

ProjektKursus Systemudvikling 2014

Delrapport 4

28. maj 2014

Gruppemedlemmer:

Kenneth Christensen: 02 08 93

Michael Jensen: 01 07 93

Rune Pedersen: 01 11 82

Rasmus Hansen: 03 12 92

Instructor: Kasper Passov

Indhold

1	Abstract	3
2	IT-projektets formål og rammer	4
3	Kravspecifikation for IT-løsningen	5
4	Systemdesign sammenfatning	16
5	Program- og systemtest	17
6	Brugergrænseflade og interaktionsdesign	18
7	Versionstyring	21
8	Projektsamarbejdet	22
9	Review	23
10	Litteratur	27
11	Bilag	28

1 Abstract

We have choosen to work on a project for a local bike shop called "bmx-butikken". Bmxbutikken is specialised in the freestyle BMX community. They sell all you need as a BMX rider, which is everything from cloths and shoes to complete bikes and parts. They have a phsyical shop on Nørrebrogade, and a webshop at www.bmxbutikken.dk. The project regards a Spotguide app for all devices, programmed as a web application. This site is supposed to help local riders to find new places to ride, and foreigners to find any places to ride and hook up with the local crew. The app can also show you where bmx-butikken is located if you need a spare part. The application is designed for all bmx riders mainly in Copenhagen, but spots can be uploaded from anywhere.

2 IT-projektets formål og rammer

Følgende model er en FACTOR analyse af projektet og giver en kort og konkret beskrivelse af rammerne for projektet. Modellen er udarbejdet efter inspiration fra et uddrag af Oriented Analysis og Design (2.7 The FACTOR Criterion).

F - Functionality: beskriver hvilke værktøjer og funktionaliteter kunden får igennem appen.

A - Application Domain: beskriver hvem vi prøver at ramme med projektet og altså hvem der vil have gavn af vores arbejde.

C - Conditions: beskriver hvilke krav der er til kørsel af projektet (applikationen), det være sig brugerens udstyr samt appens tilgængelighed.

T - Technology: beskriver hvilke udviklings værktøjer vi vil gøre brug af undervejs i projektet.

O - Objects: indeholder de objekter vi arbejder med. Disse objekter er en del af løsningen til problemområdet.

R -Responsibilities: Hvad er applikationens rolle, pligter og hvad er den grundlæggende løsning til problemet.

Funktionalitet	Give oplysninger om bmx spots Give oplysning om specialiserede bmx butikker Give rutevejledning
Application Domain	BMX udøverer og bmxbutikkens kunder
Conditions	Hjemmesiden skal kunne køres på smartphones Åben internet forbindelse Adgang til din lokation
Technology	HTML Javascript CSS mySQL PHP
Objects	Skate Spots Shops BMX Spots
Responsibilities	Give information om spots Informerer om selve bmxbutikken

3 Kravspecifikation for IT-løsningen

I denne sektion beskriver vi hvad der er funktionelle og ikke-funktionelle krav. De **funktionelle krav** beskriver hvad i systemet der kræver deciderede funktioner der modellerer med vores database. Altså hvordan system skal virke. Dette er vigtigt da de netop fortæller om funktioner der skal være i systemet og giver et indblik i strukturen.

De **ikke funktionelle krav** specificerer de krav der er for systemet der ikke har en direkte indvirkning op programmet. Dog er de en vigtig faktor da det er disse der er med til at skabe interfacet, altså hvordan det ser ud og hvordan det føles at navigere igennem det hele. Altså det hvordan de vil vise det systemet gør.

Constraints er hvilke begrænsninger vi har på vores system, altså hvordan tingene skal være opbygget. Disse er med til at danne en mindre struktur over hvad der er meget vigtigt for vores system.

Funktionelle krav

Add spot

Gør det muligt for en bruger at tilføje nye spots til kortet. Dette er til for at kortet hele tiden kommer til at udvide sig med diverse nye spots som brugerne kan finde og dele med andre. Det fungerer sådan at man udfylder en formular med lokalitet og beskrivelse og man har samtidig mulighed for at uploade billeder. Dette er et krav idet at der vil dukke nye og spændende spots op og appen skulle helst være op to date

Settings

Gør det muligt at bestemme hvor stor en radius man vil se spots i udfra ens egen lokation. Dette er især vigtigt hvis man rejser på cykel og gerne vil begrænse hvor langt man skal rejse.

Search

Gør det muligt for brugeren at finde et specifikt spot udfra dens navn. Så det bliver nemmere og hurtigere for brugeren at finde lige det spot han/hun leder efter.

Streetspot

Viser brugeren en liste over de forskellige spots med billede. Dette er til for hvis brugeren ikke vil se diverse spots på kortet, men finder det mere overskueligt at se det på en liste.

Markers

Viser på kortet 3 ting: Brugerens lokation, diverse spots og bmxbutikkens lokation. Disse markers er nødvendige for at man kan se spots'ne på kortet.

Bruger lokation

Finder brugerens lokation. Er nødvendig i forhold til at beregne forskellige funktioner ud fra brugerens placering

Shops

Skal vise en liste over de shops der er tilknyttet projekt(pt er dette dog kun en butik). Er et krav idet at der ønskes at brugerne er opmærksomme på at de pågældende butikker findes og hvor de er.

Ikke funktionelle krav:

Sprog

Siden skal være på engelsk, så den kan bruges af turister, men samtidig også af de lokale da de fleste i miljøet snakker engelsk.

Menu

Der skal være en menu der indeholder 5 ting du kan trykke på der kan vejlede bruger: Spots, shops, settings, Add spot, Search

Usability

Skal være til at bruge af alle at brugerer. Nemt og intuitivt at finde rundt i.

Reliability

Skal være oppe og klar til brug, altid.

Performance

Skal kunne holde til så mange brugere som muligt, altså ikke begrænse hvor mange der kan være på samtidigt med det fungerer hurtigt.

Farve af Markers

De forskellige typer af markers skal have forskellige farver så brugeren nemmere kan overskue det.

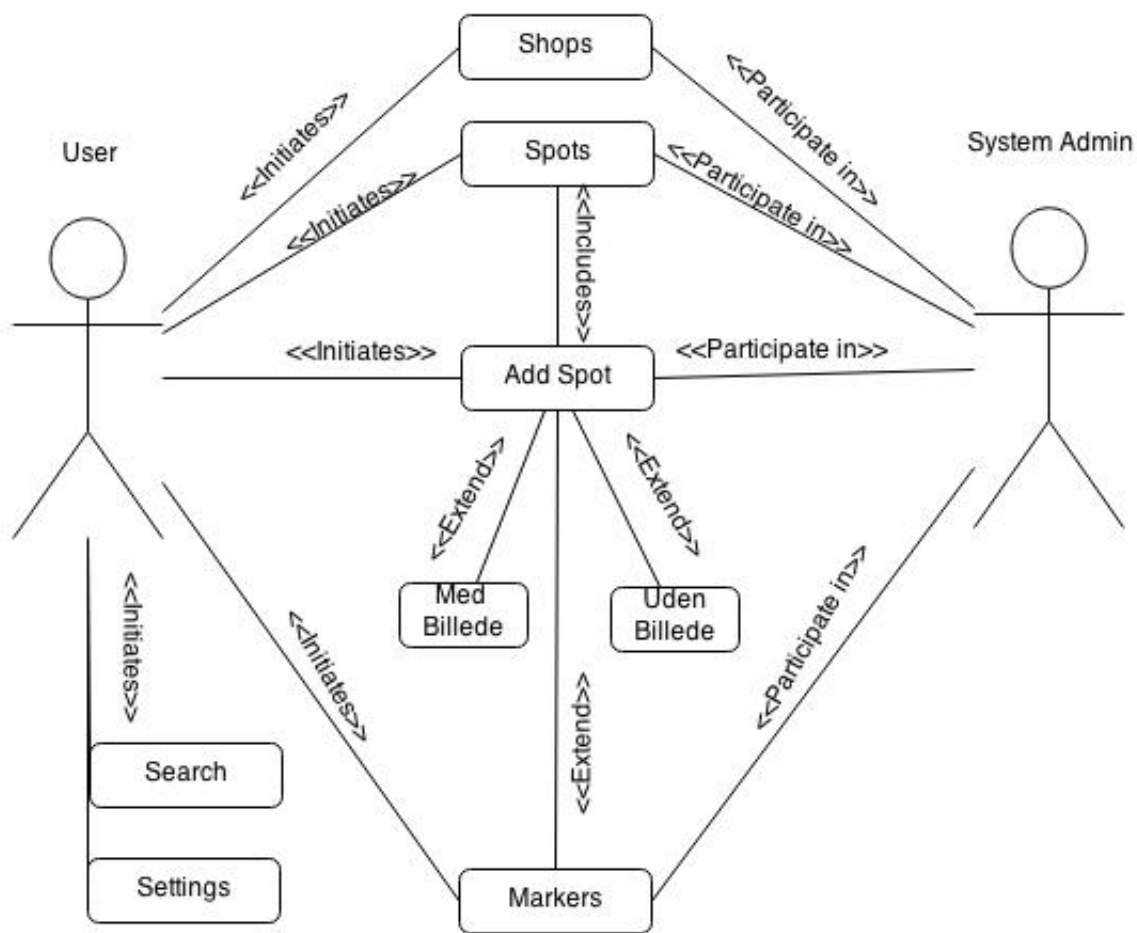
Constraints

Systemet skal være kompatibelt med både computere og telefoner.[Implementation requirement]

Kortet skal vise en marker ved bmxbutikken[User Interface Requirement]

Koden skal være skrevet i php/javascript/html/mysql.[Implementation requirement]

(b) En use case model, der beskriver system-funktionaliteten. Der ønskes et højniveau-diagram, som giver overblik over hvilke use cases de forskellige aktører har



Højniveau diagram der skal illustrere de forskellige aktører

Som det ses er der 2 aktører, brugeren og System administratoren. Brugeren benytter appen og det er altså brugeren der "initiates" eller starter en given funktion, det være sig: At klikke på Add Spot og tilføje et spot uden billede eller at ændre indstillingerne. Som det ses på modellen er alle funktioner brugeren har adgang til og brugeren selv forbundet.

Derudover har enkelte funktioner noget med hinanden at gøre. Som det ses er "Add Spot" forbundet med Markers, Spots, Med billede og uden billede. Dette er fordi at man både kan tilføje nye spots uden eller med billeder. Samtidig er det disse spots der kommer frem på kortet som "Markers" og det nye spot bliver inkluderet under den generelle mappe for "Spots".

Til sidst er der System administratoren som er forbundet med Shops, Spots, Add Spot og Markers. Dette skyldes at System Administratoren er med til at opdatere de givne funktioner. For eksempel når brugeren tilføjer et nyt spot, så kommer det til godkendelse hos system admin og derefter ligges spottet ind i Spots (eller shops) og spottet får en markør på kortet.

(c) Tre specificerede use cases, som er særlig vigtige i jeres system

Use-cases

Få information om et spot

Use case name: FindSpot

Actor: en bruger der gerne vil finde information om et spot

Entry condition: brugeren åbner appen

Flow of events:

1. Brugeren bliver præsenteret for et kort med sin egen position i centrum og en menu linje i toppen
 2. Brugeren trykker på en "marker" på kortet for det spot han vil se information.
-

Exit condition: Brugeren ser nu et pop-up vindue med information om spotet.

Dette use-case beskriver en bruger som gerne vil have nogen information om et specifikt spot. Brugeren åbner app'en på sin telefon og bliver præsenteret med kort-interfacen. Kort-interfacen er kort over brugerens position med en menu-bar i toppen. På kortet vil der være faner som repræsenteret spots. Bruger trykker på fanen for det spot han gerne vil have informationer og app'en viser så disse information i et pop-up vindue henover kortet.

Tilføj spot

Use case name: AddSpot

Actor: en bruger der gerne vil tilføje et spot

Entry condition: brugeren åbner appen

Flow of events:

1. Brugeren trykker på knappen "Add spot", der sender ham videre til en udfyldelses formular
 2. Brugeren udfylder formularen og gemmer denne, formularen sendes derefter til godkendelse hos administratoren
-

Exit condition: Brugeren sendes retur til forsiden

Dette use-case beskriver en bruger som gerne vil tilføje et spot til app'en. Brugeren åbner app'en på sin telefon og bliver præsenteret med kort-interfacen. Kort-interfacen er kort over brugerens position med en menu-bar i toppen. Brugeren trykker på "Add spot"knappen i menuen og bliver herefter præsenteret med en formular som skal udfyldes. Denne formular består af tekst-felter samt muligheden for at tilføje et billed. Brugeren udfylder formularen og submitter den, hvorefter brugeren bliver returneret til kort-interfacen.

Ændre indstillinger

Use case name: ChangeCondition

Actor: en bruger der gerne vil ændre indstillingerne

Entry condition: brugeren åbner appen

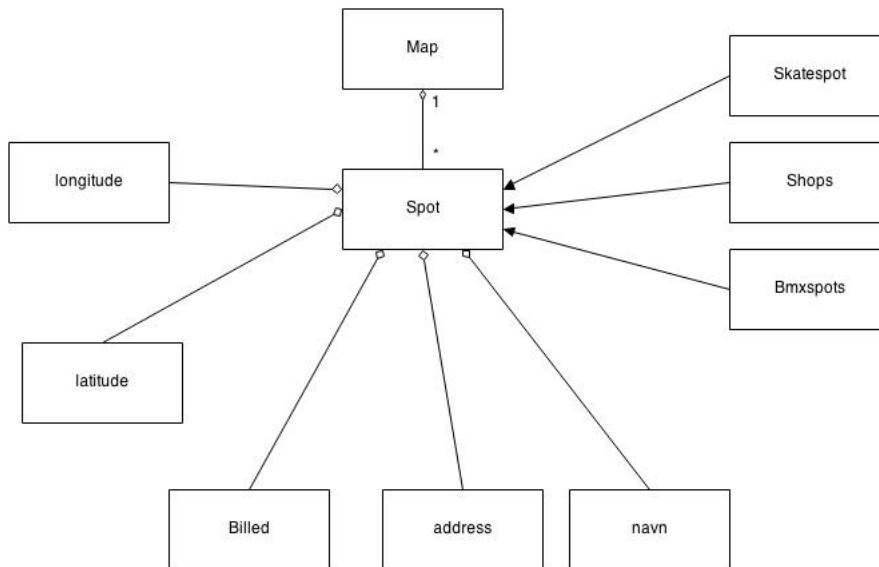
Flow of events:

1. Brugeren trykker på indstillinger, der sender brugeren videre til en ny side
 2. Brugeren bliver præsenteret for de forskellige indstillinger han kan vælge
 3. Når brugeren har ændret indstillinger er der en ok knap som tilføjer indstillingerne
-

Exit condition: Indstillingerne er nu i funktion og brugeren sendes tilbage til forsiden

Dette use-case beskriver en bruger som gerne vil ændre den radius som app'en loade spots fra. Brugeren åbner app'en på sin telefon og bliver præsenteret med kort-interfacen. Kort-interfacen er kort over brugerens position med en menu-bar i toppen. Brugeren vælger indstillinger fra menuen og bliver præsenteret med slider-bar. Brugeren vælger den ønsket radius og trykker "accept". Brugeren bliver herefter returneret til kort-interfacen.

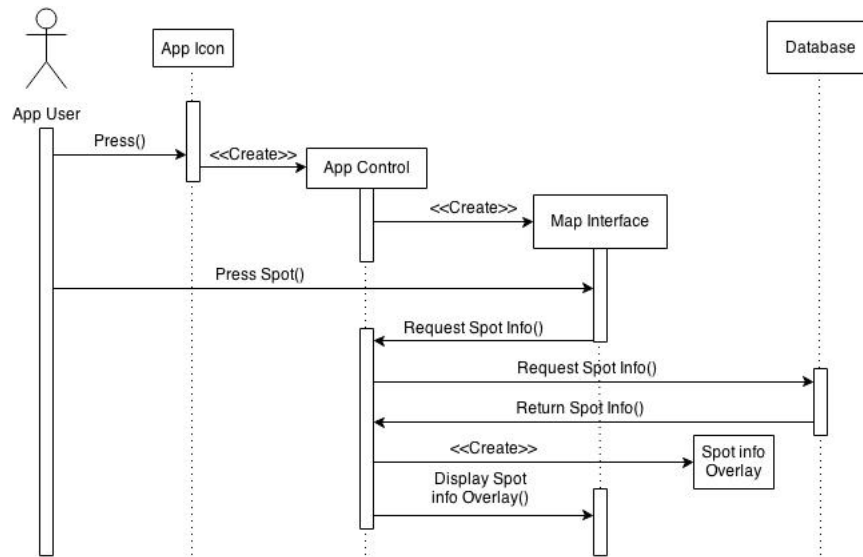
(d) Et klassediagram over jeres problemområde (solution-domain)



Klasse diagrammet viser systemets opbygning: Vi har et kort, dette kort har et "et-til-mange"forhold til vores spots. Disse spots kan være af tre typer: Bmx-spots, skate-spots og shops. Disse tre nedarver fra "spots"som har nogle attributer. Disse attributer er : Latitude, longitude, adresse, navn og muligvis et billed.

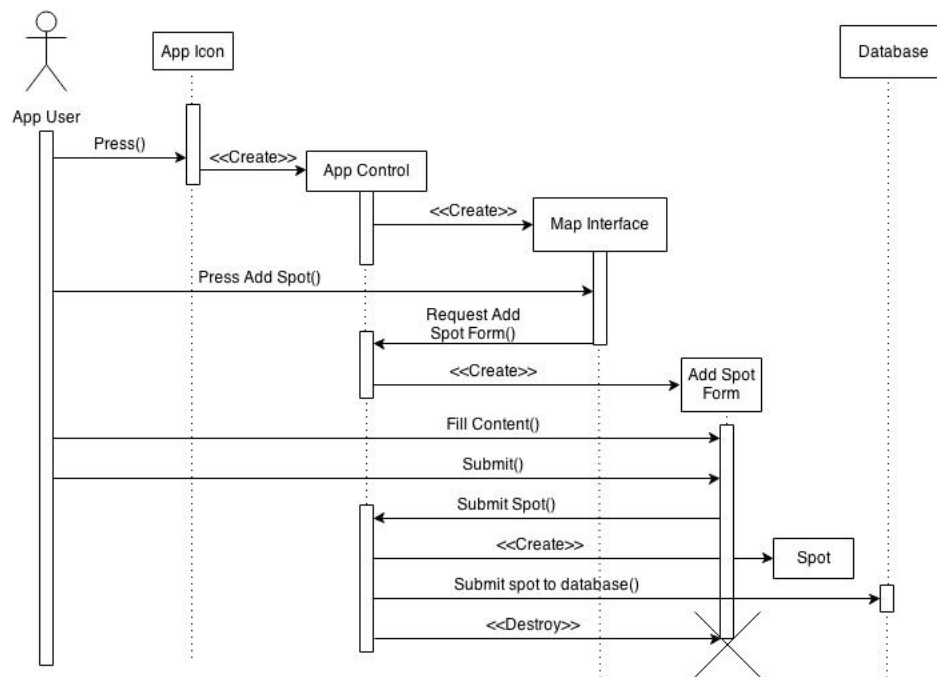
(e) Sekvens-diagrammer over de 3 use-cases specificeret i punkt (c)

Husk at alle diagrammer skal være fulgt af tekstbeskrivelser, der gør diagrammerne fuldt forståelige også for læsere uden særligt domænekendskab.



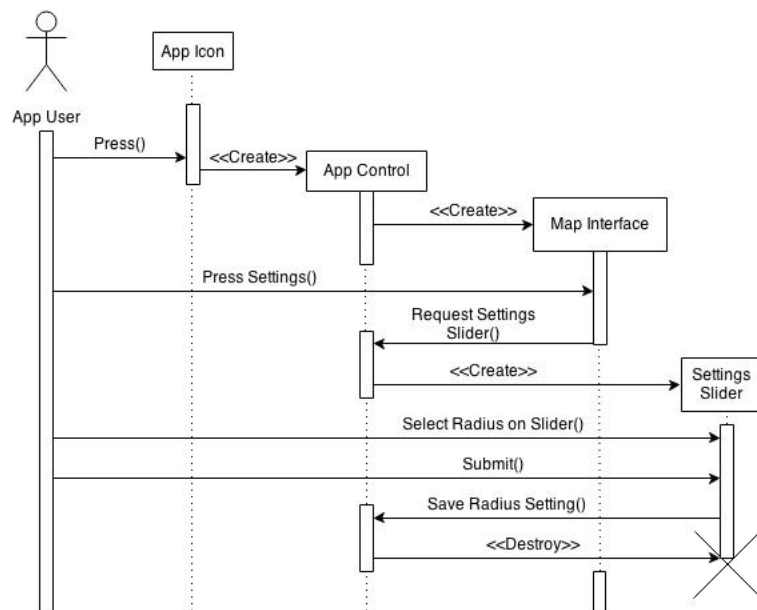
Sekvensdiagram over find spot information:

Brugeren starter App'en, ved at trykke på iconet for App'en, på sin telefon. App'en viser kort interfacen, som er et kort over brugerens nuværende position med en menu-bar i toppen. Spotsne vil blive repræsenteret på kortet i form af faner. Brugeren vælger et spot fra kortet ved at trykke på en fane. App'en henter informationerne om spottet, fra database, og viser dem i et overlay hen over kortet.



Sekvensdiagram over add spot funktionen:

Brugeren starter App'en, ved at trykke på iconet for App'en, på sin telefon. App'en viser kort interfacen, som er et kort over brugerens nuværende position med en menu-bar i toppen. Brugeren vælger add spot fra menuen, ved at trykke på knappen markeret "Add Spot" i menu-baren. App'en viser en add spot formular, som er en række tekst-felter der skal udfyldes, samt en mulighed for at tilføje et billede. Brugeren udfylder formularen med spot informationerne og trykker på knappen markeret "Add" for at submite den. App'en sender spot-informationerne til databasen.



Sekvensdiagram over indstillinger:

Brugeren starter App'en, ved at trykke på iconet for App'en, på sin telefon. App'en viser kort interfacen, som er et kort over brugerens nuværende position med en menu-bar i toppen. Brugeren vælger settings fra menuen, ved at trykke på knappen markeret med tandhjul i menu-baren. App'en viser en radius slider, som er en knap man kan hive i for at indikere hvor lang en radius man vil load spots fra. Brugeren vælger en radius, ved at hive knappen til den ønsket radius og derefter trykke på knappen markeret "Accept" for at submitte den. App'en gemmer den nye radius.

4 Systemdesign sammenfatning

Kapitlet resumerer jeres foreløbige system-design så kort og klart som muligt. Samtidig udpeger I de vigtigste udestående design- og implementationsopgaver.

Vores plan vedrørende udviklingen af appen har indtil nu, været at bygge appen op i java med android developer tools. Det gav os en del udfordring, da vi ikke kunne få den simpleste udgave til at køre på en simulator. De mange timer og forsøg har gjort at vi følte os nødsaget til at gøre projektet mindre omfattende.

Derfor har vi valgt at bygge applikationen op webbaseret i html, css, javascript og mySQL, hvilket er programmer og metoder vi har erfaring med i forvejen. Hele implementeringen af Google maps foregår med javascript(altså som en hjemmeside), hvor vi så bagefter vil lave en android webview app, så vi kan tilføje applikationen på google play store. vi mangler at implementere alt andet end kortet på nuværende tidspunkt.

Vi arbejder med hjemmesidens server gennem en ftp via programmet filezilla. Serveren er bygget op med phpmyadmin og kører mySQL, dette bruger vi til at hente spots udfra databasen og derefter indsætte på kortet. Altså vi henter alt data udfra databasen som vi skal bruge til kortet. Til diverse funktioner vi har køres der forskellige queries i mySQL til at få de resultater ud som vi gerne vil have. Disse vil typisk være skrevet i PHP eller javascript.

5 Program- og systemtest

Dokumenter jeres foreløbige test af IT-løsningen. I kapitlet sammenfattes hovedresultaterne af jeres test-aktiviteter; mens test plan, test case specification, test incident report og test report summary placeres som bilag.

Test af funktionaliteter gøres løbende internt i gruppen og dermed sikres det at funktionaliteterne gør det, de rent faktisk skal. Men derudover skal vi sikre ting som usability og performance, dette gøres f.eks ved hjælp af tænke højt øvelser.

Thinking out Loud

Opgaver:

1. Åben appen og noter dig din egen lokation
2. Find det spot nærmest dig
3. Find information om spottet i fælledparken
4. Find en liste over alle spots
5. Find en butik
6. Tilføj et spot der hedder abc, med adressen xxx uden et billede
7. Ændrer afstanden i km for søgen af spots

6 Brugergrænseflade og interaktionsdesign

(a) Præsenter skærbilleder af de mest interessante dele af jeres brugergrænseflade



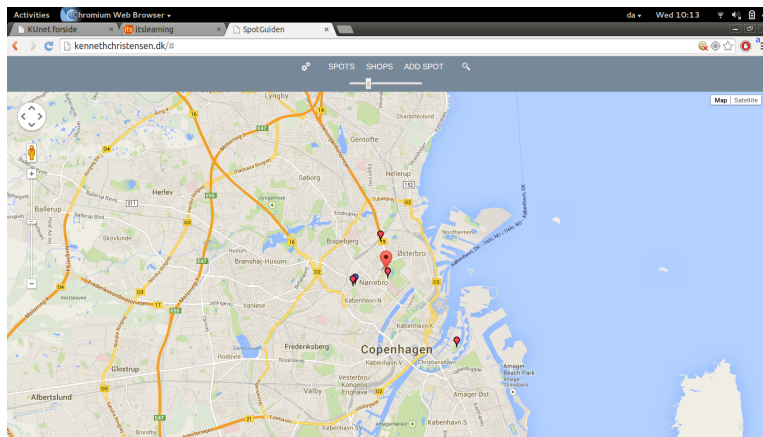
Billedet viser vores foreløbige hjemmeside der blot indeholder en menu uden funktioner, samt et kort med "spots" på den måde som vi vil præsentere dem.

Settings:

Når man trykker på settings kommer der en slider, som der kan bestemme hvor stor en radius siden skal load spots fra, vurderet fra brugerens nuværende position.

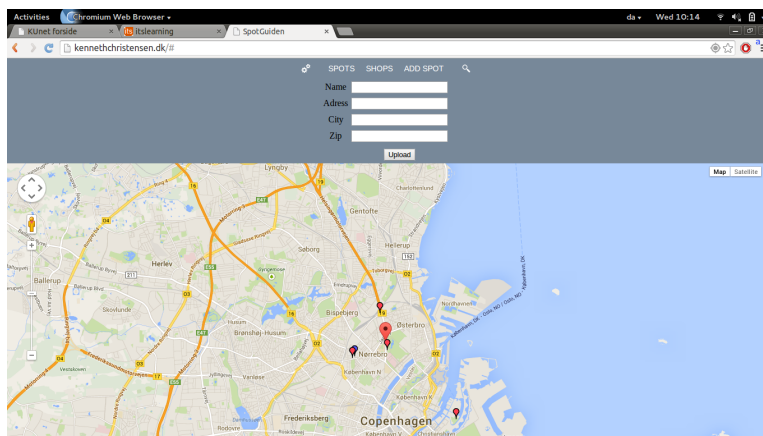
Spots: Når du trykker her kommer der en liste frem over alle de forskellige spots som man herefter trykke på dem. Dette vil centrere dem på kortet.

Shops:



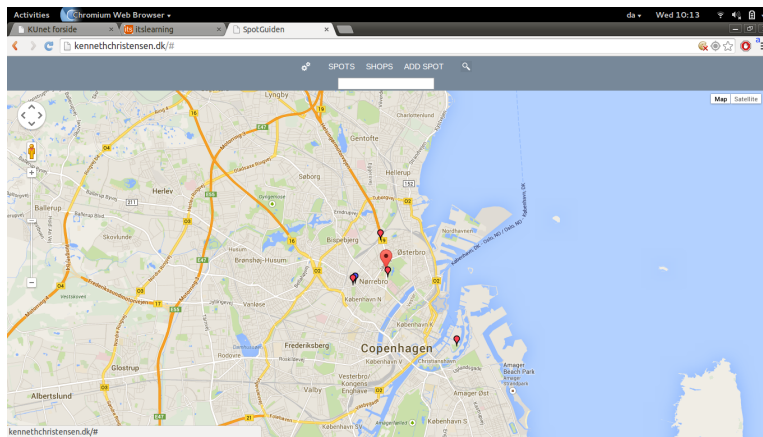
Centrerer kortet på BMXButikken.

Add spot:



Når der trykkes på dette kommer der en formular frem som man skal udfylde med: Name, Address, City, zip og billede.

Search:



Kommer frem med et søgefelt hvor man kan søge på spots via navn/zipcode/by.

(b) Illustrer flowet/dynamikken i brugerinteraktionen mellem skærbillederne

Dette vil blive udfyldt i delrapport 3 eller 4 hvor vi er kommet længere med programmet.

(c) En audio-visuel præsentation af brugergrænsefladen af den seneste kørende prototype

Dette punkt vil vi udfylde til delrapport 4 da vores projekt gerne skulle være færdig eller have mere funktionalitet på dette tidspunkt

(d) Resultatet af seneste tænke-højt forsøg gennemført med een eller flere af jeres brugere

7 Versionstyring

Som bilag skal vedlægges jeres nuværende commit-log samt jeres programkode. Kommentér kort (ca 1/2 side) de vigtigste ændringer, der er sket i programkoden

Vi har nu fået opdateret vores hjemmeside så istedet for der står settings og search er der nu kommet to forskellige ikoner som alle brugere kender, forstørrelsesglas og tandhjul. Der er ikke kommet noget funktionalitet bag det men designet til disse to knapper er nu kommet på.

Vi har fået opdateret vores database med flere spots og gjort det muligt for vores hjemmeside at der sker noget når der bliver trykket på en marker. Det der sker nu er at der kommer et info vindue op, med en adresse og et billede af spottet.

Vores app til at kunne køre hjemmesiden på android er næsten færdig.

Det eneste problem vi har omkring dette er at den ikke er kompatibel med at få en brugers position men dette vil altid blive et problem når nu den skal køre kunne køre offline.

Vores midlertidige projekt kan ses på www.kennethchristensen.dk det kan godt det stemmer over med hvad vi har beskrevet i rapporten da der stadig arbejdes videre på det.

8 Projektsamarbejdet

Beskriv konkret og oplysende hvordan det går med samarbejdet med brugerne og med arbejdet internt i gruppen. Herunder skal bl.a. oplyses antallet af møder med brugerne under projektforløbet (fx på en tidslinje), mødeformen i gruppen, samt hvorledes jeres referat- og dokumentationsform fungerer. Hvorledes prioriterer og styrer I projektindsatsen, så I sikrer fremdrift på de felter, som er mest risikable/afgørende for et succesfuldt resultat? Herunder, beskriv og diskuter:

(a) Hvad går godt?

Det interne arbejde i gruppen fungerer godt. Alle medlemmer i gruppen deltager aktivt, engageret og møder til tiden på de aftalte møde dage. Selve arbejdet prøver vi at fordele imellem os alt efter interesse og det sikrer en god stabil arbejds indsats.

(b) Hvad går mindre godt?

Vi har haft uventede problemer hvad angår systemdesignet da vi har måttet ændre kurs og gå over i en webbaseret udgave. Det har betydet at vi har haft nogle arbejds spildtimer og vi er derfor ikke noget helt så langt som planlagt.

(c) Hvad vil I gøre for at effektivisere jeres udviklingsarbejde?

I det vi tog skridtet fra android app til webbaseret app, har vi taget de første skridt i effektiviseringen. Derudover regner vi med at kunne begynde at arbejde selvstændigt på enkelte dele af systemet når vi har fået den første prototype helt op at køre. Det skulle bevirke at det bliver nemmere for de enkelte individer i gruppen at få plads i deres skema, til at få arbejdet på projektet og dermed få arbejdet flere timer.

9 Review

Cardboard computers: Mockingit- up or hands-on the future.

Artiklen handler om forholdet mellem computere og 'mennesket' som bruger dem. Artiklen argumentere for at computere skal forstås gennem mennesket brug af dem. Forfatteren starter med et hurtigt eksempel om en journalistisk process og hvordan et mock-up af et arbejdsmiljø, hvor computere er involveret, kan hjælpe med at påvirke arbejds-processen positivt.

'Why mock it up?' og 'We did not make it up'

Artiklen forklarer hurtigt hvorfor at lave mock-ups er en gode ide. Det gør det nemt for den endelig bruger at forstå hvordan det endelig produkt kommer til at fungere, og gør det nemmere at involvere brugeren i design processen. Derudover er det billigt så man kan lave en masse forsøg for at finde det som fungere bedst. Det bliver også pointeret at mock-ups ikke er noget nyt, da vi typisk som børn bruger mock-ups i form af legetøjs-biler, dukker osv. Det er også brugt i industriel design til at skabe gode og arbejdsvenlige arbejdspladser.

'Language games' og 'hands-on Experiences and Ready-to-hand Use'

Artiklen gør, i dette kapitel, det klart at det er vigtigt at alle involveret i et mock-up taler samme 'sprog', forstået på den måde at alle er klar over hvad hver ting repræsenterer og hvordan interaktion med objekter fungere. Forfatterne kommer så med et eksempel fra deres egen erfaring, hvor de havde udarbejdet en meget omfattende og detaljeret beskrivelse af det pågældende system. Dette kunne brugerne ikke relatere til. Det blev løst ved bruge et mock-up, hvor brugerne kunne bruge systemet i deres naturlige miljø og derved relatere til det.

'Beyond the Cardboard Computer', 'Second Generation Mock-ups' og 'Simple Mock-ups: Advantages and Disadvantages'

Artiklen gennemgår i disse kapitler hvad man kan gøre for at gøre sine mock-ups mere detaljeret, så som at bruge video eller slideshows. Man skal være opmærksom på at de ting bruger skal være almen-kendte så de holder sig inde for 'sproget'. En grund til at bruge simple mock-ups kan være pris begrænsninger, da store investeringer tidligt i design-processen typisk ikke er en god ide.

'Computers in Mock-ups: Overcoming the Disadvantages' og 'Computers in Mock-ups: Losing the Advantages?'

Artiklen dækker herefter fordelene ved at bruge computere til sit mock-up. F.eks at det er nemt at gemme et layout af sit mock-up og derefter afprøve andre layout uden at miste noget af det oprindelige. Dette kræver dog at dem som involveret i mock-upet har kendskab til computere. Ved brug af computere i mock-upet bliver det også muligt at dække spørgsmål rettet mere mod programmet (interface) end ved brug af papkasser o.lign. En ulempe ved den forhøjede funktionalitet computere giver, er at brugerne muligvis tror at mock-upet er et stort set færdigt produkt, men da dette ikke er tilfældet kan der opstå forvirring og misforståelser.

Det vi i gruppen mener der er værd at tage med fra denne artikel er selve ideen om at simplificere ens system ned til et punkt, hvor det er muligt for den fremtidige bruger at kunne deltage. Det nytter ikke at snakke om f.eks kode hvor brugeren ikke kan bidrage med noget. Med et program vil dog være knapt så nyttigt at lave en hel arbejds-station (i pap), men derimod bruge f.eks et story-board til at gøre tingene forståelige for ikke programmører. Story-boardet vil dog være bedst at lave på computer for at gøre rettelse o.lign nemmere og mindske risikoen for at miste materiale.

No Silver Bullet

"No Silver Bullet - Essence and Accidents of Software Engineering" er en artikel omkring software udvikling, skrevet af Fred Brooks i 1986. Brooks starter ud med at tilsidesætte software udvikling med myten om en varulv. Den eneste måde at få bugt med en varulv er som bekendt ved at bruge en "silver bullet". Brooks mener imidlertid at "there is no single development, in either technology or management technique, which by itself promises even one order of magnitude improvement within a decade in productivity, in reliability, in simplicity.". Han fortsætter i samme dur med "we cannot expect ever to see two-fold gains every two years" in software udvikling, ligesom tilfældet har været indenfor hardware udvikling. Dermed konstaterer han at denne såkaldte "silver bullet" ikke eksisterer og ikke vil blive udviklet indenfor de næste 10 år.

I følge Brooks kan grunden til at denne "silver bullet" ikke findes, deles op i to problemer henholdsvis "Essence complexity - Bliver skabt af det problem der skal løses og kan altså ikke fjernes" og "Accidents complexity - Der opstår på grund af fejl udvikleren laver og som dermed kan rettes". Men samtidig hævder han at software udviklerer på dette tidspunkt (1986) bruger det meste af sin tid på "Essence complexity", og mener altså at selvom "Accidental activities" elimineres vil det ikke resultere i en "magnitude improvement" (Tifoldig forbedring i udviklings hastigheden).

Selvom der bliver heftigt argumenteret for at der ikke findes en "silver bullet" mener Brooks alligevel at der er et par lyspunkter på vej, bl.a. i form af høj niveau programmerings sprog som Fortran, der kan sammenlignes lidt med nutidens C og Java.

Brooks proklamerer yderligere at der er forskel på gode system udviklere og fantastiske, idet at han mener software udvikling er en kreativ proces og at nogle udviklerer er grundlæggende bedre end andre, grundet deres genetiske arv. At udnytte dette og sørge for at skabe fantastiske udviklere fremfor middelmådige skulle ifølge Brooks kunne lede til en "magnitude improvement".

Artiklen set i forhold til object orienteret programmering

Brooks nævner blandt andet at han er stor fortalere for object orienteret programmering og ser store muligheder i udviklingen af dette. Han mener at OOP, sammen med et høj niveau programmerings sprog, kunne resultere i at mindske både "accidental complexity" og "essential complexity" og dermed være en mulig "silver bullet".

Vi (gruppen), mener imidlertid ikke at dette er tilfældet da kompleksiteten ikke kan simplificeres ved hjælp af disse værktøjer, men blot bedre kontrolleres. Vi ser derfor OOP som en teknologi, et koncept altså en måde at angribe et givent problem på. Konceptet tilbyder værktøjer der kan forbedre et design og derigennem gøre det nemmere at håndtere kompleksiteten, men udvikleren kan ligeså godt ende med et dårligt design der forværrer problemet og dermed kompleksiteten. Så kort sagt, hvis ikke konceptet bruges "rigtigt", risikerer du blot at yderligere besværliggøre opgaven. Vi konstaterer derfor at OOP ikke er en løsning til kompleksitet, men et (af flere) redskaber til at kontrollere det, hvis udvikleren evner dette.

10 Litteratur

- [1] Bruegge, Bernd and Dutoi, Allen H. *Object-Oriented Software Engineering Using UML, Patterns, and Java. Third Edition* 2010.
- [2] L. Mathiassen, A. Munk-Madsen, P. A. Nielsen and J. Stage. *Object-Oriented Analysis and Design kap 1* 2000
- [3] Sutherland, Jeff *Scrum Handbook* ?
- [4] Highsmith, Jim *Extreme Programming* 2000 Cutter Consortium
- [5] Brooks *No Silver Bullet* 1986
- [6] Kyng, Morten and Ehn, Pelle *Mocking-it-up* 1991
- [7] Molich, Rolf *User Testing, Discount User Testing* 2003
- [8] Frøkjær, Erik and Hornbæk, Kasper *Cooperative Usability Testing* 2005
- [9] Parnas, David Lorge and Clements, Paul C *A Rational Design Process...How and Why to Fake IT* 1986
- [10] ? *Naur Programming As Theory Building* 1985
- [11] Michael Christensen, Andy Crabtree, Christian Heide Damm, Klaus Marius Hansen, Ole Lehrmann Madsen, Pernille Marquardsen, Preben Mogenssen, Elmer Sandvad, Lennert Sloth, Michael Thomsen *The M.A.D. experience* 1998
- [12] Gould, John D. and Lewis, Clayton *Designing for Usability: Key Principles and What Designers Think* 1985
- [13] Bødker, Kensing and Simonsen *The must book Kap. 9* 2004
- [14] Klein, Laura *UX for Lean Startups* 2013
- [15] Sestoft, Peter *Systematic Software Testing* 2008

11 Bilag

Commitlog

```
* <73f7798> 2014-05-14 [mhejselbak] (HEAD, master) delrapport 3 tilføjet
* <8fff4f0> 2014-05-14 [mhejselbak] delrapport 3 tilføjet
* <a9c842d> 2014-05-08 [rasmus03] (origin/master, origin/HEAD) Genup-
load af delrapport
* <d58b7ea> 2014-05-08 [mhejselbak] Delrapport 2 genaf tilføjet
* <2572462> 2014-05-08 [rasmus03] Tilføjede en ny usecasemodel
* <04acdab> 2014-05-07 [rasmus03] Mindre opdateringer til delrapport 2 -
intet nyt
* <533b5cb> 2014-05-07 [mhejselbak] added more to part 5
* <c5d6de6> 2014-05-07 [runefranch] updateret delrapport
* <c4bdcc5> 2014-05-07 [rasmus03] Opdaterede delrapport 2 med nye se-
kvensdiagrammer
* <e4f926b> 2014-05-07 [rasmus03] Tilføjede gruppe reviews (igen)
* <a8c5bbd> 2014-05-07 [runefranch] Merge branch 'master' of https://github.com/mhejselbak/PK
| \
* | <95229b4> 2014-05-07 [runefranch] Sekvens diagram
| * <d3ad6bd> 2014-05-07 [rasmus03] Flyttede log.log til den rette mappe
| /
* <a0eb8ab> 2014-05-07 [mhejselbak] Deleted style.css and edited delrap-
port2
* <3db9ac0> 2014-05-07 [rasmus03] Opdaterede tekst i delrapport 2
* <b4614f0> 2014-04-28 [mhejselbak] CSS til hjemmeside
* <d0c4133> 2014-04-28 [rasmus03] Tilføjede første udlæg til den webbase-
rede android application
* <ba05d40> 2014-04-28 [rasmus03] Tilføjede vores review af "Mood"
* <1da40ab> 2014-04-24 [rasmus03] Updaterede filer til delrapport 2
* <89ae7a4> 2014-04-24 [rasmus03] Commitlog billede
* <6f57d2e> 2014-04-24 [rasmus03] Tilføjede vores udkast til hjemmesiden
* <2077057> 2014-04-24 [runefranch] Nyt sekvens diagram
* <a006ca4> 2014-04-24 [runefranch] sidste sekvens diagram
* <f2ad5a2> 2014-04-23 [rasmus03] Updated files and added screenshots
* <c201cd9> 2014-04-23 [runefranch] sekvens diagram
* <ee33ad2> 2014-04-23 [rasmus03] Opdaterede delrapport2 med 2/2 review
og use-cases.
```

- * <592de43> 2014-04-23 [runefranch] reviewartikel
- * <e77b576> 2014-04-23 [rasmus03] FACTOR model - Bruges i delrapport 2
- * <599276d> 2014-04-23 [rasmus03] Delrapport 2 - Består af punkt 1, 2 og 1/2 review
- * <b2271ec> 2014-04-22 [mhejselbak] Prototype
- * <6e1b71b> 2014-04-10 [runefranch] review update
- * <d81beae> 2014-04-09 [runefranch] Review
- * <19705bc> 2014-04-03 [mhejselbak] Initial commit

Referater

Referat fra sidste møde med kunde

Møde med Travis den 8/6

Vi mødtes med travis i bmxbutikken eller som deres butik på nørrebrogade hedder "pedal & Co" Her fremviste vi vores program til ham og snakkede om forbedringer og evt ekstra funktionalitet, vi diskutererede også hvilken slags spots der skulle op og om man evt. skulle holde nogle spots hemmelige for brugerene for at ikke opfordre små børn til at hoppe over hegn f.eks.

Han var rigtig tilfreds med resultatet af programmet indtil videre, og glædede sig til at det blev færdig, så han kunne få det ud til den danske bmx scene. Han havde dog også nogle ting, som han synes vi skulle tænke ekstra over. Han synes bla. ikke at hjemmesiden var helt intuitiv nok han kom med et eksempel med den lille gule mand og google street view, hvor at en erfaren bruger af google maps ved at man skal trække den gule mand ned på kortet, vil en bruger der ikke er vand til google maps ikke kende den funktionalitet. Derfor ville han gerne have en knap inde i infowindows med streetview, så en bruger af appen også trykke på en knap inde i infowindowet og se googles billeder af spottet.

Derudover snakkede vi også om at få brugere til at bruge hjemmesiden, hvor vi blev enige om at det var vigtigt at starte med en database med en del spots, simpelthen for at få brugerne til at bruge hjemmesiden. Derudover hvis der først er brugere på siden vil de forhåbentlig også selv uploade deres egne lokale spots og på den måde kan spotguiden udvide sig til at være national, da der kommer massere af bmx'ere og skatere fra Jylland til hovedstaden for at køre