da den også er offentlig.

Vi har valgt at commite tit, så vi undgår risikoen ved at en stor del af arbejdet går tabt, hvis uheldet er ude. Grunden til, at Aslak ikke har committed lige så ofte, som de andre er fordi, at han har problemer med Git på sin computer.

9 Projektsamarbejdet

9.1 Hvad går godt?

Internt

Projektgruppen har i løbet af påsken mødtes en gang om ugen, og disse har været meget produktive. Til disse møder har vi haft en dagsorden, vi har planlagt små arbejdsopgaver frem til næste mødes dagsorden og vi har planlagt opgaveskrivning individuelt. Arbejdet mellem gruppen internt, fungerer meget gnidningsfrit og alle opfylder hindandens krav, og de krav der stilles til opgaven. Alle møder velforberedte og friske, således at dagen hvor vi alle har mulighed for at mødes bliver så konstruktiv som overhoved muligt.

Det var en smule kompliceret at finde en god mødefordeling i påskeferien, da alle 3 medlemmer havde personlige ærinder der skulle imødekommes. Det har gjort at vi til dels har været tvunget til at skrive anden delrapport i punkter individuelt hver for sig. Vi mødes af den grund mandagen inden aflevering

til korrektur læsning, og forståelses/formidlingsmæssige usikkerheder. Alle i gruppen er blevet tildelt ca. lige meget rapportskrivning, så ingen i gruppen føler sig overbebyrdet, eller omvendt, overset.

Med interessant

Vi har ikke haft et møde med vores interessant til dags dato, men vi har haft 2 møder og mailkorrespondance til vores mellemlid mellem os og interessen. Interessen giver klart udtryk for et frugtbart samarbejde, med mulighed for fremtidige møder efter vores ønske - det kommer vi selvfølgelig til at benytte os af.

Vi har på nuværende tidspunkt en GUI som fungerer efter hensigten, og dette er også grunden til at vi ikke ville mødes med interessen inden. Projektgruppen ønskede at have noget brugbart rent fysisk at vise frem, således at interessen ville have nemmere ved at sætte sig ind i projektets omfang, og herunder forstå projektet. Dvs. at vi til dette møde kunne fremsætte klare retningslinier med det fokus at interessen ville kunne udføre et "tænke højt forsøg" (testing af GUI) og hermed danne sig et billede af systemet. Vi ville af dette opnå stor viden, da vi med en kontrolleret testing af vores system kan fusionere tekniske aspekter med menneskelige input.

Med denne viden kunne vi videreudvikle vores system, således at der måske sættes fokus på brugervenlighed, eller andet. Det er op til os og interessen i samarbejde.

9.2 Hvad går mindre godt?

Internt

Som benævnt i punktet ovenover, ville det være mere hensigtsmæssigt at mødes flere gange i påsken. Men det har desværre ikke været muligt. Vi synes dog at vi har fået det optimale ud af vores samarbejde. Et medlem i

gruppen havde store problemer med github, eftersom hans system er Mac OS X. Det gjorde at han måtte ty til andre løsningsmidler og gruppen har således ikke mistet nogle former for data. Men man blev derfor tvunget til at sende stoffet til de andre medlemmer i gruppen således at de kunne overføre data til github som planlagt. Derfor ses det også i "contributions" fra denne person. Heldigvis burde problemet være løst med en virtuel maskine som benytter Windows på personens Mac OS X i stedet for.

Med interessant

I og med at gruppen ikke har haft møde med interessenten, er det svært at foretage en vurdering af dette punkt. Vi synes dog i og med at interessenten viser meget interesse og er nem at få kontakt til telefonisk, at vi ikke har nogen problematik her.

9.3 Hvilke optimeringer i forhold til projektarbejdet kan vi foretage?

Internt

Vi vil holde lidt flere møder fremover, alt efter behov. Gruppen arbejder velfungerende i det mønster som den har sat sig i pt. Så planlægningen og strukturen af arbejdet forbliver uændret indtil videre.

Med interessant

Gruppen begynder planlægning fremover telefonisk med interessenten, og så vil der fra møde til møde indgå klare dagsordner hvis projektet skal fungere optimalt når gruppen skal diskutere emner mere dybdegående med denne.

10 Reviews

10.1 Review: A rational design process

Artiklen beskriver den rationelle design process og hvorledes denne er uopnåelig. Dette skyldes i høj grad uvished og menneskelige fejl, men problemet opstår også ved kompleksiteten af store programmer [2, side 251]. Denne kompleksitet gør det svært at overskue processen og alle dens detaljer, hvilket fører til menneskelige fejl. Man kan løse dette problem med brugen af dokumentation. Artiklen fokuserer på, hvordan man skaber en fiktiv rationel process, som man herefter følger bedst muligt [2, side 252]. Dette kaldes en forfalskning af design processen. Ovenstående kan gennemføres ved brug af forskellige former for dokumentation, hvoraf den vigtigste er et requirements document. Dette dokument indholder informationen om projektet klart og præcist overordnet - ikke mere [2, side 252-253]. Derudover findes der modul guides som definerer ansvar mellem de medvirkende i et projekt.

Til sidst i artiklen sættes der fokus på en diskussion omkring esencen i at dokumentere software før under og efter en designprocess. Der fokuseres på hvilke problemer der opstår, hvilke styrker og svagheder der findes i dokumentationen, og hvordan man optimere styrkerne, og forminsker svaghederne mest muligt, sådan at målet, den rationelle design process, bliver realiseret så meget så muligt [2, side 255].

Det er tydeligt at se, hvordan opfattelsen af at løse et problem mest effektivt er meget bred. Der findes utallige muligheder, som alle har deres styrker og svagheder. Vi oplever dog samtidigt, at alle disse mulige fremgangsmåder har enkelte dele, som synes at være ens. Disse teoretiske og også praktiske dele er helt essentielle for alle opfattelser. De ses også her i denne artikel. Et godt eksempel på dette er artiklens beskrevne requirements document. Dette er i og for sig knyttet til forfatterens opfattelse af at løse et problem, men hele kernen ved brugen af dette dokument, sætter et øget fokus på vigtigheden af dokumentet. Vi finder samme typer af dokumenter i både agile arbejdsme-

toder, som f.eks. Scrum metoden, men også i mere lukkede arbejdsmetoder som f.eks. vandfaldsmetoden. I vores projekt danner disse delrapporter i samme forstand en forståelse for arbejdsmetoden vi benytter. Et dokument der fungerer som en helhed. Et dokument der præcist beskriver, hvad målet og dets måde at opnå dette på, bliver yderst centralt i enhver design process. Hvad end man er en enkelt person, eller flere 100 mennesker i et projekt, så er denne type dokument altid omdrejningspunktet, som man vender tilbage til.

10.2 Review: Designing for usability

”Designing for Usability: Key Principles and What Designers Think- Kursiv beskriver tre hovedprincipper inden for system design. Artiklens forfattere John D. Gould og Clayton Lewis er fortalere for disse tre principper. De har i en lang årrække indenfor systemdesign benyttet sig af disse. Artiklen beskriver først og fremmest de 3 principper overordnet, og sætter dernæst fokus på hvordan man, af flere grunde, ikke bruger dem i praksis. Forfatterne vælger udfra disse grunde at underbygge og fremane en bedre forståelse for principperne. Til sidst i artiklen tages der et reelt projekt fra IBM og sættes i kontrast principperne, da projektets design process har benyttet sig af disse.

De tre principper som beskrives i artiklen er:

- Tidligt fokus på brugere og deres opgaver.
- Emperiske målinger: observationer, analyser og registreringer af brugere i systemet.
- Iterativt Design: et system design som løbende i processen udvikles i takt med at projektet forløber.[1, side 300]

Problemerne ved systemdesignere i praksis: Som nævnt underbygger forfatterne argumenter for en bedre forståelse af principperne. De tre principper burde være implicite for designerne, men desværre er det ikke tilfældet. Det problem adresseres også i section 2 i artiklen, hvor forfatterne har interviewet en større gruppe designere ift. deres erfaringer om systemdesign. Og det viser

sig generelt, at designerne ikke benytter sig af principperne.[1, side 301]

Det store problem ved designprocessen munder ud i, at designerne ikke benytter alle tilgængelige virkemidler optimalt. Deraf bliver det endelige design ikke er fuldt ud så godt som det kunne blive. Ydermere nævnes der forskellige typer af problemer til grund for dette store problem. F.eks. at man som designer ikke ser princippet som værende værd at følge; misforståelse af begreber som er næsten identiske, men har kritiske forskelligheder i praksis; brugerinteraktionen og udviklerinteraktionen nedprioriteres og underestimeres; man føler at principperne er upraktiske og uhåndterlige som designer. [1, side 303]

De tre principper er hele tiden omdrejningspunktet for artiklen. Men det er en nødvendighed, da de sættes i kontrast til andre principper for system design og projektudvikling, som f.eks. "Design guidelines" og "getting it right the first time". [1, side 304]

Ligesom i den første artikel, ser vi også at processen for et design har mange muligheder, og man kan ikke enes om hvad der er bedst. Dette problem beskrives også af forfatterne (principles of design are arguable, including these)[1, side 303], men samtidigt bruger de igen argumenter til at understøtte hvorfor deres principper burde følges.

Vi finder disse principper andre steder. F.eks. den iterative designprocess ses i alle former for agile arbejdsprocesser, da hele omdrejningspunktet bliver optimering på baggrund af testning. Denne cyklus skaber et frugtbart resultat sammen med de andre principper i IBM's ADS udviklingsproces, beskrevet i artiklen.[1, side 309]

Også i vores projekt finder vi paralleler til artiklens synspunkter. Herunder de sidste to principper: Værd at nævne er en af de former for empiriske målinger projektgruppen kommer til at benytte. Artiklen nævner: "Users should be given simple tasks to carry out, and their performance, thoughts, and attitudes should be recorded and analyzed"[1, side 302]. Det er noget projektgruppen kommer til at gøre med interessenten i et tænke højt forsøg

lidt længere fremme i projektet. Desuden vil unit testning og andre interne systemtests kunne foretages, hvilket vil kunne placeres under en løbende og skiftende design process med testning, også benævnt som Iterativt Design.

11 Litteraturliste

Litteratur

- [1] John D. Gould, Clayton Lewis. *Designing For Usability: Key Principles And What Designers Think*. Communcations of the ACM: 28 (March 1985)
- [2] David Lorge Parnas, Paul C. Clements. *A Rational Design Process: How and Why to Fake It*. IEEE Transations On Software Engineering: SE-12 (Februar 1986)