

A decorative graphic on the right side of the page consists of three blue circles of varying sizes, each with a darker blue center and a lighter blue outer ring. Two thin blue lines intersect at the top right, forming a large 'V' shape that frames the circles.

Nettside, Webshop og Beregningsmodell

Hovedprosjekt våren 2009

[Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.]

Innholdsfortegnelse

Presentasjon.....	3
Mål.....	3
Resultatmål.....	3
Effekt mål.....	3
Kravspesifikasjonen.....	3
Systembeskrivelse.....	4
Use case.....	4
Rammekrav i systemet.....	5
Design.....	5
Valg av verktøy.....	5
Prosjektgjennomføring.....	7
Metode.....	7
Fremdriftsplan.....	8
Aktivitetskart.....	8
Risikoanalyse.....	8
Use Case.....	8
Prototyping.....	9
Designprototype.....	12
Analyse.....	13
Dataflytdiagram.....	13
Hva er dataflytdiagram:.....	14
Hvorfor dataflytdiagram:.....	14
Design.....	15
Historie.....	15
Formater.....	16
Fordeler.....	16
Ulemper.....	17
Implementasjon.....	17
Ordbok for prosjektet.....	17
Vedlegg.....	18

Presentasjon

Hovedprosjektet gjennomføres ved Høgskolen i Oslo, avdelingen for ingeniørutdanning. Målet med oppgaven er å utvikle en online webshop for bestilling av postkasser. Dette vil lette arbeidsmengden til de ansatte i postkasseavdelingen betraktelig, siden de da slipper å snakke fysisk med hver enkelt kunde. I tillegg skal vi lage en beregningsmodell som gjør det mulig for kunder og arkitekter å beregne mål for plassering av postkasser i eksisterende bygg og nybygg. Disse systemene må stille store krav til brukervennlighet da brukerne ofte kan være uerfarne databrukere.

Mål

Resultatmål

Målet med vår prosjektoppgave er å lage et nettbasert system som gjør det mulig for kunder av Stansefabrikken products AS å bestille postkasser over internett, med en webshop. Siden skal også ha en beregningsmodell for plassering av postkasser i en oppgang. Denne beregningsmodellen skal kunne gi kunden mulighet til lage et oversiktlig bilde av hvordan målene til kassene er og hva de har plass til i sin egen oppgang.

Effektmål

Stansefabrikken Products ble i 2007 skilt ut som et eget forretningsområde som fokuserer på salg av produkter og løsninger innen et bredt spekter av elektroskap og postkasser. Selskapet er lokalisert på Fornebu og Fredrikstad er et datterselskap av Stafa Industrier AS.

Virksomheten er salg av postkassesystemer til boligblokker i Norge. Kassene er produsert i Litauen og Norge. Disse har solgt jevnt over mange år på bekjentskaper og navn. I de senere år er det blitt mer og mer viktig å være på nett med brosjyrer og andre hjelpemidler som bedriftens kunder kan benytte. Stansefabrikken AS har mange kunder med spesielle behov slik som arkitekter, byggefirmaer og forhandlere. De mangler hjelpemidler på nettet for oppsett av kasser i forhold til størrelsen på kassene og de lover som til en hver tid gjelder. Bedriften ønsker å utvikle enkle hjelpemidler for kundene sine i form av tabeller eller regneark som letter arbeidet for begge parter. Samtidig vil det bli behov for å utvikle en netthandel som også vil lette arbeidet. Systemet vil kunne lette arbeidsmengden på selgere av postkasser fordi kunder kan henvises til beregningsmodellen når det gjelder plassering av postkasser. Utligere har de til dags dato ingen form for bestilling over internett, alt salg foregår over telefon og e-mail. Dermed vil det å ha en nettbutikk forbedre stansefabrikkens nettsider i forhold til salg betraktelig. Effektmålet er å øke salget ved å gjøre produktene tilgjengelig for kundene via internett

Kravspesifikasjonen

Kravspesifikasjon er et formelt dokument, som brukes som en juridisk bindende kontrakt mellom kunde og utvikler om hva som skal gjøre i prosjektet. Det er viktig og utarbeide en god kravspesifikasjon slik at kunden og utvikler har satt seg klare mål for prosjektet.

Denne kravspesifikasjonen ble utarbeidet av gruppen sammen med oppdragsgiver, hvor oppdragsgiver måtte godkjenne kravspesifikasjonen før videre arbeid ble fastslått.

Kravspesifikasjonen er beregnet for de medvirkende i prosjektet, altså oppdragsgiver,

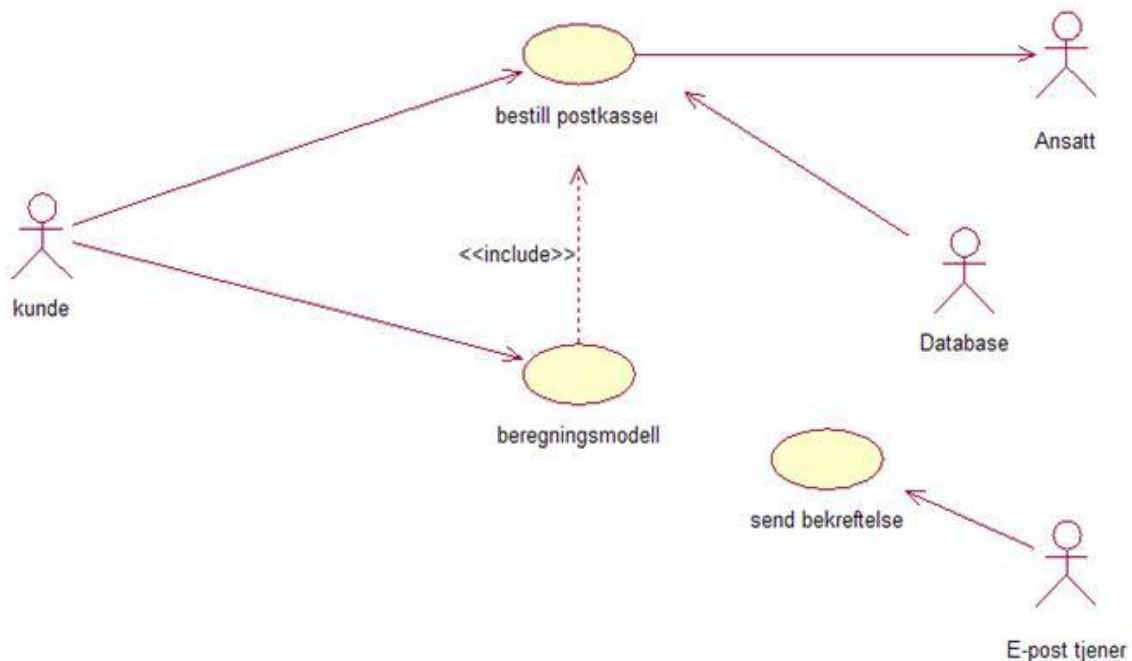
gruppemedlemmene og veileder. Den er også beregnet for en sensor som skal evaluere og bedømme prosjektresultatet. Systemets funksjonalitet, spesifikasjoner og rammebetingelser er beskrevet i dette dokumentet, som er en instruks for hvordan systemets skal fungere.

Systembeskrivelse

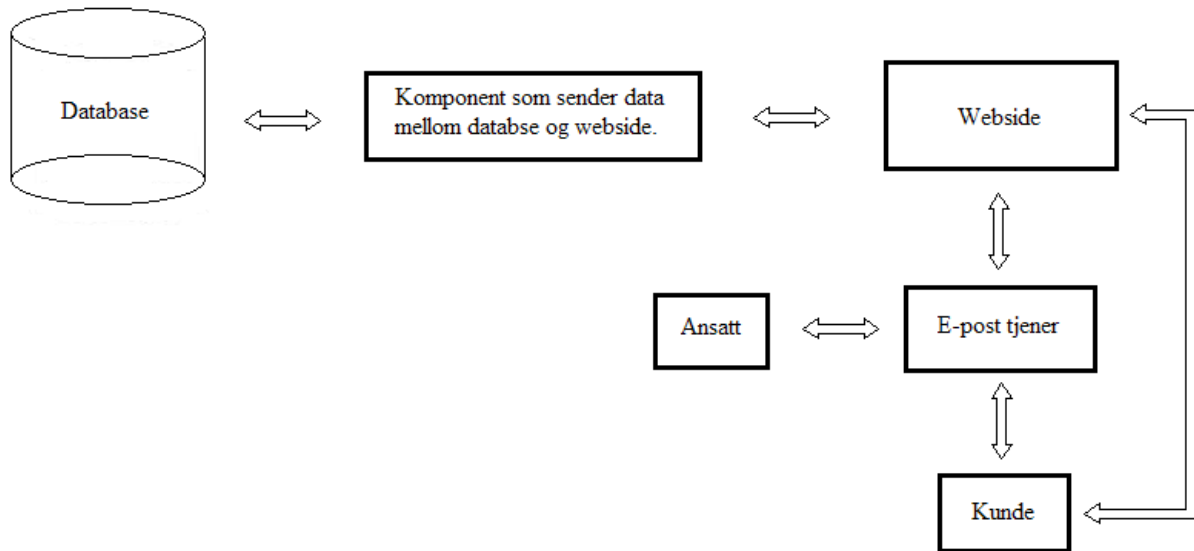
Systemet skal inneholde:

- Mulighet til å bestille postkasser på nett. Det skal være mulig å velge antall og type postkasser.
- En webshop hvor kunder kan bestille ønskelige postkasser uten å benytte beregningsmodellen.
- En beregningsmodell i flash som regner ut hvor mye plass som trengs til postkasser i henhold til standard for plassering av postkasser og mål oppgitt av kunde/arkitekt.
- Mulighet for kunde å sette sammen ønskelige postkasser ved hjelp av ”drag and drop” i flash.
- Database over alle typer postkasser med mål i millimeter.
- Mulighet for å skrive ut resultatet fra beregningsmodellen.
- Bekreftelse til kunde via e-post.
- Ansatte varsles om bestillinger på e-post.

Use case



Dette bildet viser et overordnet bilde av systemet for kunden.



Illustrasjon over strukturen i systemet.

Rammekrav i systemet

- Hoveddelene skal programmeres i PHP (objektorientert).
- Beregningsmodellen skal utarbeides i flash.
- Systemet skal kunne utvides ved at det kan brukes direkte på nettsidene til stansefabrikken. It avdelingen til stansefabrikken skal også kunne gjøre ytterligere forbedringer eller endringer i senere tid

Design

Valg av verktøy

Vår kunde har opplyst oss om at vi står fritt til å bruke de løsningene vi synes passer best til å utvikle dette systemet. Vi har valgt å utvikle systemet primært i PHP siden dette er et programmeringsspråk som alle på gruppen har vært borte i. Når det gjelder design på siden har vi valgt å bruke CSS, javascript, flash og html for å utvikle et brukervennlig grensesnitt. For redigering av bilder/grafiske enheter bruker vi GIMP. IBM Rational Rose brukes i planleggingsperioden for å få et konkret overblikk over systemets oppbygning og hendelsesforløp. Databasen er en mySQL database og designes i DB designer, hvor blant annet alle tabeller og deres sammenhengighet planlegges.

Her kommer en list med forklaringer på forkortelser og ord beskrevet i avsnittet over:

HTML ("HyperText Markup Language") er et programmeringsspråk for det å lage nettsider. I et slik dokument kan man legge til den informasjonen som skal vises på en nettside.

Vi bruker dette språket til og utvikle nettsiden.

PHP ("Hypertext Preprocessor") er et verktøy som brukes til å utvikle dynamiske nettsider. PHP er programkode som kjøres på serveren ikke på brukerens maskin. Dette vil si at når du som bruker besøker en nettside vil koden bli kjørt på serveren du kontakter, mens resultatet av denne kjørte koden blir sendt til deg. PHP gjør det mulig og koble nettsiden din opp mot database og kan motta input fra bruker og gjøre operasjoner på dataen.

Vi har valgt å bruke PHP i vårt prosjekt fordi det er et programmeringsspråk som vi har vært borti før og vi vet om dens muligheter innen nettside bygging. Likevel trenger vi også og lære mer om sikkerhet i php, siden vårt prosjekt krever noe mer sikkerhet en vi har laget tidligere.

Php : <http://no.wikipedia.org/wiki/PHP>

CSS ("Cascading Style Sheets") brukes til og forme design og utseende til en nettside. Nettsider er skrevet i html eller xml, ved bruk av CSS som suplemang til slike filer kan du ha mye av utforming, design og utseende valg i CSS.

Vi bruker CSS for nettopp dette formål, det og kunne forme sidene slik vi ønsker de skal være ut mot kunden.

CSS: http://no.wikipedia.org/wiki/Cascading_Style_Sheets

GIMP er et program for manipulering å behandle digital grafikk og fotografier.

Vi bruker GIMP til å produsere bilder for design av nettsiden, vi bruker også dette programmet til og lage bilder til beregningsmodellen i prosjektet, dette mest fordi bildene som brukes i flash animasjonen skal stemme overens med virkelige mål. Bildene må da lages presist i størrelser.

MySQL er det mest populære databasehåndteringssystemet i dag med åpen kildekode (open source). Grunnen til dette kan være at databasen kan kjøres på de fleste operativsystemer og kan kobles til ved hjelp de aller fleste programmeringsspråkene. Databasen selv er for øvrig skrevet i C og C++. MySQL er også brukt på flere av de mest trafikkerte nettsidene i verden for lagre data og brukerdata, blant annet Google, Youtube og Wikipedia. Dette sier oss at databasen er driftsikker og troverdig.

I vår oppgave benyttes MySQL for å ha et sted å lagre produktdata og hente ut login-info til administrasjonssidene på weben. Webshopen henter da ut info om de forskjellige produktene, inkludert bilder.

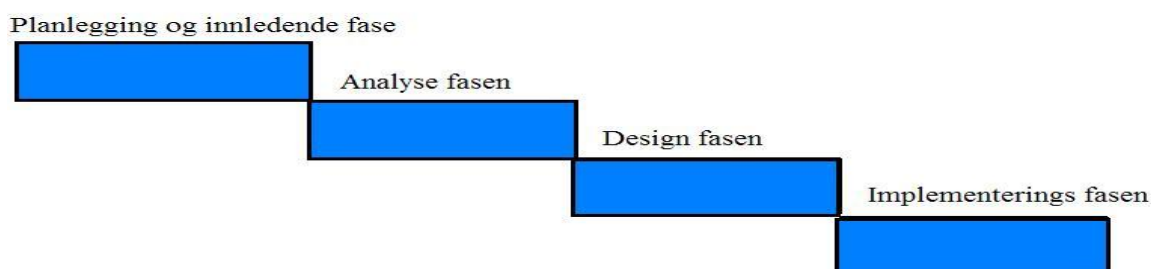
Mysql: <http://en.wikipedia.org/wiki/Mysql>

Prosjektgjennomføring

Metode

Fossefallsmetoden er en metode for system utvikling. Fossefallsmetoden går ut på at du deler opp hele prosjektet i faser. Man utfører disse fasene en etter en i rekkefølgen man har satt opp de forskjellige fasene. Når man bruker denne formen for systemutvikling må en fase gjøres ferdig før man begynner på en ny en. Vårt fossefallsdiagram:

Fossefallsmetoden



Ved bruk av en slik metode jobber man seg ned hver fase til man kommer til slutten. Vi har valgt å bruke fossefallsmetoden fordi denne er god å bruke på mindre prosjekter hvor man har klare krav og godt dokumentert planlegging. Hvis man jobber på større prosjekter der mange er involvert og kravene endres ofte er ikke fossefallsmetoden den beste utviklingsmetoden. Vi har delt opp vår utvikling under 5 faser.

- **Planlegging og innledende fase:**
I denne fasen setter vi mest vekt på å planlegge systemet som skal utvikles. Vi arbeider med å utvikle en kravspesifikasjon for systemet hvor systemet er mest mulig fastslått på forhånd. I denne perioden valgte vi å lage en papirprototype for å fastslå hovedtrekk ved design av siden.
- **Analyse fasen:**
I denne fasen går vi litt dypere i systemet og prøver og analysere på et litt lavere nivå enn i planleggingsfasen. Her benytter vi Er-diagram for planlegging av database. Vi ser på strukturen i systemet og lignende.
- **Design fasen:**
I denne delen av utviklingen av systemet lager vi design prototype og planlegger design av den endelige side mye dypere, jobber med kompetanse utvikling av nytt materiale for utvikling av det som skal lages.
- **Implementeringsfasen:**
Er den delen av metoden der vi faktisk produserer systemet. I denne fasen lager vi systemet ut i fra planlegging fra tidligere.

- **Testing:**
Siste fase er da vi tester det endelige systemet på brukere av systemet.

Fremdriftsplan

Fremdriftsplan er en oversiktlig plan over fremdriften i prosjektet. Fremdriftsplanen brukes for å sette tidsrammer for hele prosjektet. Dette er et overordnet dokument som hjelper oss og se når vesentlige deler av planlegging, analyse, utvikling og implementering skal være ferdig. Vi valgte å bruke dette styringsdokumentet for å få et overordnet blikk på hva som skulle gjøres og sette frister for disse oppgavene.

Fremdriftsplan (vedlegg 1).

Aktivitetskart

Vi valgte å lage en litt mer detaljert plan for arbeidet vårt under dette prosjektet ut over den allerede utviklede fremdriftsplanen. Dette dokumentet kalles aktivitetskart, der vi puttet inn alle punkter som skulle gjøres under veis. Vi fylte ut i denne planen når ting skulle være ferdig, hvem som hadde ansvaret for punktet, hvorfor punktet skulle utføres og hvor lang tid det egentlig tok og fullføre det. Dette dokumentet kunne da være en ganske enkel 2 ukers plan med oversiktlig blikk på hva som måtte gjøres. (vedlegg 2)

Risikoanalyse

Risiko analyse brukes til å kartlegge problemer som kan oppstå under et prosjekt eller en oppgave. Man bruker det til å kartlegge hvor stor fare det er for at et problem kan oppstå og konsekvensene av et slikt problem. Dette dokumentet har vi valgt å lage for å ha et oppslagsverk ved eventuelle problemer underveis. (Vedlegg 3)

Use Case

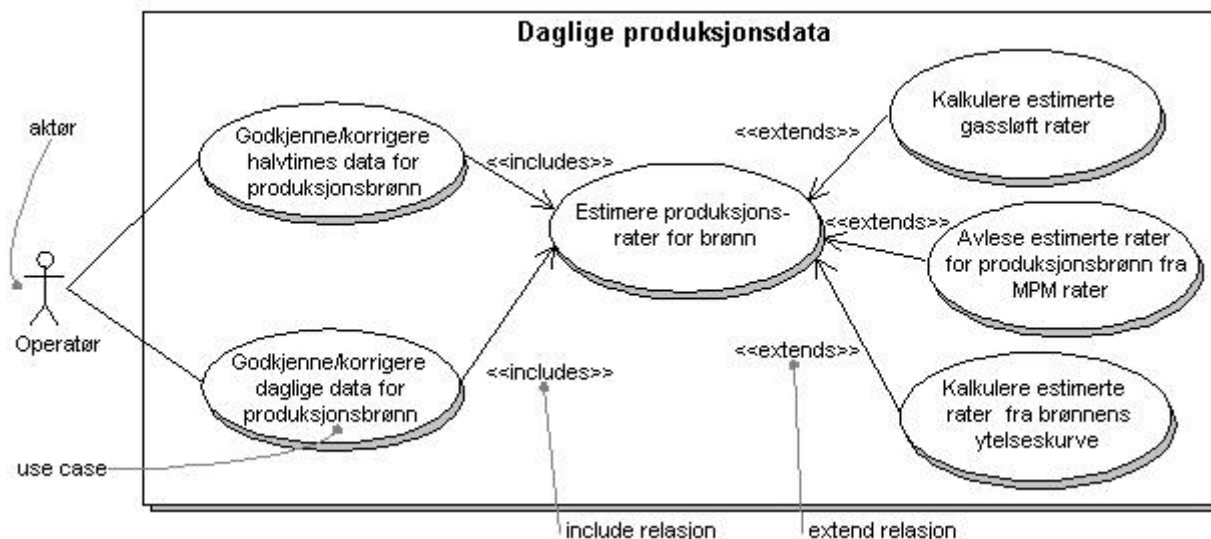
Use Case diagrammer brukes til å illustrere bruken og oppførselen av systemet. Use Case diagrammer forteller hva systemet skal gjøre, men ikke hvordan det skal gjøres.

Diagrammene brukes derfor til enten å modellere sammenhengen, oppbygningen av systemet, eller til å vise kravene til systemet.

Use Cases kan forbindes med følgende relasjoner:

- **Include:** En include relasjon mellom use case A og use case B, hvor pilen peker fra A til B, betyr at use case A arver oppførselen til use case B ved en spesifisert lokasjon. Dvs. use case A vil alltid inkludere (utføre) use case B.
- **Extend:** En extend relasjon fra use case A til use case B, hvor pilen peker fra A til B, betyr at use case B legger til oppførselen til use case A ved en lokasjon spesifisert indirekte i use case B. Dvs. use case B trenger ikke alltid utføre use case A. Merk at pilen peker mot use casen som blir utvidet.

Eksempel på et use case diagram:



Her kan vi se at selve systemet er inne i firkanten, mens aktøren er plassert utenfor. Dette er for å skille handlingene og hvem som utfører de. (Vedlegg3)

Kilde: http://home.online.no/~moestboe/uml_oversikt.htm

Prototyping

Papirprototype

Når man lager en papirprototype setter man seg ned med kravene til systemet og lager enkle strektegninger av systemet. Det som er greit og ha med i en papirprototype er litt funksjonalitet slik at man kan se litt hvordan systemet skal reagere på brukernes valg. På den måten kan man få en pekepinn på hvordan deler av et system er og teste den ut, eventuelt bruke det som planlegging for videre arbeid. Det kan også gjøre deg klar over eventuelle problemer man kan møte og hjelper og få start på design av brukergrensesnitt.

Vi valgte å lage en papirprototype av systemet for og planlegge enkel design og hvordan systemet skal se ut i grove trekk og få en enkel grov skisse over systemet som vi kan bruke til og planlegge systemet sammen med våre arbeidsgivere. Dette vil da sammen med kravspesifikasjonen vår gi oss et god utgangspunkt til og lage et bra system. Tankene og layout knyttet til papirprototypen er laget i samarbeid med arbeidsgiverne våre.

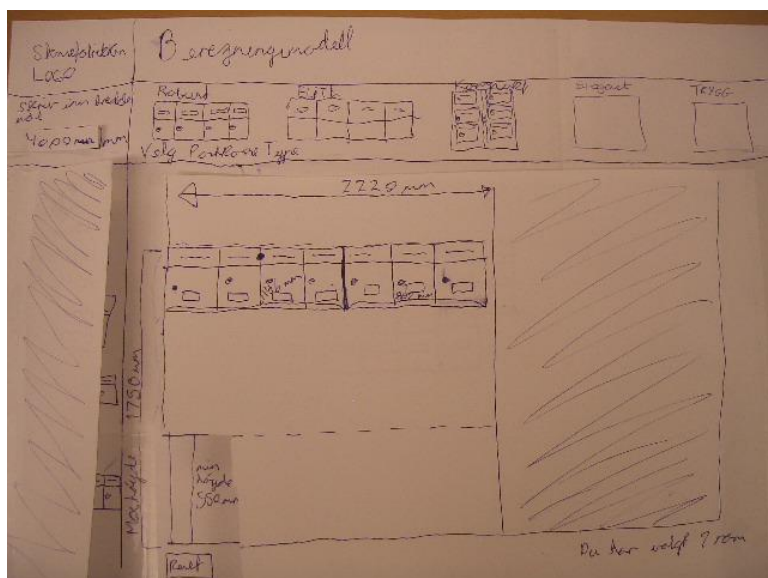
Siden systemet vårt er delt i 2 deler (beregningsmodell og web-shop) har vi valgt å lage disse hver for seg. Det vil komme en sammenslåing av disse i implementeringsfasen som da vil si at de vil henge sammen ved at de linkes til fra samme side.

Beregningsmodell

Vi startet med å tegne en enkel layout til en webside lik den stansefabrikken products AS har i dag, det neste var og lage selve beregningsmodellen, først fant vi ut av hvordan layout den skulle ha. Så valgte vi plassering av de mulighetene den skulle ha. Da vi var ved dette punktet måtte vi ha litt tanker om hvilket utviklingsverktøy som kunne klare å lage den funksjonaliteten vi ønsket. Vi valgte å se litt på nettet etter teknologi som gir støtte for det vi ønsket og lage. Vi kom da frem til at flash (beskrevet om i avsnitt "hva er flash?") var et

verktøy som ville egne seg til en slik beregningsmodell. Det neste vi gjorde var da å lage en enkel funksjonalitet i papirprototypen vår som viste hvordan den skal reagere i et gitt scenario.

Her er et bilde av papirprototypen, denne viser i grove trekk hvordan sluttproduktet av beregningsmodellen virker og ser ut i grove trekk:

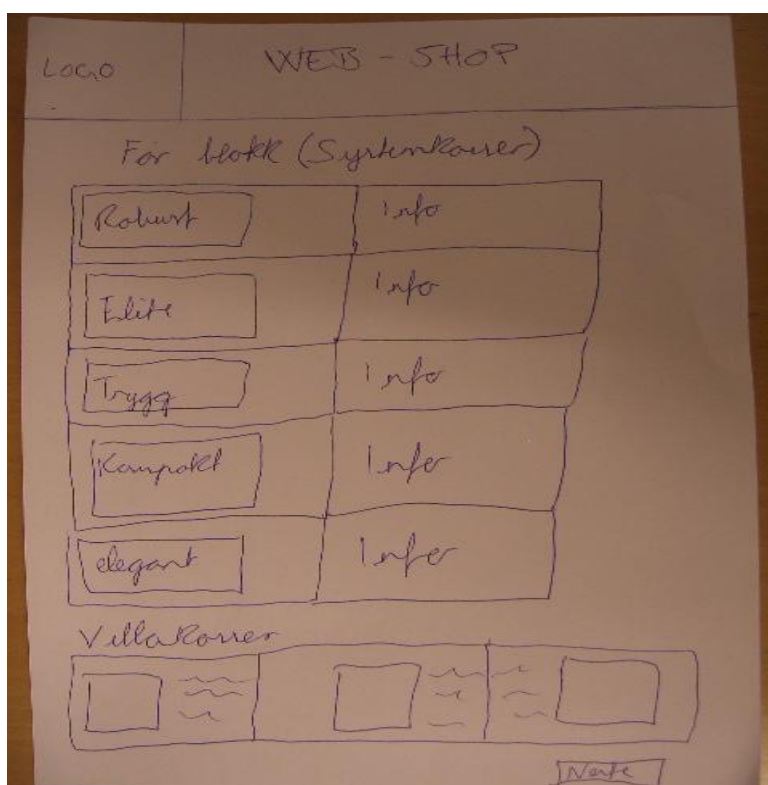


Tanker og Planlegging av beregningsmodellen i papirprototypen

Vi valgte å la produktene til stansefabrikken products AS (postkasser) stå på rekke øverst på nettsiden, siden dette er det stede kunden først ser å det er det mest relevante som må komme frem, når kunden da velger en type postkasse etter ønske, vil han få frem et nytt vindu med alle størrelser av den utvalgte postkassetypen. Her

valgte vi å legge til en funksjon som ber kunden skrive inn breddemål (ledig plass) i oppgangen. Når kunden fyller ut dette vil en tegning med mål komme opp til høyere for postkassene. Disse vil ha høydemål fastslått i følge den standard som postverket krever. Det systemet skal gi mulighet for nå er å dra postkasser over til det avgrensede området slik at du får en detaljert plantegning for kassene med mål. Hvis kunden velger å dra en kasse det ikke er plass til i bredde vil systemet forby dette.

WEB-shop



Ved utviklingen av papirprototypen for web-shop delen startet vi med enkel layout for siden som i beregningsmodellen, Siden dette må være likt slik at den eventuelle brukeren av systemet føler at han eller hun er enda på samme side.

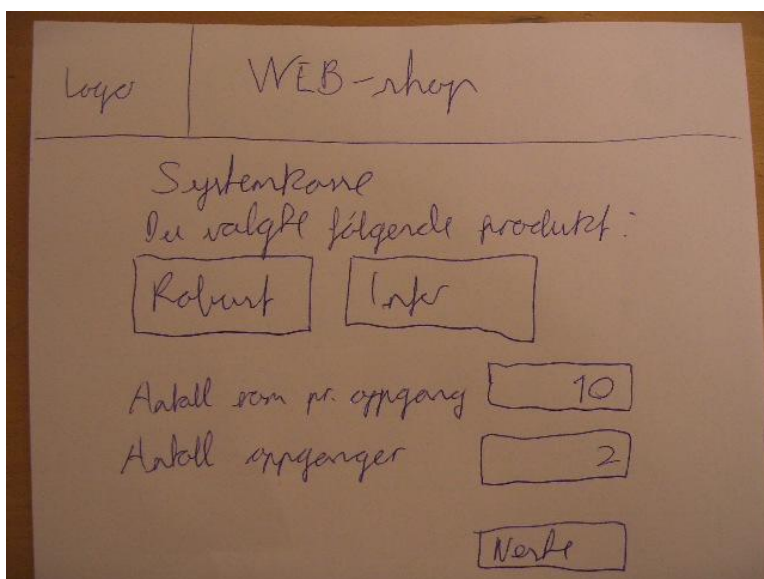
Det neste var og få med de tingene vi har beskrevet i kravspesifikasjonen. Først vil siden gi brukeren bilder av alle

produktene som vist på bilde ved siden av:

Her valgte vi å sette produktene inn i en tabell siden dette syntes å være den mest oversiktligte måten og plassere de på. Info om postkasser vil da få en god grafisk plassering i forhold til bilde av produktet. Vi valgte å dele opp postkassene i systemkasser og villakasser. Forskjellen på disse to er at systemkasser er kasser ofte relatert til hus med flere

postkasser i en oppgang, Villakasser er mer for et hus eller postkassestativ og trenger da mindre planlegging antall oppganger.

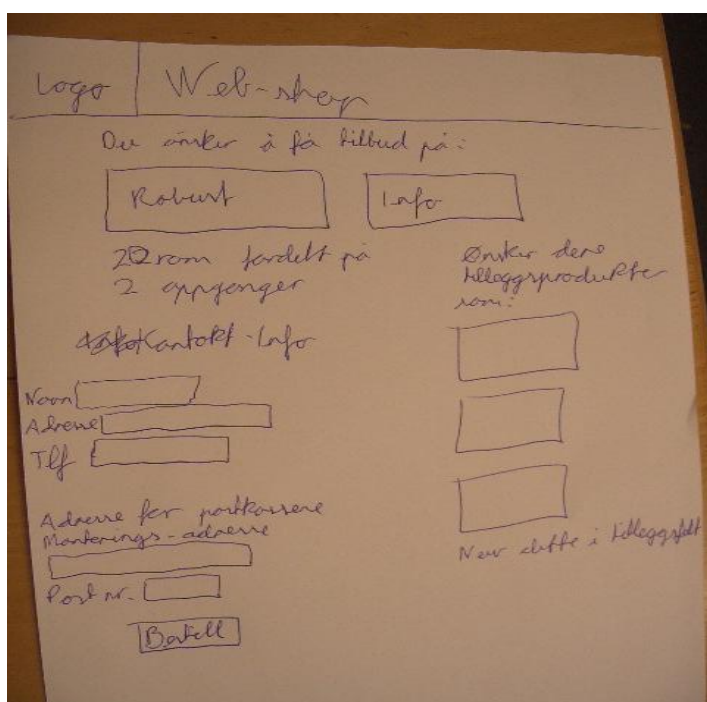
Når kunden har valgt en postkasse type vil han eller hun bli sendt til dette skjermbildet.



Her kan kunden lese litt om produktet de har valgt, de får nå mulighet til å velge hvor mange kasser de skal ha av produktet. Vi har valgt å ha flere sider for å gjøre det mer oversiktlig for kunden underveis. Kunder kan fort ha en tendens til og droppe siden hvis personen møter "a wall of text", med dette mener vi at hvis siden blir for overflyt til at kunden mister helt fokus og dette vil medføre at han eller hun kanskje dropper å gjennomføre

kjøpet.

Når kunden har valgt antall postkasser vil personen bli sendt til dette skjermbilde:



Her blir kunden nødt til å legge inn informasjon om seg selv slik at kundebehandler kan kontakte kunden om bestillingen. Vi har valgt å plassere dette på en oversiktlig måte til venstre på siden hvor det skal komme klart frem hva som behøves og fylles ut. Vi vil også forklare at informasjonen ikke vil bli brukt til annet enn å knytte bestillingen til kunden.

Når kunden har gjort dette vil systemet sende bestilling og en bekreftelse til kunden.

Etter utvikling av papirprototype:

Når papirprototypene var ferdige, valgte vi og ta en enkel brukertest på systemet vårt på våre arbeidsgivere for å høre med de om dette var slik de tenkte i grove trekk at systemet skulle virke og se ut. De ga tilbakemelding om at papirprototypen av enkel å manøvrere og virket som er bra layout for systemet som ønskes.

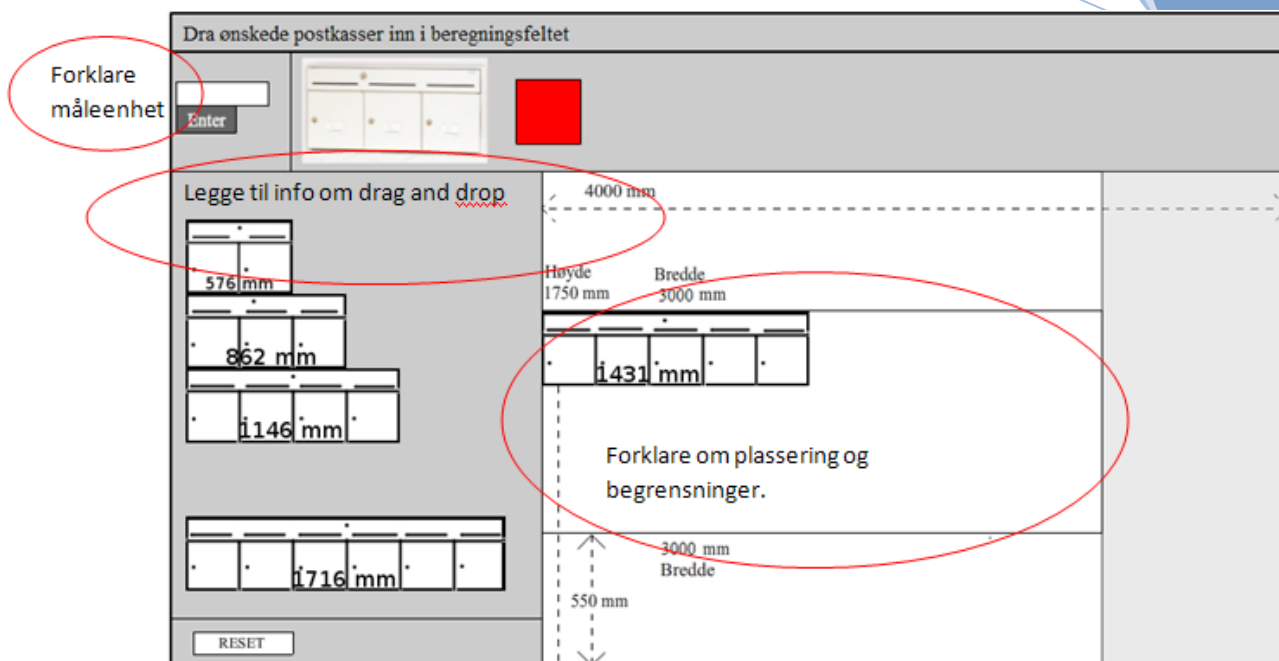
Brukertesting

Designprototype

Designprototypen er en prøve av en liten del av systemet der man legger stor vekt på hvordan systemet skal se ut for kunden. Funksjonalitet er ikke det som er mest vektlagt i en designprototype, derimot hvordan designet på brukergrensesnittet er. Vi valgte å lage en design prototype av den delen av systemet vi følte trengte dette. Vi laget en design prototype av beregningsmodellen som vi testet på noen med studenter. Vi testet beregningsmodellen på fire medstudenter fordi de vil ha eventuelt de samme forkunnskapene som en eventuell kunde, eller arkitekt som skal bruke systemet. Dette gjorde vi for å se hvordan folk reagerte på designet vi har valgt og om systemet og brukervennligheten til systemet svarte til våre forventninger.

Brukertesting av designprototypen

Under brukertesting fant vi fort ut av hva som kunne forbedres. Alle våre testsubjekter nevnte at forklaringsfeltet som var plassert øverst var vanskelig å få øye på. Dermed var testsubjektene ikke klar over hva de skulle gjøre tidlig i prosessen. For å motarbeide dette problemet har vi funnet ut at en mer markant skrifttype / størrelse gjør det enklere og se skriften. I tillegg til dette vil vi sette en forklaring på hva kunden skal skrive inn i feltet. En av våre testpersoner mente at hvis vi plasserte forklaringene nærmere eller i samme ramme som der man skal utføre en aktuell oppgave så ville det vært en forbedring. Det viste seg også at brukerne ikke forsto at det var drag and drop funksjon på postkassene i hovedfeltet så her trengs det også å utarbeides et forklarende felt.

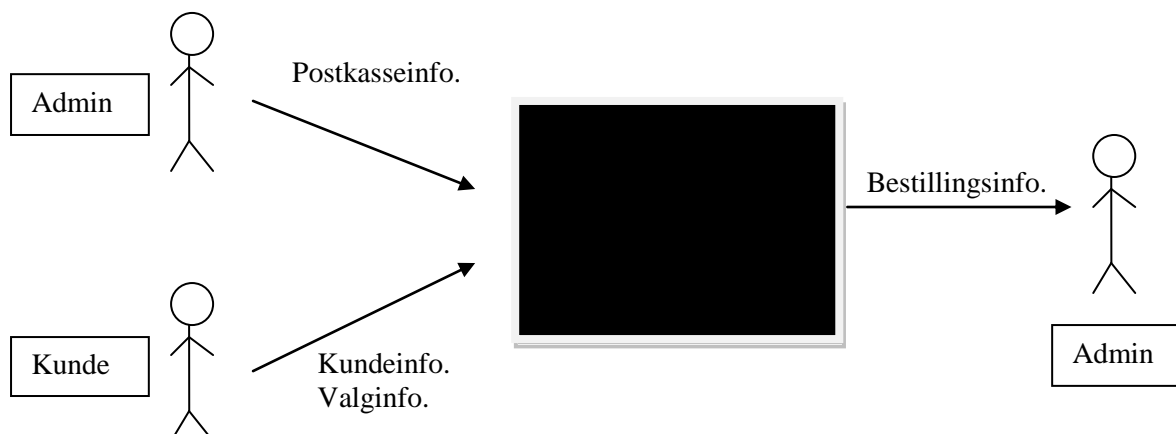


Videre utover de nevnte punktene over syntes brukere at systemet var veldig fint og greit og bruke. Kun tilbakemelding om dårlig forklaringer.

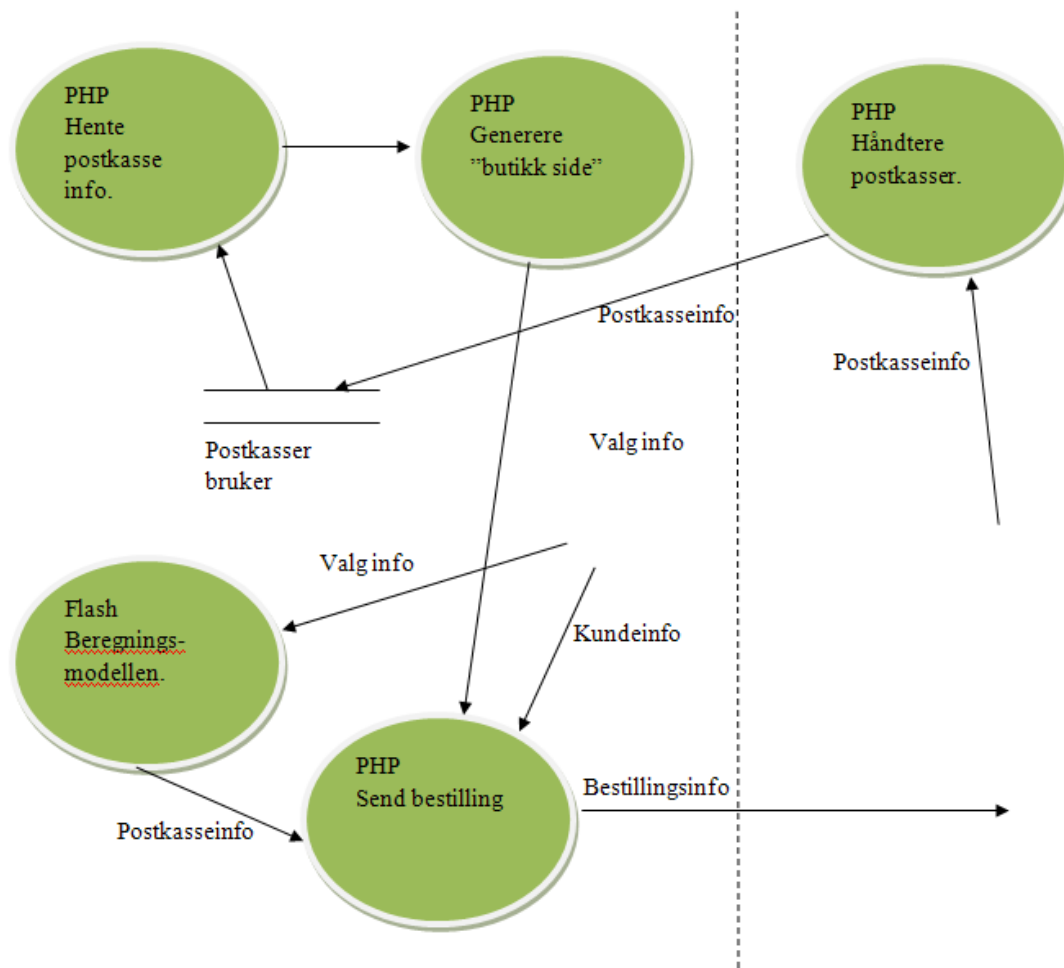
Analyse

er –diagram

Dataflytdiagram



Et kontekstdiagram gir en oversikt over dataflyten i et informasjonssystem. I kontekstdiagrammet representerer den svarte boksen systemet. Pilene representerer dataflyten mellom aktørene og systemet. Diagrammet viser hvilken informasjon som går fra aktørene til systemet og hvilken informasjon aktørene mottar fra systemet. Aktøren "Kunde" sender "Kundeinfo" som er kundens personalia og "Valginfo" som er kundens valg av postkasser, til systemet. Aktøren "Admin" sender data om postkasser i "Postkasseinfo" til systemet. "Admin" mottar også bestillingsinformasjon fra kunder gjennom "Bestillingsinfo" fra systemet.



Hva er dataflytdiagram:

Dataflytdiagrammet viser sammenhengen mellom forskjellige prosesser, datalagring, og ekstern entitet. Dens oppgave er å gi et oversiktlig blikk på hvordan prosessene er bygget opp og sammenhengen mellom dem.

- En sirkel er en prosess. En prosess beskriver en handling eller en transformering av data.
- En pil viser i hvilken retning dataflyten går. Dette vil vise at dataen går fra eller til en prosess.
- To like streker rett over hverandre er et datalager, her er data lagret og pilen fra et slikt lager viser om data hentes ut eller blir puttet inn i datalageret.
- En ekstern entitet er personen eller aktøren som er utenfor systemet og er den som starter de forskjellige prosessene.

Hvorfor dataflytdiagram:

Vi har laget et dataflytdiagram til vår oppgave for å se hvordan systemets dataflyt henger sammen. Dette vil da gi oss et oversiktlig bilde av hvordan datastrømmen er og hvilke prosesser som er i systemet. Vi kan da se sammenhengen mellom delene i systemet.

Design

Adobe Flash



Historie

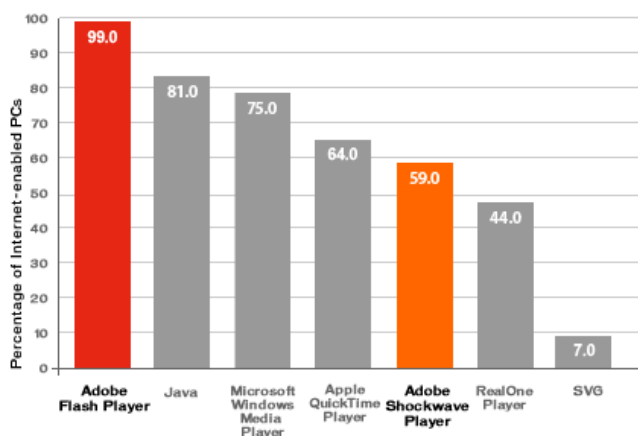
Flash er en av de hotteste innholds-teknologiene på weben i dag. Programmet så dagens lys i juli 1996 under navnet FutureSplash Animator. På samme tid var Macromedia også i gang med å lage sitt eget system under navnet Shockwave. Ressursene til Macromedia og produsenten av FutureSplash Animator ble slått sammen til ett program — Flash 1. i desember 1996. Siden det har programmet utviklet seg til å bli et meget kraftig verktøy for å lage webanimasjoner.

Flash brukes i dag til å lage alle mulige typer av webinnhold som: spill, tegnefilmer, logoer og annonser, men har med litt scripting også mulighetene for å kunne være en "stand alone" webbløsning hvor alt innholdet for en side er presentert i Flash istedenfor HTML. Flash kan også knyttes opp mot databaser og presentere XML data.

Worldwide Ubiquity of Adobe Flash Player by Version - December 2008

	Flash Player 7	Flash Player 8	Flash Player 9	Flash Player 10
Mature Markets¹	99.1%	99.0%	98.6%	55.9%
US/Canada	99.1%	99.1%	98.9%	54.5%
Europe²	99.1%	98.9%	98.2%	56.5%
Japan	99.0%	98.8%	98.3%	59.3%
Emerging Markets³	98.7%	98.6%	98.1%	55.9%

Oversikt over hvilke versjoner av flash som benyttes i dagens marked (per desember 2008)



Flash er verdens mest utbredte software plattform (basert på markedet i USA, Canada, Storbritannia, Frankrike og Tyskland).

Formater

Når du lager en Flash film bruker du kanskje både bilder, lyd, tekst og animasjon som du pakker sammen i en fil som presenteres på weben.

Flashprogrammet som du lager dette i kan lagre disse filene i to forskjellige formater:

- **fla**
"Work in progress" filen som brukes av Flash for å lagre originalfilen. Denne kan være relativt stor i størrelse (kb).
- **swf** (uttales *swiff*)
Den ferdige filen som eksporteres fra Flash for bruk på weben. Dette er en komprimert fil som tar mye mindre plass enn .fla filen.

Når en bruker besøker siden din blir *.swf filen* lastet ned på maskinen hans/hennes. Ved hjelp av en plug-in, eller en programutvidelse for browseren, vil brukeren kunne se din flashanimasjon. For de brukerne som har plug-in vil sidene se helt like ut, uavhengig av maskin og browser. Flash har også sitt eget programmeringsspråk, "ActionScript". Foreløpig siste versjon av AS er 3.0, som kom med Flash Player 9.

Noen av de mest kjente programmene for å utvikle flash er Adobe Flash, Adobe Flex, Adobe Air, Swift 3d og Swish.

Fordeler

Flash er bygget på vektorer. All grafikk du lager/tegner i Flash er basert på vektorer, i motsetning til bilder som er bygget opp med pixler. Fordelen med vektor-grafikk er at den er matematisk basert og alltid vil være like sylskarp i kanter og overganger uansett hvor mye du forstørker grafikken, i motsetning til bilder (gif/jpeg) som vil bli pixelert eller hakkete i kanten når du forstørker.

Det at den er matematisk basert betyr også at den fysiske størrelsen på grafikken ikke har noe å si for filstørrelse. Du kan lage en firkant fylt med blå farge så stor som et frimerke, så lage en kopi som du forstørker opp så mye du vil, og lagre begge filene. De er og vil alltid være like store.

At Flash er vektorbasert, betyr at selve grafikkfilene vi lager i programmet er relativt kompakte og raskt nedlastbare. Det som virkelig kan gjøre prosjektet ditt stort er når vi begynner å animere innholdet. Men ved å planlegge Flash-prosjektet grundig på forhånd, kan du lage raskt nedlastbare eller streamede del-filer.

Det at grafikken i Flash kan animeres er selvfølgelig en stor fordel.

Ulemper

En av de største ulempene er også nettopp det at Flash er gøy å jobbe med, og da har ting litt lett for å ta av. Nøyer man seg med statiske små Flashbildefiler vil ikke dette være noe problem, men Flash er et animasjonsprogram, og det er i animasjonsprosessen at filene virkelig legger på seg og blir tungt nedlastbare.

En annen ulempe er at flashprogrammene kan bli for avanserte og dermed virke mot sin hensikt. Designet går på bekostning av ytelsen.

Kilder: <http://www.2tp.no/webdesign/flash/index.htm>
http://www.adobe.com/products/player_census/flashplayer/

Implementasjon

Ordbok for prosjektet

PHP	PHP: "Hypertext Preprocessor". Programmeringsspråk. Brukes til å utvikle nettsider.
Flash	Betegnelse som refererer til programvaren Adobe Flash Player. Gir animasjon på nettsider.
Javascript	Skriptspråk. Tilfører dynamiske elementer til nettsider.
GIMP	"GNU Image Manipulation Program". Blir brukt til å behandle og lage illustrasjoner, digital grafikk og fotografier.
Webshop	Uttrykk for handel på nett.
Netthandel	Norsk uttrykk for handel på nett.
"Drag and Drop"	Betyr at brukeren kan interagere med eventuelle elementer ved å ta tak i ett element og plassere det på ønskelig sted.
Database	Samling av data.
mySQL	Versjon av SQL (Structured Query Language). Relasjonsdatabase for behandling av data.
CSS	"Cascading Style Sheets". Brukes til å definere utseende på filer skrevet i HTML eller XML.
HTML	"HyperText Markup Language". Markeringspråk for formatering av nettsider med informasjon som skal vises i en nettleser.

Grensesnitt

Betegnelse for hvordan en bruker kommuniserer med ett system.

Vedlegg