

HOVEDPROSJEKT:

TITTEL

DLAM

**DatabaseforLagringogAnalyseav
Mikrostrukturbilder**



Avdeling for **T**eknologi

FORFATTERE:

HansArneHalla,FrodeH.JahrogFinn -KjetilStemsrudhagen

Dato:

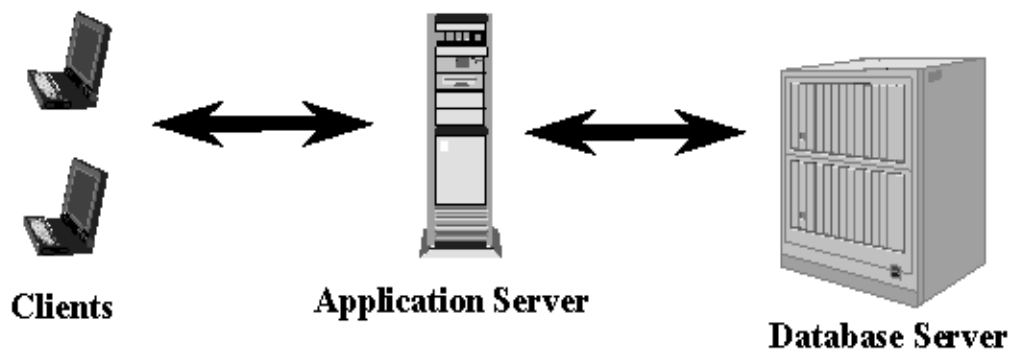
22.05.2001

SAMMENDRAG AV HOVEDP ROSJEKT

Tittel:	DLAM DatabaseforLagringogAnalyseavMikrostrukturbilder	Nr.: 9 Dato: 22.05.01
Deltaker(e):	Finn-KjetilStemsrudhagen(prosjektleder) FrodeJahr HansArneHalla	
Veileder(e):	SveinGautestad	
Oppdragsgiver:	RaufossASA	
Kontaktperson:	NoraDahle	
Stikkord (4stk)	Klient-server, database,java,mysql	
Antallsider:152	Antallbilag:9	Tilgjengelighet:Åpen
<p>Kortbeskrivelseavhovedprosjektet: MateriellteknologiskavdelingvedRaufossASAforetarbildefangstviascannere,digitalkameraer ogelektromikroskop.Dadetidligereikkeharhattetsentraltsystemtilåivaretadissebildene,har detværtmyejobbfordeansatteåholderedepådisse.Dennegruppensoppgavebestodiåutvikle enapplikasjonsomkunnekatalogisereogsystemeredissebildeneiettsentraltsyste mpådet lokaleintranett.Prosjektetbleenklient -serverløsningmedbrukavjava,jdbc,mysql,javascript oghtml.Herbledetlagdensentraldatabasesombletilkobletenjavaapplikasjonviajdbc. Enavgrunnpilarenei prosjektet,settfragruppens side,varatallsoftwareskulleværeavdenfritt distribuerbaretypen.</p>		

DLAM

DatabaseforLagringogAnalyseavMikrostrukturbilder



Skrevetav:



**Finn-Kjetil
Stemsrudhagen**



FrodeH.Jahr



HansArneHalla

Forord

Databasefor Lagring og Analyse av Mikrostrukturbilder

Dav verdens første datamaskin, ENIAC, i 1945 ble satt sammen og gikk på teetherelt lagerbygg, skulle en vel aldri tro at den nå er grekoloassen kunne ha noen innvirkning på samfunnet vårt. Hviskulle en vel med en gigant som var så vanskelig å bruke at en byggeplan for IKA møbler ble rene barne lekeni forhold? Itillegg kunne den også bare taimotinstrukser og utførelse uten noen form for lagringssmulighet.

Den utviklingsom har på gått siden da har vel neppe verdens sett makentil tidligere. Vi har idag et verdensomspennende nettverk med millioner likedataenheter som koblet sammen og som virker som en enorm informasjonsbærer. Itillegger dagens enheter så smått de kan plasseres opp på en skrivepult og med en kapasitet som er enormt mye større.

Hvordan skal en da kunne påberope seg etter tre år ved en høyskole å være godt skolert innen fagfeltet data. Enskulle tro at 5 års akkumulerte erfaringer ikke ville kunne overføres til løpet av tre år? Mens å lære det seg heller ikke gjør! Fagfeltet data er idag så svært at det må oppdeles i en rekke forskjellige spesialiseringer. En kan utdannes og tilf. eks systemutvikler eller programmerer, hvorav disse igjen vil bestå av en rekke undergrupper. Det vander for med stor spenningsatvitte gruppentokfatt på dette prosjektet. Vi hadde alle forskjellige retninger og interesser for vårt valg av spesialisering og ville hatt store problemer å kunne starte på en slik oppgave hver for oss.

Men ved å kombinere våre styrker fant vi fort ut at vi hadde det som trengtes for å få dette prosjektet til å gå i havn. En person kunne ha sin store interesse inne i systemutvikling og der med utfylle andre som manglet noen kunnskaper der. Mens denne igjen fikk hjelp av en av de andre hvis store interesse var programmering. På denne måten skred prosjektet sakte men sikkert framover. Det viste seg også at mange av de kunnskapene vi har tilegnet oss under årene ved HIGOs som vitidligere ikke har innsett nytte verdien av nå måttet i bruk. Oppgavene ga oss mer overordnet syn på alle de tidlige fagene, noe som gjorde at vi innså hvor de forskjellige fag kunne utfylle og brukes om hverandre. Derfor vil vi gjerne rette et takk til HIGOs men institusjon som har gitt oss en bred kompetanse innen datafagene. Her under må det rettes spesiell takk til Svein Gau testadsom har vært vår veileder under prosjektperioden.

Han har vært en god hjelp pogen på driver i dest under hvor vi kanskje har måttet stoppe opp for å vurdere hvilke valg vi hadde og i de perioder vi begynte å lure på om våre historiske valg hadde vært gjennomtenkte. Slike under hardt tross alt vært en del av, da oppdragsgiveren vår har vært av en såpass stor aktør som Raufoss ASA.

I regi av å være en stor og anerkjent bedrift både etter norske og internasjonale standarder så kan en skjønne at de har visse kvalitetsbegreper som må tilfreds stilles.

Samarbeidet med Raufoss har vært ut på klage, og vi vil takke kontaktpersonen vår, Nora Dahle, for samarbeidsomgående utfyllende kravspesifikasjon og senere applikasjon. Det vil her under også være å plass med et takk til IT ansvarlig på Raufoss, Hans Kristian Nordengen. Som med sin inngående kunnskaper innen IT har vært behjelpelig med spørsmål vedrørende dette tekniske rammeverket.

Gruppen kan nå etter prosjektet set til bak på spennende og lærerik periode, med både glede og sorg. Men med et produkt i likhet med det som nå står ferdig så luter det til at vi i ettertid mest føler glede!!



Gruppedlemmersunderskrifter;

Finn-KjetilStemsrudhagen
(Prosjektleder)

FrodeJahr

HansArneHalla

Sted: _____

Dato: _____

INNHALDSFORTEGNELSE

Kapittel1 Innledning

1.1	Utførelseavrapports.1
1.2	Oppgavedefinisjons.2
1.3	Softwarevalgogtidligereprosjekters.3
1.4	Målgruppeforrapports.3
1.5	Studentenesfagligebakgrunns.3
1.6	Arbeidsformerigruppens.4
1.7	Terminologibruks.4

Kapitel2 Kravspesifikasjon

	Innholdsfortegnelseforkravspesifikasjons.6
1	Introduksjon	
1.1	Bakgrunn s.9
1.2	Kortomkravtilsystemet s.9
1.3	Kortomsystemetsomgivelser s.10
1.4	Dokumentetsstrukturs.10
1.5	Definisjoner s.11
1.6	Referansers.13
2	Brukerbeskrivelser	
2.1	Omgivelsers.14
2.2	Systemetsbrukere s.14
2.3	Funksjon s.14
2.4	Operasjon s.16
2.5	Aspekteromkringlivssyklus s.17
2.6	Ytelse s.18
2.7	Begrensninger s.18
2.8	Antagelser	
3	Detaljertkravspesifikasjon	
3.1	Systemfunksjoner,funksjonskategorierogtverrrelasjoner s.19
3.1.1	Usecasediagram s.19
3.1.2.1	Usecasebeskrivelseogfunksjonslistefor“Lagrenyttbilde” s.20

DatabaseforLagringogAnalyseavMikrostrukturbilder

3.1.2.2	Usecasebeskrivelseogfunksjonslistefor”Søkeetterbi	ilde”	s.21
3.1.2.3	Usecasebeskrivelseogfunksjonslistefor”Navigasjon”		s.22
3.1.2.4	Usecasebeskrivelseogfunksjonslistefor”Visningavbilde”		s.23
3.1.2.5	Usecasebeskrivelseogfunksjonslistefor”Slettingavb	ilde”	s.24
3.1.2.6	Usecasebeskrivelseogfunksjonslistefor”Autentiseringogregistrering”			s.25
3.1.2.7	Usecasebeskrivelseogfunksjonslistefor”Brukeradministrasjon”		s.27
3.1.2.8	Usecasebeskrivelseogfunksjonslistefor”Loggføring”		s.28
3.1.3	Konseptuellemodellen		s.29
3.2	Dataspesifikasjon		s.30
3.2.1	Datainput		s.31
3.2.2	Dataoutput		s.31
3.2.3	Tverrfunksjonaledatadefinisjoner		s.31
3.3	Overordnedeoperasjonellesystemkrav		s.32
3.3.1	Normaloperasjon		s.32
3.3.1.1	Modusogkontroll		s.32
3.3.1.2	RangeringavUsecase		s.33
3.3.1.3	Yt else		s.33
3.3.1.4	Sikkerhet		s.34
3.3.1.5	Oppstartognedtagning		s.34
3.3.1.6	Tilgjengelighet		s.35
3.3.1.7	Innebygdetester		s.35
3.3.2	Operasjonifeilsituasjoner		s.35
3.3.2.1	Feilrapportering		s.35
3.3.2.2	Gjenervervelseetterfeil		s.35
3.3.2.3	Sikkerhet		s.35
3.3.2.4	Ytelse		s.36
3.4	Funksjonellekrav		s.37
3.4.1	FunksjonellekravtilDLAMsystem		s.37
3.4.1.1	Sekvensdiagramogkontrakterfor”Lagrenyttbilde”		s.41
3.4.1.2	Sekvensdiagramogkontrakterfor”Søkeetterbilde”		s.43
3.4.1.3	Sekvensdiagramogkontrakterfor”Navigasjon ”		s.45
3.4.1.4	Sekvensdiagramogkontrakterfor”Visningavbilde”		s.47
3.4.1.5	Sekvensdiagramogkontrakterfor”Slettingavbilde”		s.49
3.4.1.6	Sekvensdiagramogkontrakterfor”Autentiseringogregistrering”		s.51
3.4.1.7	Sekvensdiagramogkontrakterfor”Brukeradministrasjon”		s.53
3.4.1.8	Sekvensdiagramogkontrakterfor”Loggføring”		s.55
4	Begrensinger			
4.1	Softwaredesignbegrensninger		s.57
4.1.1	Softwarestandarderogspråk		s.57
4.1.2	Softwarepakker/verktøy/dataserogoperativsystemer		s.57
4.1.3	Toleranser,marginer,muligheter/tilfeller		s.58
4.2	Hardwaredesignbegrensninger		s.58
4.3	Brukerdesignbegrensninger		s.58

DatabaseforLagringogAnalyseavMikrostrukturbilder

5	Aspekteromkringlivssyklus	
5.1	Dokumentasjon s.59
5.2	Modulogintegrasj onstesting s.59
5.3	Konfigurasjonsogversjonsstyring s.59
5.4	Kravtilsupport,serviceogvedlikehold.....s.59
5.5	Kravtilutvidelser s.60
6	Aspekterominnstallasjon	
6.1	Hardwareinstallasjons.61
6.2	Overgang/omleggings.61
6.3	Opplærings.61
7	Akseptansekravs.62
8	Prosjektstyringinkludertkvalitetssikrings.62

Kapitel3Design

3.1	Introduksjons.63
3.2	Målogobjektivers.63
3.3	Softwareomgivelsers.63
3.4	Hovedrestriksjoner/begrensningers.63
3.5	Datadesigns.63
3.6	Beskrivelseavarkitekturogkomponenters.64
3.7	Brukergrensesnittfordesigns.64
3.8	Begrensningerogrestriksjoners.64

Kapitel4Impl ementering

4	Implementerings.65
----------	-----------------------	-----------

Kapitel5Testing

5	Testings.66
5.1	Planleggings.66
5.2	Teststrategiers.66
5.3	Gjennomførings.67
5.4	Oppdagedefeils.67

Kapitel6Diskusjonavresultater

6	Diskusjonavresultaters.69
----------	------------------------------	-----------

Kapitel7Konklusjon

7	Konklusjons.70
----------	-------------------	-----------

Kapittel8Litteraturliste

8	Litteraturlistes.75
----------	------------------------	-----------

Vedlegg

- A FORPROSJEKTRAPPORT**
- B STATUSMØTER**
- C MØTEREFERATER**
- D UTDRAGFRALOGGBOK**
- E DATABASESKJEMAFORDLAM -DATABASE**
- F KODE-LISTINGER**
- G GANTT-SKJEMAER**
- H PROSJEKTAVTALE**
- I PUBLISERINGSAVTALE**

KAPITTEL 1 Innledning

1.1 Utførelse av rapport

Idell av rapporten vil det bli en vnt litt generelt om og grundtoppgaven.

Det vil blant annet bli skrevet om rapportens sammensetning, oppgave definering, foremål, målgrupper, gruppe medlemmers kompetanse, arbeidsform og terminologi. Nårens såkalt erinnelse vil det være gruppenes komplette kravspesifikasjon for systemutviklingsforløpet. Del 2 omhandler de ønsket og krav som ble stilt til applikasjonen.

Forøvrig er denne igjen oppdelt i 8 deler, hvorav de viktigste er del 2, 3 og 4. Disse ga oss som utviklere en pekepinne om hvilke komponenter som burde utvikles og med hvilke funksjoner.

Som en fortsettelse på krav og ønsker fra kravspesifikasjonen kommer designinn del 3.

Innholdet i denne går på hvilke tekniske løsninger som ble valgt og hvorfor. Denne gir altså et konkret svar på spørsmålet som dannes i del 2.

Her kommer det innoversikt over de moduler og komponenter utviklet og begrunnelse over hvorfor akkurat disse ble valgt til denne løsningen. Som et naturlig ledd i kommunikasjonen prosessen vil det her bli illustrert endel for å markere poenger. For å fremtidig brukere og lesere denne rapporten skal kunne gjenskape det miljøets systemer bygget for, så vil det bli forklart nøyede utviklingsmiljø som har blitt brukt. Her vil det fremkomme de software systemer som er blitt brukt til utviklingen og de software systemer som systemet primært er bygget for.

Videre vil prinsippene og standardene som er valgt under kodingsfasen bli tatt fram og begrunnet. Dette for å vise de beste valgene og å kunne tilfredsstillende ønsket som kommer i del 2.

Som et naturlig element i utviklingen vil vi i del 5 vise og beskrive hvordan applikasjonen har blitt testet for feil. Hvilke resultater som har kommet og slutninger av disse.

Kravspesifikasjonens om utgjør del 2 og som er analysert krav før utviklingsforløpet, utgjør fundamentet for en slik oppgave. Da en nå har oppgående og funksjonell løsning vil vi i del 6 diskutere resultatene vi har fått opp mot de krav og ønsket som var fremsatt i spesifikasjonen. Denne vil inneholde en evaluering av resultatene vi har oppnådd.

Del 7 vil inneholde en konklusjon etter utviklingsforløpet vårt. Noe som vil inneholde slutninger og diskusjoner rundt resultatene.

Her vil vi også ta hensyn til fremtidig bruk av deløsninger vi har kommet frem til.

For å autentisere oppgaven kommer det i del 8 underskrifter fra gruppenes medlemmer.

Litteraturliste vil komme i del 9 og vil inneholde referanser til kilder og nettsteder som er blitt brukt.

Som en avslutning kommer kapittel 10 til å inneholde relevante vedlegg i form av kildekode, referater, loggbok, fremdriftsskjema osv.

DatabaseforLagringogAnalyseavMikrostrukturbilder

1.2Oppgavedefinisjon

MateriellteknologiskavdelingvedRaufossASAhargjennomenår rekkegjennomført analyseravforskjelligematerialsammensetninger,noedegjördendagidag.Som regelblirdissebildeneunntil digitaltviaenscanner,digitalkamera,eller elektromikroskop.Detharikkeeksistertetsystemellerprogramforensystematisk lagringavdissetidligere,ogdetteharførttilatdeansatteharlagrettilfeldigpå varierendePC`erellerlagringsmedium.

Noesomharførttilstoreproblemer når bildeneeventueltettertidskulletasfram igjen.

Derforbledetførjuli2000stiltspørsmåltilHIGomdettekunneværeetpotensielt hovedprosjektforeleverveddatalinjen.

Etresultatavdettevarattreelevvisteinteresseforprosjektetogmeldtesegpå.

UnderførsterundemedbrukereoginitiativtakerevedRaufossbleproblemmrådetog oppgavedefinertmeryaktig.Oppgavensartbestodiålageetsystemsomsulle ivaretaenslikansamlingogkatalogiseringavbilderviadetlokaleintranett.

Detbleklarheltfrastartenavdetteskulleværeenhoveddatabasesomalleklienter skullekunnebenyttesegav.Atssystemetnødvendigvisdamåtteblienklient-server løsningsvarheltåpenbart.Systemetvillemåtteblidesignetpåenslikmåteatallede individuellePC`enesomvartilkobletintranettetskullehatilgangtil densentrale databasen.

DelokalePC`eneskulleikkeutsettesunødvendigmedarbeidsbelastningogderfor skulledetbyggesenapplikasjonsomblegrensesnittetutmotdatabasen.Dennemåtte ogsåinneholdedetmesteavlogikkenisystemet.

Noenbegrensningersomblepålågtossfraoppdragsgiverssidevaratsystemetskulle kunnekjørespåforskjelligeoperativsystemerogforskjelligebrowsere.Detskulle ogsådesignesmedtankepåatdetskullekunnebrukesisamarbeidmedforskjellige databaser.

Samtidigbledetlagtrestriksjonerpåhvilkeprogrammeringsspråkogserversideskript somkunnetillatesåbruke.Dettefordidewillehaetsystemsomvarbygdpåspråk somdehaddekunnskaperom.

Istartenavprosjektetvardetnevntat en analyse-funksjonav bildene gjerneskulle implementeres,mendettegikkutdarepresentanterfraskoleninnsåattidsressursforbrukvillemåtte langtoverstigetdenne gruppenhaddetilrådighet.

Detsomblebestemtavfunksjonalitetvaratdetskulleværeenklient-server løsnings hvorbrukerneviasinerespektivebrowserekunnelasteinnehjemmeside.

Fradennesidenskulledekunneleggeinnbildersomvarlagrettidligere,iensentral databasesomfysiskvillemåttefinnesegpåenserverettellerannetstedpåområdet.

Dettevilledagjennomføresviaenapplikasjonsomlåpånokenserver.

Søkpåtidligere lagrede bilder og en visning av disse skulle ogsåværetilgjengeligeved hjelpavbrowseren.

Foråbegrensedenressursmengdesomåtteinvesteresiprojektetbestemtegrupper segforågåforengjenbruksbasertløsningidengraddettevarmulig.Detsyntesikke åværeløsgiskåprøveog”finneopphjuletpånytt”dadeteksistertemuligheterforå forbedredeløsningsomalleredefantes.Dettevilsiatgruppenvillemåtte benytte seg av eksisterendesoftwaremodulerunderkonstruksjonenavapplikasjonen.

Databasefor Lagring og Analyse av Mikrostrukturbilder

Et understøttende argument foretstik valget av hovedprosjektet ikke utgjør hele arbeidsmengden i vår semesteret, men 40% av det endelige pålagt fraskolen side.

1.3 Softwarevalg og tidlige tilsvarende prosjekter

På bakgrunn av gruppe medlemmers interessefelt, tidligere erfaring og personlige aversjoner mot mange av softwaregigantenes en velde i dagens samfunn, ble det bestemt at systemet skulle bygges via gratis programvareløsninger. Det vil si at vi ville holde oss til løsninger som eksisterer til bruk for alle og som er gratis. Derfor falt vårt valg på en kombinasjon av MySQL, Java, JSP, JDBC, HTML. Ingen av disse enhetene isamspill utgjør noen nyhet, men det å legge inn bilder som en binær fil i databasen MySQL kan anses som noe nytt. Etter mangt søk på internett og i manualer fant vi ut at dette hadde blitt lite praktisert på den databasen. Akkurat dette feltet vil det på om det er gjort noe tilsvarende til tidligere studentprosjekter. Mye av dette kan komme av at MySQL er en ganske enkel database i oppbygning. Den understøtter ikke mange av funksjoner som kommersielle databaser skilte med og kan på det vis virke "dårligere". Men dette anses som et problem, da mangt kan programmeres i applikasjonen, dog med fare for større bryder i dataens kaloverføring til andre databaser. Kombinasjonen av Java, JDBC og JSP er helt vanlig i dagens systemutviklingsmiljøer og det finnes sikkert mange prosjekter som er gjort med disse som verktøy tidligere.

1.4 Målgruppe for rapport

Denne rapporten blir primært skrevet for veileder, studenter og andre med teoretiske kunnskaper innen fagområdet. Dog vil det ikke bli brukt gjennom syret stil med hard teknisk terminologi og forståelige begreper foren uten forstående. Den terminologi som blir brukt og som kan misforståes vil bli forklart, om ikke helt inngående, i definisjonslisten.

Illustrasjoner og en lettere forklaring av begreper vil også følge underveis, der hvor vi føler dette er på sin plass. Så med litt hjelp fra annen litteratur så skulle dette også kunne være lektyre for personer som ikke driver med dette som fag til daglig.

1.5 Studentenes faglige bakgrunn

Alle studentene som deltar i dette prosjektet er tredjeårsdatastudenter ved HIG. Dette medfører at alle har hatt systemutvikling, objektorientert programmering, programmering mot web, databaseteorier og alle defagsomskullet til si at en slik oppgave kunne løses.

Systemutviklingsfaget har gitt oss en sikker støtte når det gjelder å kunne tre inn i en situasjon, analysere krav og ønske og formulere dem til programmeringsaspekter.

Den rekke med programmeringsfagsom har vært pensum har gitt oss nok kunnskaper til å kunne ta lede problemersomkravspesifiseringsforløp på legges oss.

Databaseteorien har både vært et fag og kommet inn som et naturlige element i mange andre fagkombinasjoner. Disse samlede har gitt fundamentet til å gjennomføre

Database for Lagring og Analyse av Mikrostruktur bilder

ett slikt løp, samtidig som en har lært å søke opp informasjon til de problemene som oppstår.

1.6 Arbeidsformer i gruppen

I starten av systemutviklingsfasen var det nær kontakt med Raufoss ASA for å få kartlagt behov og ønsker. Denne kontakten ble redusert da design og kodingsfase ble påbegynt.

Dissemøtene har stort sett bestått av muntlige utspørringer med hjelp av spørsmål som var definert før møtet tok til.

Gruppen har jevnlig gjennomprosjektperioden hatt statusmøter og møter med veileder.

Noe som har gitt oss impulser til hvilke retninger vi har kunnet gå.

Utenom dette har gruppen hatt møter når viktige avgjørelser har blitt tatt, slik at disse har blitt tatt i plenum. Ellers har arbeidsoppgaver blitt utdelt til hver enkelt

grupped medlem, slik at mesteparten av kodingsfasen har blitt utført på privatbasis. Noe som har ført til større behov for koordinering mellom de forskjellige individer.

Den støttelitteraturen som har blitt brukt under systemutviklingsfasen har stort sett bestått av lærebøker fra systemutvikling I og II. Programmeringsteknikk har blitt innhentet i informasjon fra lærebøker og annen litteratur. En vurderer ligkildet i den fasen var all informasjon som fantes på internett. Her fantes nøkkel til løsning av mange problemer som dukket opp underveis.

1.7 Terminologi i bruk

Blant de kapitlene som finnes i denne rapporten er det sannsynligvis

kravspesifikasjonen og designkapitelet som inneholder mest fagsjargonger. Disse kan foruten forstående virkninger være kryptiske, men for å underlette forståelsen fordesom er i målgruppen vår, så må tenke seg disse være med. Definisjonslisten inneholder forøvrig de vanskeligste ord og forkortelser som har blitt brukt.

Men ellers er det lite tilgjengelig informasjon om fagfeltet som har noen problemer med å forstå innholdet i denne rapporten.

Kapittel 2:

Kravspesifikasjon

for

DLAM

Innholdsfortegnelsekravspesifikasjon

1 Introduksjon

1.1	Bakgrunn	s.9
1.2	Kortomkravtilsystemet	s.9
1.3	Kortomsystemetsomgivelser	s.10
1.4	Dokumentetsstruktur	s.10
1.5	Definisjoner	s.11
1.6	Referanser	s.12

2 Brukerbeskrivelser

2.1	Omgivelser	s.13
2.2	Systemetsbrukere	s.13
2.3	Funksjon	s.13
2.4	Operasjon	s.15
2.5	Aspekteromkringlivssyklus	s.16
2.6	Ytelse	s.16
2.7	Begrensninger	s.16
2.8	Antagelser	s.17

3 Detaljertkravspesifikasjon

3.1	Systemfunksjoner,funksjonskategorierogtverrelasjoner.....	s.18
3.1.1	Usecasediagram s.18
3.1.2.1	Usecasebeskrivelseogfunksjonslistefor" Lagrenyttbilde" s.19
3.1.2.2	Usecasebeskrivelseogfunksjonslistefor" Søkeetterbilde" s.20
3.1.2.3	Usecasebeskrivelseogfunksjonslistefor" Navigasjon" s.21
3.1.2.4	Usecasebeskrivelseogfunksjonslistefor" Visningavbilde" s.22
3.1.2.5	Usecasebeskrivelseogfunksjonslistefor" Slettingavbilde" s.23
3.1.2.9	Usecasebeskrivelseogfunksjonslistefor" Autentiseringogregistrering"	s.24
3.1.2.10	Usecasebeskrivelseogfunksjonslistefor "Brukeradministrasjon".....	s.26
3.1.2.11	Usecasebeskrivelseogfunksjonslistefor" Loggføring" s.27
3.1.3	Konseptuellemodellen s.28
3.2	Dataspesifikasjon s.29
3.2.1	Datainput s.30
3.2.2	Dataoutput s.30
3.2.4	Tverrfunksjonaledatadefinisjoner s.30
3.3	Overordnedeoperasjonellesystemkrav.....	s.31

DatabaseforLagringogAnalyseavMikrostrukturbilder

3.3.1	Normaloperasjon	s.31
3.3.1.1	Modusogkontroll	s.31
3.3.1.2	RangeringavUsecase	s.32
3.3.1.3	Ytelse	s.32
3.3.1.4	Sikkerhet	s.33
3.3.1.5	Oppstartogn edtagning	s.33
3.3.1.6	Tilgjengelighet	s.34
3.3.1.7	Innebygdetester	s.34
3.3.2	Operasjonifeilsituasjoner	s.34
3.3.2.1	Feilrapportering	s.34
3.3.2.2	Gjenervervelseetterfeil	s.34
3.3.2.3	Sikkerhet	s.34
3.3.2.4	Ytelse	s.35
3.4	Funksjonellekrav		
3.4.1	Funksjonellekravti IDLAMsystem	s.36
3.4.1.9	Sekvensdiagramogkontrakterfor”Lagrenyttbilde”	s.40
3.4.1.10	Sekvensdiagramogkontrakterfor”Søkeetterbilde”	s.42
3.4.1.11	Sekvensdiagramogkontrakterfor”Navigasjon”	s.44
3.4.1.12	Sekvensdiagramogkontrakterfor”Visningavbilde”	s.46
3.4.1.13	Sekvensdiagramogkontrakterfor”Slettingavbilde”	s.48
3.4.1.14	Sekvensdiagramogkontrakterfor ”Autentiseringogregistrering”	s.50
3.4.1.15	Sekvensdiagramogkontrakterfor”Brukeradministrasjon”	s.52
3.4.1.16	Sekvensdiagramogkontrakterfor”Loggføring”	s.54

4Begrensninger

4.1	Software-designbegrensninger	s.56
4.1.1	Softwarestandarderogspråk	s.56
4.1.2	Softwarepakker/verktøy/datbaserogoperativsystemer.....		s.56
4.1.3	Toleranser,marginer,mulighete r/tilfeller.....		s.57
4.2	Hardware-designbegrensninger	s.57
4.3	Brukerdesignbegrensninger	s.57

5Aspekteromkringlivssyklus

DatabaseforLagringogAnalyseavMikrostrukturbilder

5.1	Dokumentasjons.58
5.2	Modulogintegrasjonstesting s.58
5.3	Konfigurasjonsogversjonsstyring s.58
5.4	Kravtilsupport,serviceogvedlikehold.....	s.58
5.5	Kravtilutvidelser s.59

6Aspekterominstallasjon

6.1	Hardwareinstallasjons.60
6.2	Overgang/omlegging s.60
6.3	Oppl�ring s.60

7Akseptansekrav s.61**8Prosjektstyringinkludertkvalitetsstyring** s.61

1 Introduksjon

1.1 Bakgrunn

Raufoss ASA er et stort og anerkjent navn fra næringslivet i Norge. De har gått fra en barndomsomammunisjonsprodusent i 1896 til i 1999 å bli utpekt av GM (General Motors) som produsent for deres fjæringskomponenter av aluminium. Forretningsområdet de idag opererer på er: personbilkomponenter, vann & gass og komposittvirksomhet. Ammunisjonsdelen av bedriften har dens enerett blitt skilt ut som egen enhet, og blir nå drevet av den største aksjonæren som er staten. Staten eier for øvrig 50,3% av hele Raufoss konsernet. Selve hovedkvarteret til Raufoss er plassert i stedet Raufoss, selv om de allikevel har fabrikk rundt omkring i Tyskland, Sverige, Belgia, Tsjetsjenia, England og Nederland. Området de opererer ut fra er ganske unikt, både hargreid på få mange forskjellige selskaper med forskjellige eiere til å bosettes og her. Slik har samarbeidsformen mellom selskapene blitt mye tettere og enklere å forholde seg til. Bedriften har omkring 2400 ansatte og hadde i 1999 en omsetning på 2,062 milliarder norske kroner. Visjonen for fremtiden består i å spesialisere seg endast på færre felter, men å lage produkter innendiss som kan brukes innen flere fagfelt. På den måten håper de å kunne øke kundegrunnet.

Metallurgisk avdeling ved Raufoss konsernet tar bilder av og analyserer ulike materialer og sammensetninger foreventuelt senere bruk i industrien. Dadenne bilde -innhentingen har blitt foretatt via skannere, elektromikroskop og digitale kameraer så harden naturlig bevaringsmetode vært å lagre disse på PC. Dette har blitt gjort uten å ha et sentralt database system til å katalogisere og foreta en systematisk lagring av bildene. Dette har ført til at de innhentede data er blitt plassert rundt omkring på forskjellige deler av intranettet uten en ryddig struktur. De ansatte har laget det og da uten å ta hensyn til fremtidig bruk av bildene. Etter hvert som de ansatte har gått til å lete etter tidligere scannede bilder så har de innsett behovet for å ha et system som i varetardisse. Derfor tok en av forskerne fra Raufoss initiativ og tok kontakt med HIG for å høre om dette kunne være en potensiell oppgave for noen av deres elever. Etter et møte mellom Raufoss og representanter fra HIG, ble det i god tid før årsskiftet 2000/2001 bestemt at dette ble godkjentsom etthovedprosjekt for tre år datastudentene. To av dagens gruppe medlemmer var med på dette møtet og bestemt seg umiddelbart for å bli med på dette prosjektet. Deretter endte gruppe medlemmene som er på prosjektet idag ble integrert inn i gruppa etter årsskiftet.

1.2 Kort om kravtilsystemet

De essensielle egenskapene ved systemet:

- Detskal være et system som skal kunne brukes på alle klienter ved Raufoss
- Bilderskalkunne lagres med egen definert navn
- Bilderskalkunne hentes fram i klienters browsere

Databasefor Lagring og Analyse av Mikrostrukturbilder

Databasens skal være plattformuavhengig, og bilderskalderforikkellagres i filsystemet, men i databasen .

Dets kalkunnesøkes på tidlige lagrede bilder

Skalogså kunne søkes på bilders dato, legering, prosjekt, tittel og avdeling

Sletting av bilders kalkunneskje av eier av bildet

Systemets kalkunne håndtere 20 brukere maksimalt, hvorav 3 samtidig

Alle kalkunne se andre lagrede bilder

Muligheter for å tilknytte og lagre dokumenter til bilde(r)

Responstid skal være av en slik art at personer ikke må sitte å vente over lengre tid på sine resultater

Systemets skal håndtere bilder opp til 1900x1350 opp løsning og med fysisk lagringsstørrelse opp til 10 mb

De ønskelige egenskaper ved systemet:

Viser thumbnail av resultat på søkerbilder

Loggbok ved transaksjoner av bilder

Autentisering av bruker

Trashcan

Som et ledende ingeniørutdanningen inngår detaljerte av systemet er en oppgave som består i å utføre et prosjekt etter et oppdragsgivers ønske.

Da dette utgjør 4 av totalt 10 vektfullsår det begrenset tidsom kan benyttes til dette prosjektet. Dette setter derfor en viss takhøyde på antall timer som kan investeres i et slikt foretagende.

I regi av våre studenter kan en også skjønnat det ikke kan investeres noe særlig pengebeløp i et slikt utviklingsforløp. Så tid og penger er her svært begrenset ressurs.

1.3 Kort om systemets omgivelser

Da Raufoss i løpet av året kommer til å skifte ut sin hardware og oppgraderer dette, så blir systemets klienter bestående av Pentium 4 maskiner med mer enn nok RAM.

Servere kommer til å bli NT - bokser med 4 prosessorer og brå med minne.

Klientene kommer til å kjøre Microsoft operativsystemer med enten Explorer - eller Netscape browsere.

Intranett på Raufoss består av masse klienter og endelservere. Klientenes om dette prosjektet er relevant desom står i bygning 100 som tilhører materielle teknologiske avdeling. Disse vil måtte ha tilgang til databasen via sine respektive browsere.

1.4 Dokumentets struktur

Kravspesifikasjonene er utformet ved at del 1. N tar for seg det mer generelle rundt oppgaven. Bakgrunnen for initiativtaker, gruppesammensetning, definisjoner, overordnede brukerkrav osv.

Del 2. N er mer overordnet syn på selve det fysiske produktet brukernes skal få. Dette innebærer en litt nærmere syn på omgivelser, brukere, livssyklus osv.

Database for Lagring og Analyse av Mikrostrukturbilder

Når vi er trødler, ser det ut som at selve vedtaket til å lage et system som skal utvikles. Her omtales klassediagram, konseptuelle modeller, use case beskrivelser, sekvensdiagrammer osv. Dette klarlegger de krav og ønsker som stilles til utviklingsforløpet og gir et klart definert rammeverk for jobbet. Senere vil det bli brøst inn på begrensninger som er lagt på systemutviklingsforløpet ved hardware og software begrensninger. Dette viser oss konkrete rammeverksomsettergrenser for hvilke hjelpemidler vi kan bruke og hardware. 5. Noe handler om spekteret av livssyklus til produktet. Det viser hvilke livssyklusapplikasjoner som vil komme til å få krav til backup, support og testing. 6. Noe handler om installasjon og opplæring, mens 7 går inn på akseptansekrav og hva som må til for at systemet skal kunne godkjennes. Spesifikasjonen avslutter med en bit om prosjektstyring idel 8. N.

1.5 Definisjoner

Browser

Program som er lagd for å vise i nettsider (bygd opp av formateringspråk likhet med HTML). Å browse betyr å bla seg gjennom en rekke sider bestående av HTML sider.

Cookie

En liten tekstsnutt med brukerdata som blir brukt som informasjon bærer for visse web-applikasjoner. Altså en fil som Web -leseren mottar fra en Web -tjener. Filen lagres lokalt på harddisken din, og sendes tilbake til Web -tjeneren hver gang Web -leseren prøver å lese en side på tjeneren. Dette kan brukes f.eks. for å skreddersy Web -sider for hver enkelt bruker, ved at Web -tjeneren kan identifisere deg nestegang du logger deg på.

Frame

En del av oppskjembildet i mindre og teknisk individuelle vinduer som kalles frames. Disse kan betraktes som separate deler, selv om de utgjør en web -side. Men jobber og interagerer vanligst med hverandre.

GIF Graphics Interchange Format
Dette er et bildeformat som benytter pakking.

GUI Graphical User Interface
Grafisk brukergrensesnitt. Det visuelle vinduet som oppstår på dataskjermen.

HTML HyperText Markup Language
Dette er et formateringspråk som tillater bruk av grafikk, lyd, video og lenker til andre dokumenter. WWW bruker denne.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Hub

Enhetsomtillaterfleremaskineråværetilkobletenannenmaskin,ellerværetilkoblet ettnettverk.

JAVA

Ett"globalt"objektorientertprogrammeringsspråk.

JDBC JavaDatabase Compiler

Enstandardisertm åteåintegrerererelasjonsdatabasermedjavaprogrammer

JSP JavaServerPages

Htmlsidersomblirkonstruertogvistfraenlokalserver

NTboks

PCmeden -ellerflereprosessorermedtilhørendeMicrosoftNToperativsystem (serverversjon)

RAM Random AccessMemory

Datamaskinens"korttidshukommelse"

Relasjonsdatabase

Etsystemsomoppretterlogiskekoblinger mellom lagrede bilder og informasjon. Slik atseneregjenfinning forenkles.

SQL StandardQueryLanguage

Standardforforespørsler mot database r

Switch

Enhetsomfungerersomenformfor"bro" mellom forskjellige nettverk.

Thumbnail

En kopi av ettbilde som gjengis i en "liten" utgave. Det vises en forhåndsvisning i redusert størrelse.

Usecase

Beskriver endel av et større handlingsforløp i systemutviklingen. Etsystem som regel deltoppen mengdeslike usecase, som vil bli angert etter hvor viktige de er for at systemet skal kunne fullføres.

VPN VirtualPrivateNetwork

Leidelinjersombrukestil intern dataoverføringer.

Disse har da full mulighet for kryptering noe som gjør at nettet blir tilsynelatende privat.

1.6 Referanser

"Applying UML and patterns", Craig Larman

ISBN 0 -13-748880-7

"Thinking in Java", Bruce Eckel.

ISBN 0 -13 -659723 -8

"Hacking Java - the Java professional resource kit", Mark Wutka.

ISBN 0 -7897 -0935 -X

Callaway, Dustinn R.: Inside Servlets, Server - Side Programming for the Java Platform, Addison Wesley Longman, 1999

Eckel, Bruce: Thinking in Java, 2nd ed. Revision 1.1, 2000

MySQL AB, MySQL Reference Manual for version 3.23.33, March 2001

Schildt, Herbert: Java 2, Fourth Edition, The Complete Reference, Osborne/McGraw Hill, 2001

Stallman, Richard: GNU Emacs Manual Thirteenth Edition Updated for Emacs Version 20.3, 1998

Sun Microsystems, Java™ 2 SDK 5.0 Documentation, Enterprise Edition, 1.3 Beta Release, January 2001

Sun Microsystems, Java™ 2 SDK Documentation, Standard Edition, Version 1.3, 2000

The Apache Software Foundation, Jakarta TOMCAT Version 3.2.1 Documentation, 2000

2 Brukerb eskrivelser

2.1 Omgivelser

Dettesystemetskaltilfredsstillereettbehovforlagringavbilderfordeansatteved materiellteknologiskavdelingpåRaufoss.Deholdertilbygning100somerett forholdsvisstortbyggmedmangeklienter.

Servereogklienterforbundetviastjerneformet10/100mbitLANnettverk,hvorav switcherinnerstoghubberytterst.

VPN(VirtualPrivateNetwork)blirbruksomenekestensjonfordereasetintranett.

2.2 Systemetsbrukere

Brukerneavdettesystemetbeståravhøy utdannedepersoner,dogikkeutdannet innendata.Alleharværtikontaktmeddatamaskineriforbindelsemedjobbtidligere ogkjennertildevanligeMicrosoftproduktene.Debørderforhaetgodtgrunnlagtilå tilegnesegnyekunnskaperomdensoftware somnåskalutvikles.

MedtankepåatbrukernesfagfeltikkeerdatasåmåGUIlagespåenslikmåteatdet blirlettfordemåhåndtereogbrukesystemet.

Detvililøpetavutviklingsprosessenblilagtutenbetaversjonavsystemetpåskolens intranettslikatbrukernefraRaufosskantestedetsfunksjonalitetviainternet.

Dettevilmedføreatdevilkommeikontaktmedsystemetpåetttidligstadiumog forenkleropp-gavenmedopplæringnårprodukteterferdig.

Systemetskalkunbrukesavengruppe,mererellteknologi,noesomgjøratvikunhar engruppeåforholdeossilnårdetgjelderopplæringsbehovfordetferdigeproduktet.

Itilleggmådetnevnesatdetteprodukteteravenslikartatbrukerenepptrengerengren lengreinnføringforåkunnetad etibruk.

2.3 Funksjon

Systemetskalbeståavklient/serverløsningssomharfølgendefunksjonalitet:

FørstegangbrukerprøverålasteDLAM -sidenpåenklientvilhanmåttefåoppet skjermbildehvorhanmåskriveinnnavn,brukernavnogpassord (*autentiseringog registrering*).Detteskalhankunmåttegjøreddenneengangend,systemetvillagredisseopplysningeneieninformasjonsbærerpåbrukershjemmområde.Etterdenne innskrivingavopplysningerskalbrukerenslippesinnpåDLAMshovedside. Nestegangbrukerprøverålasteopphevedsiden,ogforøvrigallekommendeganger, vilhanikkemerkenoenautentiseringellerskriveinnnoesomhelst.Dettefordien autentiseringdavlforegåidetskjultesystemet.

BrukerenkanfinnefremmettiligereilagretbildepåsinlokalePCellerannetstedpå detlokalenettverk(*Lagrenyttbilde*).DettekanhansålagreIDLAMdatabasenmed tilhørendeattributterogdokument(er).Attributteneerf.eks.legering,dato,navn, prosjektosv.Dokumentetsom kanlagresvedbildetinneholdergjernemerspesifikke ogmeromfattendeopplysningerommotivet.

Detkansøkesettertidligereilagrededebildersomliggeridatabasenfråfør.

Dakanbrukerbestemmeattributteroginformasjonsøketalskjetteer

Databasefor Lagring og Analyse av Mikrostrukturbilder

(*Søkeetterbilde*).

En annen mulighet tillegges til "Søkeetterbilde" er å bruke et grafisk skjerm bilde som visertidligere lagrede bilder i databasen.

Dette vil tillate brukeren å få opp bildene i form av ettskjerm bilde som viser en hierarkisk trestruktur og som gir en total oversikt over bildene. Dens søke -løsningen blir da en mer intuitiv form å finne fram til på (*Navigasjon*).

Når brukeren har funnet ønsket bilde via en av de nevnte søkemetoder, vil bildet bli vist i browser (*Visning av bilde*).

Hvis tidligere lagrede bilder skal slettes kan dette skje via vanlige søk/navigasjon metoder (*Sletting av bilde*). Det er kun en av bildets omskaltillates å kunne slette det. Dette håndteres systemet selv ved å autentisere brukeren (*Autentisering og registrering av bruker*).

Slik sletting vil systemet automatisk loggføre i en intern liste som senere kan gi oversikt over historiske handlinger (*Loggføring*).

En utvalgt person/administratør skal ha mulighet for å bestemme hvilke brukere som skal kunne benytte seg av DLAM databasen (*Brukeradministrasjon*).

Funksjonene som er nevnt i kursiv i teksten ovenfor, skal være i systemet:

(Disse vil her etterblikalt *Use case*)

- Lagre nytt bilde
- Søkeetterbilde
- Navigasjon
- Visning av bilde
- Sletting av bilde
- Autentisering og registrering
- Brukeradministrasjon
- Loggføring

DatabaseforLagringogAnalyseavMikrostrukturbilder

Deforskjelligeusecaseharfølgendeoverordnededatasomskalkunnematesinnog medpåfølgenderesponstilbruker:

	INPUT:	OUTPUT:
Lagrenyttbilde	Tittel	
	Prosjekt	
	Avdeling	
	Dato	
	Legering	
	Kortinformasjon	Bildem/attributterog dok.
	Tittelpådokument	Bekreftelse
Visningavbilde	Initiertviasøk/navigasjon	Bilde
Navigasjon	InputviaGUI	Tittelpåbildem/ tilhørendeattributterog dokumenter
Søkeetterbilde	Tittel	
	Prosjekt	
	Avdeling	
	Dato	
	Legering	
	Tittelpådokument	Tittelpåbildem/ tilhørendeattributterog dokumenter
Slettingavbilde	Tittel	Bekreftelse
Autentiseringog registrering	Navn,brukernavn,passord	Bekreftelse
Loggføring	Bildeattributter	Tekstfil
Brukeradministrasjon	Brukernavn	Tillatelse
	Brukergrupper/navn	Bekreftelse

2.4 Operasjon

”Oppetiden” på systemets skal være kontinuerlig, under forutsetning at serverne er i drift og at systemet får kontakt med databasen. Hvis systemet mot formodning ikke skulle få kontakt med databasen, er programmet konstruert slik at bruker får tilbakemelding om dette.

Ved feil i databasen følger normale sikkerhetsregler ved å enten akseptere komplett transaksjon eller nekte hele overførselen. Dette vil føre til at systemet i varetar alle tidligere transaksjoner unntatt densom variferd med å overføres da systemet brøt sammen.

Sikkerhet med tanke på brukers identitet vil bli ivaretatt ved at systemet verifiserer brukere ved lagring og sletting. Dette vil forhindre at de som ikke eier bildene får mulighet til å slette andres bilder.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Viforutsetteratdenlokalesystemadministratorfølgernormalereglerforbackupog vedlikeholdsrutiner,dadetteikkeerunderstøttetiselveprogramvaren.

2.5As pekeromkringlivssyklus

Filosofienbakdetteprosjekteteratdetskalværeengjenbruksløsningav komponenter,ogprogramvareskalværeavdenfritt distribuerbaretypen. Dettemedføreratsystemetifremtidenvilværeenkeltåmodifisereogtilpasseti vekslendefysiskeforhold,samtidigsomdetergratismedtankepåsoftware. DasystemeterbygddoppavJava,JDBCogMySQLvildetteværeenenkelenhetå kjøre på andresystemer enn NT. Dette kan brukes på det mest som finnes av hardware og software løsninger idag. Databasefølger standarder og er en "gratis" programvareløsning.

1

En fremtidig utvidelse av dette systemet bør være problemfritt da Java er et globalt programmeringsspråk og da kildekoden vår blir gitt videre til Raufoss med full kommentering. Samtidig villøsningsmetodikkens vår være på en slik måte at det ikke legges restriksjoner på fremtidig utvidelse av systemet. Relasjonsdatabase som vi konstruerer etter dets samme prinsipp.

Vedlikehold vil fra systemadministratorsside bestå i å ta backup av loggfiler og relasjonsdatabase.

Fremtidig modifikasjon av komponenter og programvare vil måtte foregå på en inkrementell måte. Slik at hver enkelt komponent som legges til eller byttes ut vil måtte testes opp mot eksisterende funksjoner.

Den programvarens omgruppe vil overlates til Raufoss versjon 1.0.

Ved eventuell fremtidig modifisering bør det tas hensyn til versjonsnummerering, slik at en mindre modifikasjon vil gi et nytt nummer f.eks versjon 1.1.

Større forandringer bør gi utslag ved forandring av førstesiffer, f.eks 2.0.

Forandringer bør loggføres i egne kvalitets sikrings dokumenter og oppdateres for alle forandringer av programvare. På den måten vil det for all uoverskuelig fremtid være enkelt for nyansatte eller andre interesserte å tilegne seg informasjon om hvordan ting fungerer.

2.6 Ytelse

Log medat systemet vil agere kun skal brukes av maks 20 personer, hvorav maksimum 3 samtidig, vil det ikke måtte tåle noen enorm belastning av data mengder.

Bildenesom database skal håndtere er maksimalt 1900x1350, med en fysisk data størrelse på maks 10MB.

Responstid vil være avhengig av brukerne ikke må sitte å vente på at systemfunksjoner skal skje. Da i nettet er lite belastet og grasktså servii ikke nødvendigvis er nødvendig å oppgis spesifikk responstider.

DatabaseforLagringogAnalyseavMikrostrukturbilder

2.7 Begrensninger

Fordisystemets kalkjørespanye PC`er og servere med 4 prosessorer og mer enn nok minne, er det ikke lagtnoens spesielle restriksjoner på denne utviklingsprosessen i form av hardware. Lagringsmedier nå blitså passbillig at de ikke lenger er noen begrensning for denne oppgaven. Hvajelders software så er begrensningen at det kun skal brukes gratis programvare gjennom utviklingsforløpet.

2.8 Antagelser

Fundamentet vårt i starten av denne utviklingsprosessen er at brukerne på Raufoss har gitt oss korrekt informasjon vedrørende eksisterende hardware/software, fremtidig hardware/software, funksjoner, søkekriterier og utseendet. De programmerer vil velge rutsomt utviklingsverktøy vil vi i antahar allden funksjonalitet som ligger i kommersiell software, og at det er muligheter for å programmerer rundt eventuelle svakheter i disse programpakken. Det antas også at mange av de programmeringstekniske komponentene vil kunne bruke eksisterer fritt tilgjengelig utepå internet. Ut fra samtaler med oppdragsgiver går vi ut ifra at det vil bli satt opp server med tomcat og en MySQL database, slik at vår løsning kan installeres og integreres direkte med deres intranett.

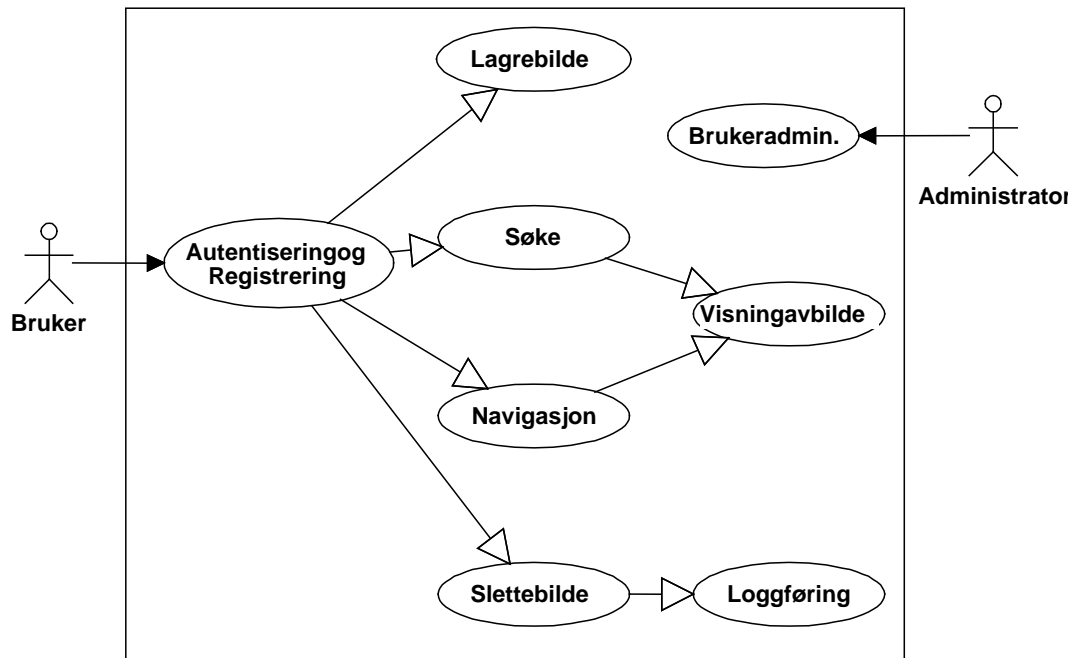
Denne programvareutviklingen vil baseres på å løse oppgåvene som består av eksisterende komponenter, derfor vil ikke den kravspesifikasjonen gå helt i dybden i likhet med andre spesifikasjoner. Ved normal bruk av UML og RUP vil som regelen utvikling av kravspesifikasjon gå så langt ned i utviklingsprosessen at det blir dannet ett kodeskjell. Dette vil vi da unngå da vi prøver å bruke de eksisterende navngivninger i de komponenter vi finner brukbare. Slik unngår vi ekstra arbeid ved å rette opp skadd kode i de eksisterende moduler. Vi vil bruke utviklingsprosessen i lærebok av vår systemutvikling 2, ”applying UML and patterns”, med en del restriksjoner. På denne måten kan den kravspesifikasjonen oppfattes mer abstrakt enn tilsvarende spesifikasjoner som er lagt for å generere start på kode.

3 Detaljert kravspesifikasjon

3 Funksjonell spesifikasjon

3.1 Systemfunksjoner, funksjonskategorier og tverr -relasjoner

3.1.1 USECASE DIAGRAM



Diagrammet overviser den i usecase`ssom utgjør systemutviklingsprosessen. Disse 8 interagerer på varierende vis for å få en helhet som tilbyr kundens ønskede funksjonalitet. Illustrasjonene som følger vil disse være fundamentet, i likhet med krav og ønskelige egenskaper fra kapittel 2.

På følgende overordnede beskrivelser av use case negirett grov rissomhvilken retning utviklingsprosessen må ta for å oppnå ønsket resultat.

Databasefor Lagring og Analyse av Mikrostrukturbilder
3.1.2.1 Usecasebeskrivelse og funksjonsliste for:

Usecase: **Lagrenyttbilde**
 Aktør(er): Eier
 Type: Primært
 Beskrivelse: En bruker finner et tidligere lagret bilde på det lokale nettverket. Dette velger han da navn og attributter til, samt oppgir informasjon som beskriver bildet. Han kan også tilknytte et eksternt dokument til bildet, for deretter å lagre dataene.
 Kryssreferanse: Autentisering og registrering.

LAGRENYTTBILDE(R):

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R6.N	Autentisering og registrering	Skjult	Systemevent	Sikkert	Må
R1.1	Finne bilde	Synlig	Interface metafor	Dialogboks	Må
R1.2	Bestemme attributter	Synlig	Logisk oppbygning	Brukervennlig	Må
R1.3	Skrive inn tilleggsinfo	Synlig	Enkelt	Brukervennlig	Må
R1.4	Finne tilleggsdok.	Synlig	Interface metafor	Dialogboks	Ønskelig
R1.5	Lagrebilde/ tilhørende info.	Skjult	Feiltoleranse	Ingen	Må

Typisk Handlingsforløp
Aktørhandling

1. Brukernyttbrukssystemet
4. Bruker velger bildet
6. Bruker velger attributter, beskrivelse
7. Eventuelt tilknyttet dokument til bildet
9. Bruker velger dokument
11. Bruker trykker "lagre"

Systemreaksjon

2. System godkjenner bruker
3. System viser dialogboks
5. Bildeinformasjon blir hentet
8. Systemet viser dialogboks (pkt. 3)
10. System tilknytter dokument til bilde
12. Oppdaterer system

3.1.2.2 Usecasebeskrivelseogfunksjonslistefor:

Usecase:	Søkeetterbilde
Aktør(er):	Bruker
Type:	Primært
Beskrivelse:	Brukerønskeråfinneettidligere lagretbilde. Hankansøkepåforskjelligekriterier. Resultatavsøkvildablivistpåbrukersskjerm.
Kryssreferanse:	Autentiseringogregistrering,visningavbildet

SØKEETTERBILDE(R):

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R2.1	Oppgir søkekriterier	Synlig	Brukervennlig	Visemulige attributter	Må
R2.2	Søkeretterbilde	Skjult	Responstid	Akseptabel	Må
R2.3	Viserresultat	Synlig	Bruker-vennlig	Viseres.påen oversiktelig måte	Må
R2.4	Velgerønsket bilde	Synlig			Må
R2.5	Viserbilde	Synlig	Responstid	Akseptabel	Må

TypiskHandlingsforløp
Aktørhandling

1. Dette usecaset starter med at bruker klikker på "Søke".

3. Bruker velger ett eller flere søkekriterier.

4. Bruker klikker på Søk.

6. Bruker velger blant de viste treff,

8. Bruker velger nytt søk.

Systemreaksjon

2. Viser søkefeltet.

5. Systemet søker og viser søketreff.

7. Systemet viser valgt bilde eller

9. Systemet går tilbake til pkt. 3.

3.1.2.3 Usecasebeskrivelseogfunksjonslistefor:

Usecase: **Navigasjon**
 Aktør(er): Bruker
 Type: Primært
 Beskrivelse: Brukerønskerenoversiktovertidligere lagrede bilde og dette får han ved å navigere seg gjennom trestruktur med filnavn. Deretter klikker han på ønsket bilde og navnet som deretter vil bli vist.
 Kryssreferanse: Autentisering og registrering, visning av bilde.

NAVIGASJON:

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R3.1	Viseren trestruktur	Synlig	GUI	Oversiktelig	Må
R3.2	Velger del av tre	Synlig	GUI	Oversiktelig	Må
R3.3	Velger prosjekt	Synlig	Responstid	Akseptabel	Må
R3.4	Velge fil	Synlig	GUI	Enkel	Må
R3.5	Vise bilde	Synlig	Responstid	Akseptabel	Må

Typisk Handlingsforløp
Aktørhandling

1. Brukervalg navigasjon
3. Brukervalg rett objekt i trestruktur
5. Velger bilde i tabell

Systemreaksjon

2. Viser navigering.
4. Viser objekt og det stilhørende informasjon i tabell
6. "Viser bilde" eller
7. "Sletter bilde"

DatabaseforLagringogAnalyseavMikrostrukturbilder

3.1.2.4 Usecasebeskrivelseogfunksjonslistefor:

Usecase: **Visningavbilde**
Aktør(er): Bruker
Type: Primært
Beskrivelse: Brukerharfunnetønsketbildeog bersystemetomåvisedettei påskjermen.
Kryssreferanse: Søkeetterbilde,navigasjon.

VISNINGAVBILDE:

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R4.1	Hentebildem/ attributterfra database	Skjult	Responstid	Akseptabel	Må
R4.2	Henter tilleggsdok.	Skjult	Responstid	Akseptabel	Må
R4.3	Viserbildet	Synlig	Feiltolerans	Ingen	Må

TypiskHandlingsforløp**Aktørhandling**

1. Brukerharfunnetønsketbilde
3. Brukervalgerettobjekt
trestr uktur
5. Velgerbildeitabell
6. "Viserbilde"

Systemreaksjon

2. Visernavigasjon
4. Viserobjektogdetstilhørende
informasjonitabell

3.1.2.5 Usecasebeskrivelseogfunksjonslistefor:

Usecase: **Slettingavbilde**
 Aktør(er): Eier
 Type: Primært
 Beskrivelse: Girbrukermulighettilåslettesineegnetidligerelagredebilder.
 Kryssreferanse: Autentiseringogregistrering.

SLETTINGAVBILDE:

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R5.1	Finnerønsket bilde	Synlig	GUI	Oversiktlig	Må
R5.2	Verifiseringav bruker	Skjult	Feiltoleranse	Ingen	Må
R5.3	Slettebilde	Skjult	Feiltoleranse	Ingen	Må
R5.4	Tilbakemelding	Synlig	Interface metafor	Dialogboks	Må
R8.N	Oppdaterer logg	Skjult	Feiltoleranse	Ingen	Ønskelig

TypiskHandlingsforløp
Aktørhandling

1. Brukerharvalgtobjekt
2. Velger”slette”objekt

Systemreaksjon

3. Systemslatterobjektmedtilhørende referanseridatabase
4. Bekreftelsepåsletting

3.1.2.6 Usecasebeskrivelseogfunksjonslistefor:

Usecase: **Autentiseringogregistrering**
 Aktør(er): Bruker
 Type: Sekundært
 Beskrivelse: Idetbrukerønskeråbenyttesegavsystemetvildetautomatisk hentesinformasjonombbrukersidentitet,event ueltlagres informasjonomdennehvisdeterførsteganghanbenytterseg avdet.
 Kryssreferanse: Lagringavbilde,slettingavbilde,navigasjon,søke.

Autentiseringogregistrering:

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R6.1	Undersøkerom bruker finnes frafør	Skjult	Feiltoleranse	Ingen	Må
R6.2	Hvisbruker finnesslippes denneinni systemer	Skjult	Feiltoleranse	Ingen	Må
R6.3	Hvisikkescal bruker registreres	Skjult	Feiltoleranse	Ingen	Må
R6.4	Brukerslippes innisystem	Skjult	Feiltoleranse	Ingen	Må

TypiskHandlingsforløp

Aktørhandling

Systemreaksjon

1. Brukerhenterwebseiteviabrowser

2. Systemsjekkerombrukereksisterer:

 a. Hvisjase *Eksisterendebruker*

 b. Hvisneise *Nybruker*

3. Systemetgodkjennerbruker

4. Brukerkanbenytte systemet

Seksjon:Eksisterendebruker

- 1.Brukerharhentetwebsiteside
- 2.Systemetfinnerbrukericookie
- 3.Systemet slipperbrukerinn

Seksjon:Nybruker

- 1.Brukerharhentetwebsiteside
- 2.Systemetfinnerikkebrukericookie
oggirderforbrukerbeskjedomåskrive
innattributteromsegselv
- 3.Brukerskriverinnnavn,brukernavnog
passord
- 4.Systemetlagrennybrukerogslipper
denneinnidatabasen

3.1.2.7 Usecasebeskrivelse og funksjonsliste for:

Usecase: **Brukeradministrasjon**
Aktør(er): Administrator
Type: Tilleggsfunksjon
Beskrivelse: Administrator vil ha mulighet for å opprette og slette brukere og brukergrupper.
Kryssreferanse:

BRUKERADMINISTRASJON:

Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R7.1	Logger seg inn	Synlig	Sikkerhet	Absolutt	Må
R7.2	Velger ønsket funksjon	Synlig	GUI	Enkel	Må
R7.3	Systemutfører valgte funksjon	Usynlig	Feiltoleranse	Ingen	Må
R7.4	Admin logger ut	Synlig	Feiltoleranse	Ingen	Må

Typisk Handlingsforløp**Aktørhandling**

1. Velger modus "brukeradministrasjon"
2. Logger på
4. Velger ønsket funksjon
6. Logger ut

Systemreaksjon

3. Viser brukeradministrasjonsside.
5. Systemutfører valgte funksjoner

3.1.2.8 Usecasebeskrivelseogfunksjonslistefor:

Usecase: **Loggføring**
Aktør(er): Eier
Type: Tilleggsfunksjon
Beskrivelse: Vedslettingavbilderoginformasjonvildettebliloggførtiegen
listesomer ”synlig”forbrukerne.
Kryssreferanse: Slettebilde.

LOGGFØRING:

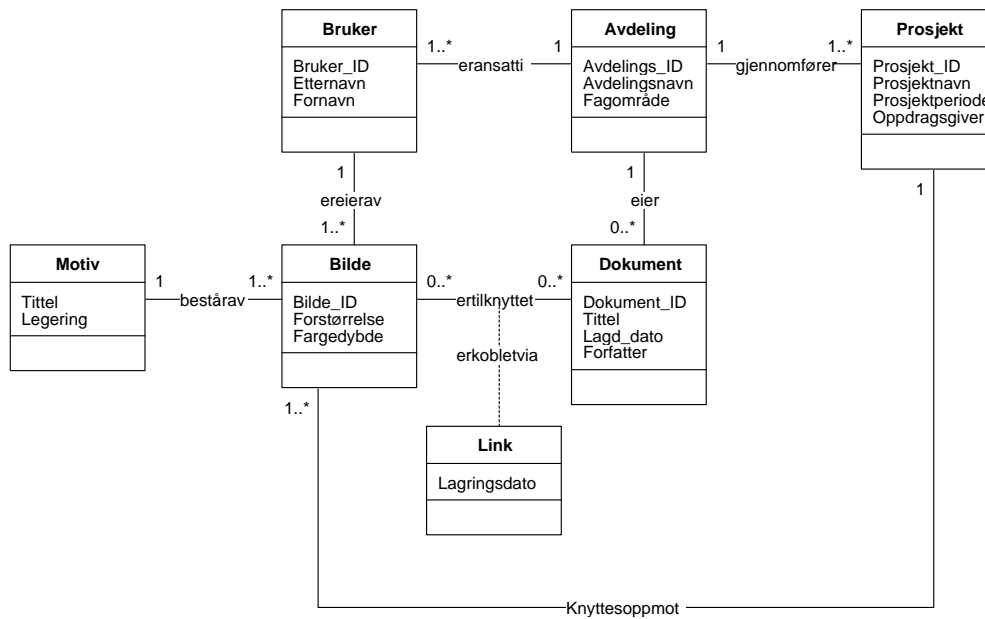
Ref#	Funksjon	Kategori	Attributt	Begrensninger	Kategori
R5.N	Slettingav bilde	Synlig	Feiltoleranse	Ingen	Må
R8.1	Leggerinn slettingsinfoi liste	Skjult	Feiltoleranse	Ingen	Må
R8.2	Lagrerjustert liste	Skjult	Feiltoleranse	Ingen	Må

TypiskHandlingsforløp**Aktørhandling**

1. Brukerharvalgtåslettebilde

Systemreaksjon

2. Systemlagrerendringer ilogg

3.1.3 KONSEPTUELLEMODELLEN:


Dette er systemmodellen sett med et virkelighetsperspektiv. Det vil si at vi formulerer systemet med en illustrasjon som viser de fysiske komponenter vi finner i virkeligheten og ikke som softwaremoduler. Dette gir oss en foreløpig abstraksjon av den kommende softwaremodelleringen. Denne modellen vil føre frem til neste avsnitts foreløpige utkast til klasser.

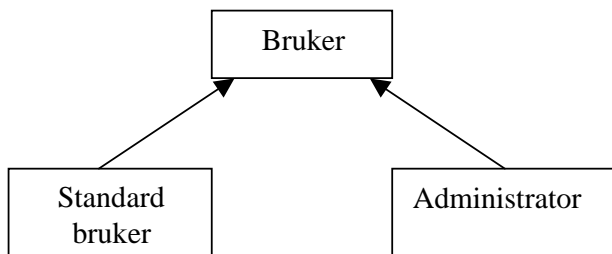
3.2 Dataspesifikasjon

Systemetskalbeståaven modulsomivaretarbeildersomskalleggesinnibasen. Viderevildetbeståavenmodulsomvilsørgeforavdeling,enforbrukerogentildokumenter.

Moduler:



Klassenbrukermånødvendigvishånderetoforskjelligtypebrukere. Denenevil væreenstandardbruker,mensdenandrevilværeenadministrator. Envilmåttetahensyntilatenadministratorvilkunneværeenbruker,menenbrukerkanikkeværeenadministrator.Klassenadministratorvilværeenforlengelseavbrukermodulen,med tilsvarendetilleggsfunksjoner.Dennevilværenavansert versjonavbrukerhvordetimplementeresfunksjonerforstyringogvedlikeholdavsystemet.Dennevilogsåinneholdeekstraattributtersettiforholdtilbruker.



Beskrivelse:

Avdeling:

Vil inneholdefunksjonerogattributterfordeforskjelligeavdelingstilfeller. Detvilhermåtteværeattributterforblantannethvilkeprosjekttersomstammerfra hvilkenavdeling,avdelingsnavn,identifikatoroginnehafunksjonerforsøkingog slettingavdisse.

Bilde:

Hervildetbilibliggendeattributtersomivaretaretbildesdedikerteinformasjon. Herunderenentydigidentifikator,navn,forstørrelse,datooogmotiv. Funksjonersommåimplementereskanværelagring,slettingogsøking.

Bruker:

Databasefor Lagring og Analyse av Mikrostrukturbilder

Iden nemådetnødvendigvis ligge oversikt over brukersentydige data. Blant disse kan det være navn og identifikator. De metodene som er tilbørlig kan være søking, sletting, lagring og medattributter som angir om de er administratoreller standardbruker.

Dokument:

Etdokument må inneha tittel, identifikator, forfatter og dato. Det bør kunne søkes på, lagres og slettes.

3.2.1 Datainput

Alle disse registrene vil bli oppdatert manuelt via tastatur. Bildersom skal innlegges i basen hentes fra nettverk, eksternt lagringsmedium eller internet.

Det skal implementeres en feilsjekkingsrutine for felte hvor korrekt input er av stor betydning. Blant annet vil det måtte legges en sjekk i inntastingsfeltet for bildenavn, som kontrollerer om det innta stede bildeformat finnes i systemet. Det vil si at hvis en bruker prøver å legge inn et bilde med ekstensjon kalt .xxx og systemet ikke kan vise det, så vil brukeren måtte gis beskjed om dette. Slik at bildersom ikke kan vises vil bli nektet adgang til basen.

Feil i inntasting i søkefeltet vil det kanskje testes på, slik at en får tilbakemelding om en skriver inn feil.

Utgangspunktet vil systemet bli designet på en slik måte at en kan legge inn alle mulige formater, ved en mindre justering av det programmet som er utviklet teknisk. Slik at hvis Raufoss i fremtiden vil ha behov for å utvide applikasjonen til å omfatte lagring av f.eks. video, så vil dette være mulig.

3.2.2 Dataoutput

Systemets skal gi output til brukeren i form av visning av ønsket bilde i eget browser vindu. Eller i informasjon om bildet, bruker, avdeling eller dokument skal vises i oppsatt tabeller i form av tekst. Dokument skal også kunne tas opp for lesing. Informasjon som skrives ut skal være av en slik art at det vil være trielt foren ukynlig data bruker å kunne lese og lete seg fram. Hvis ønskelig skal det også implementeres en funksjon for thumbnail generering. Slik at brukeren kan få en mindre representasjon av bildet han for øyeblikket ser på i.

3.2.3 Tverrfunksjonale data definisjoner

Dette vil bli en gjennomsnittsløsning vil vi gå for et system hvor tett kobling mellom komponenter unngås. Dette for at det i fremtiden skal kunne påbygges og ekspanderes. Der hvor moduler kobles vil det måtte gis en konsistent navngivning på variabler. Disse vil måtte forklares inngående i detaljert beskrivelse i kodestruktur.

3.3 Overordnede operasjonelles systemkrav

3.3.1 Normaloperasjon

3.3.1.1 Modus og kontroll

Det eksisterer seks forskjellige hovedmoduser i dette systemet, disse er:
lagrenyttbilde , **slettebilde** , **søkeetterbilde** , **navigere** , **visebildet** , **autentisering**
og **registrering** .

Når en bruker logger seg på for første gang vil han måtte registrere navn og passord. Dette er en gang for et se og nest gang bruker åpner nettsiden vil han komme rett inn i systemet uten å måtte logges på. Da **autentisering og registrerings** rutinen allerede etter første gang har lagt informasjonen om brukeren. **Lagrenyttbilde** er primærfunksjonen i systemet. Den håndterer nye bilder som skal katalogiseres og oppbevares i databasen. Deler av applikasjonen bør ha kontroll for å sjekke at samme bilde ikke blir lagt to ganger. Dette blir den mest brukte delen og vil nødvendigvis være svært stabil og sikker. Når et slikt bilde skal lagres må det samtidig skrives inn en del informasjon som senere gir oss mulighet for å søke og finne det frem igjen. Denne informasjonen vil bli bestående av en rekke attributter som er definert av brukerne ved Raufoss. Disse attributter dannet grunnlag for funksjonen **søkeetterbilde** som gir oss en rask og sikker måte å finne tidligere lagrede bilder. Et slikt søk må nødvendigvis foregå ved at vi skriver inn søkekriterier i forhåndsbestemte felter for deretter å initiere søk. Resultatet av et slikt søk vil avises i tabeller og gir oss mulighet for å **visebildet** . Dette ved at bildet kommer opp i egen rute på skjermen i full størrelse. En alternativt veifradetå vil søke etter et bilde til at det er opp på skjermen, vil kunne være å **navigere** seg frem. Dersom vi i stedet for å bruke søkekriterier bare kunne "klikke" oss gjennom tre struktursom blir dannet på skjermen. Slik at filen på lagringsmediet vil være illustrert via ikonene i strukturskjerm. Når en klikker på et av disse vil informasjonen til disse bli vist i tilsvarende tabell som hos søkeetterbilde. En naturlig funksjon som må være med her i og med at man geting legges inn **slettebilde** . Denne vil gjøre at vi får muligheten til å slette bilder som ikke lenger har noen funksjon i data basen. Slik at vi slipper at det hopper seg opp med bilder som egentlig ikke lenger har noen relevans for Raufoss.

Itillegget til disse seks hovedmoduser vil det være åpent for å implementere det til for å underlette vedlikehold og sikkerhet ved systemet.

Dette vil være **loggføring** og **brukeradministrasjon**.

Når en bruker kansletter tidligere lagrede bilder vil dette kunne medføre at hans letter data han muligens ikke skulle ha slettet. For å gi brukeren mulighet til å se hva han historisk sett har gjort så kan det implementeres en form for loggføring av brukers handlinger. Dette vil bestå av en liste som inneholder hvilke brukere tidligere har slettet, når og attributter.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Medtankepååholdedatabasenryddigogvedlikeholdtsåkan detogsåprogrammeres oppen egenadministratørfunksjon. Denne vil gi en person mulighet til å innog bestemme og rydde opp i brukere og brukergrupper. Samtidig som hankanselogg.

3.3.1.2RangeringavUsecase

Etter kritiske vurderinger, ifellesskap med utvalgte fra Raufoss, har vi funnet ut at Usecase bør rangeres på følgende vis ut ifra hvor kritiske de er for at dette skal bli et system som tilfreds stiller brukernes krav og ønsker:

Rangering:	Usecase:	Type:
1	Lagrenyttbilde	Primært
2	Visningavbilde	Primært
3	Navigasjon	Primært
4	Søkeetterbilde	Primært
5	Slettingavbilde	Primært
6	Autentiseringogregistrering	Sekundært
7	Loggføring	Tilleggsfunksjon
8	Brukeradministrasjon	Tilleggsfunksjon

3.3.1.3Ytelse

Etter bruker har vært på loggetgang skal applikasjonen være i stand til å gjennkjennedens samme bruker en neste gang han logger på, uten at den trenger å skrive inn noe. Nestegang hans skriver inn webadressens så vil hans slippe direkte inn på DLAM sidene. Programmets kalfunger er helt som andre web-applikasjoners likat start og stopp av nettsideskjerproblem fritt.

Med hensyn til brukernes skalventetid ved forskjellig bruk av system holdes på et minimum. For å sette en referanse kan vi si at dette skal holdes på et slikt nivå at en ikke overstiger 5 - 10 sekunder på en bildeinnlegging med en normal bildestørrelse eller et søk. I tillegg vil systemet, etter brukernes ønske, være dimensjonert for opp til 10 mb størrelse på bildene, men dette betraktes som å pass stort at ventetid vil kunne bli noe lengre. Det vil også implementeres algoritmer for å unngå dobbeltlagring av desamme bildene.

Ved bildeinnlegging vil det ikke være av noen betydning om bildet hentes fra lokal PC, nettverk eller lagringsmedium. En slik bildehenting vil ikke overstige den tiden tar å lagre et nytt bilde. Men forholdesegnoen lundeliktidsmessig sett.

Når det gjelder lagringsplass så gjorde Raufoss det klart at dette ikke var av relevans, da lagringsmedier idag har blitt så billig. Så det å minimere den plass data en tar er ikke et stort hovedpoeng under den utviklingsprosessen.

Vedrørende backup rutiner så vil det ikke implementeres noen spesielle funksjoner med tanke på dette hensendet. Men det skulle allikevel ikke by på noen problemer for administrator å ta sikkerhetskopier av databasen.

I og med at brukerne ved Raufoss ikke er utdannet innen data eller har noen spesiell bakgrunn innenfor det, så må GUI designes med dette i tankene. Dette fører til at GUI

DatabaseforLagringogAnalyseavMikrostrukturbilder

måværeintuitivtogenkeltåbruke. Detmåsålangtdetermuligu nngåsatennå
brukeinnvikledesystemeravlistbokserogcheckbokser. Istedetbørdetsatsespå
predefinerteverdierogenklemensystemermedetinnbydendeutseendet.
Mendetviktigsteeratdetforblirenkeltåbruke. Slikatenforhindreratdetoppst år
romformisforståelser.
IogmedatdettesystemetskalbyggesoppavJavaogstandarderienklient -server
løsning, såskaldetlagesslikatdetskalkunnekjøresfraforskjelligeklienterog
nettverk. Herunderforskjelligeoperativsystemerogsoftw areløsninger.
Slikatsystemetskalkunneoverførestilandredatabaser, somf. eksOracle.
EllerkjørespåbådeWin2000ogLinux. Derforerdetrelativtviktigatstandardersom
HTML, JSP, SQLosv. følges.

3.3.1.4Sikkerhet

Klient-serverløsningenvi lværeoppedøgnetrundtheleåret.
Medenegentilgangforadministratorvildetikkeværenoeproblemfordenneågå
innogkunnevedlikeholdeinformasjonensomliggередatabasen.
Nårdetgjelderback -upvildetteavhengeavdelokalerutineradminist ratorharfor
dette, dasystemetisegselvikkeharfunksjonerforslikt.
Vedrørende **autentiseringogregistrering** såvilpåloggingsrutinenværeavenslikart
atbrukerikkekanbrukehverandresidentitet. Enesteunntaketherkanværehvisen
brukertar overenannenbrukersalleredepåloggedePC. Davildennekunnemisbruke
denandresidentitetisystemet. Menenmågåutifraatallebrukereerpåpasseligemed
åikketilteandreåmisbrukedereskonto.
Førstegangenenbrukerloggersegpåvilhanbl ibedtomåskriveinnbrukernavnog
passord. Dettevilværelagretisystemetogværeuniktforakkuratdennebrukeren.
Nesteganghanloggersegpåvilhannikkemerkenoetildenaautentiseringensom
foregåribakgrunnen. Menivirkelighetenvildetforeg åensjekombrukervirkeliger
registrertisystemet. Denstørstegrunnentildetteeratenbruker kunsalkunneslette
debilderhanharlagret selv. Dettevilføretilatdeforskjelligepersonersbildertil
kunnesesavallemenkunslettesaveier en.

Etavdeviktigstekriteriene frabrukerneeratsystemetskalivaretalagringenpåen
sikkermåte, ogatdetskalværedriftssikkert.

Detskalkunnesifrahvisenbrukertasterinnugyldigekombinasjoneravtegni
attributtfelteneoggitilbakemeldi ngomdette. Detvilkunnesjikkeomfilformateter
gyldigviapredefinerteverdier. Brukerskalkunnefåenmeldingførslettingavbilde.

3.3.1.5Oppstartognedtagning

Førstegangenenbrukerloggerseginnpåsystemetskalhanmåttetasteinnnavn,
brukernavnogpassordtoganger, førmanslippesinntilselveDLAMhovedmenyen.
Dennemenyenvilblibasertpåkjøringietbrowser -vindu, ogkanderforbareliggei
bakgrunnenmensbrukerenjobbermedandreapplikasjoner.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Nårmanlukkerbrowserenvildett ikkeføretilnoenunormaloppførsel,menfungere akkuratsomenhverannennettside.

3.3.1.6Tilgjengelighet

Systemetskaliutgangspunktetkunnekjøres24timeridøgnetåretrundtuten nevneverdige problemer. Detvilsannsynligvisblittnedenganginnimellomfor vedlikehold, mendetteskulle daikkegjørenoehvisoperatørutførerdettoforsvarlig medgodebackuprutiner.

3.3.1.7Innebygdetester

- Underførstepåloggingvilnavn,brukernavnogpassordblilagretisystemet.
- Denestegangeneen brukerloggerpåvilsystemetsjekkedenneinfo.
- Attributfeltertilblisjekkettmotpredefinerte verdier.
- Filekstensjonen vilblisjekkettmotlegale verdierisystemet.
- Bildersomskallagresogsomminneromhverandrestørrelsesmessigvilbli sjekketoppmothverandreslikatenunngårålagresammebildettoganger. Dettevilskjeviaalgoritmerpåsjekksm.

3.3.2Operasjonifeilsituasjoner**3.3.2.1Feilrapportering**

Vedfeilinntastingifelturvildetmåttegiltibakemeldingtilbrukere nomathanhar gjortenfeil. Foreksempelhvisenbrukerbenyttersegavenukjentfilekstensjonsåvil systemetgitilbakemeldingtildenneomatekstensjonenikkeeksistererisystemetog ikkekanbrukes. Dettemådanødvendigvisgispåenslikmåtea tbruketikkeblir ytterligereforvirret. Detmåogsågismeldingipåloggingsrutinenhvisinformasjon ellerbrukerikkeblirgodkjent.

3.3.2.2Gjenervervelseetterfeil

Vedrørendelagringsåvildetteikkeskjeførbrukerharklikket”lagre”. Slikatet eventuellavbruddmellom”funnetbilde”og”klikketlagre”vilkunne medføreatbildetikkeblirlagret. Dettevildaikkeblibetraktetsomfeildabildetvilliggelagretpådetsammestedet somføravbruddet. Operasjonenvildabaremåttegjentasengangtil.

3.3.2.3Sikkerhet

Denenestesikkerhetsfunksjonensomhervilblilagtinner **autentiseringog registrering**. Dennevilbesørgeatbrukerikkeletterhverandrestidligerelagrede bilder.

Databasefor Lagring og Analyse av Mikrostrukturbilder

Det vil i kommandeutvikling bli implementert algoritmer omforhindrer at bilder blir lagret i databasen. Dette selv om bilder lagret under forskjellige navn. Dette for å forhindre unødvendige løsninger med disk -ressurser.

3.3.2.4 Ytelse

Det ble fra brukernes side lagt vekt på at systemet ikke skulle bli for tidkrevende å bruke. Med dette menes at under normal belastning ved innlegging av bilder skal det helst ikke gå over 10 sekunder. Tilsvarende vil det være en slik grense for **søketter bilder og navigasjon**.

Det ble fra de ansattes side sagt at absolutt maksimal bildestørrelse vil bli 10 mb, men med et normalt snitt på 2 -3 mb.

IT ansvarlig ved Raufoss ASA la vekt på at diskplass ikke er av noen betydning så derfor kom det til å bli et lite etterretning og ikke legges noen begrensninger i den tekniske utførelsen.

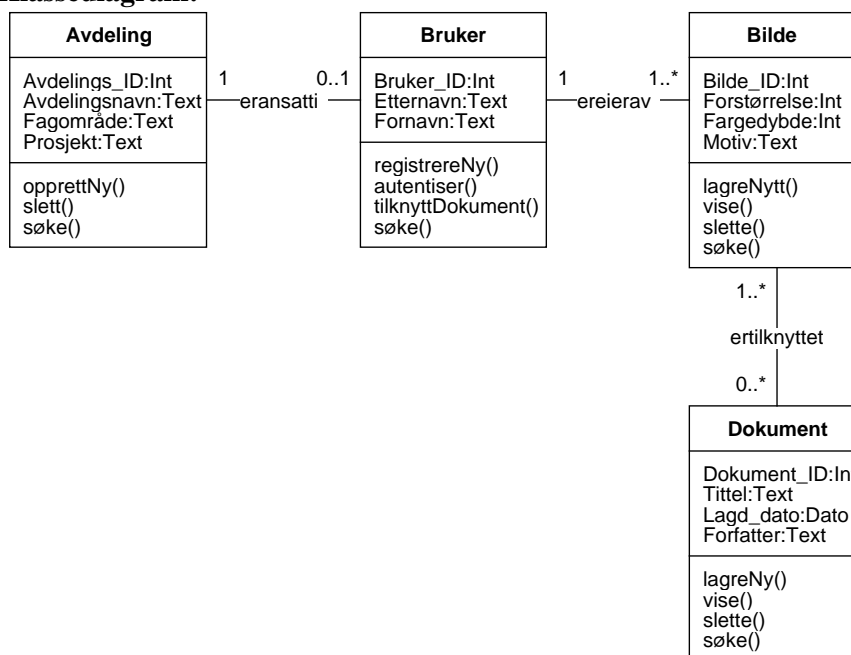
Antall brukere kommer maksimalt til å være 10, hvor av maks 4 samtidig. Systemet vil bli dimensjonert deretter og med kontroll for flerbrukere samtidig.

Mendete heller ikke spørsmål om den helt store brukerpågangen.

3.4Funksjonellekrav

3.4.1FunksjonellekravtilDLAMsystem

Klassediagram:



KlassenAvdeling:

Attributter for klassen Avdeling :

1. Avdelings_ID
2. Avdelingsnavn
3. Fagområde
4. Prosjekt

Operasjoner for klassen Avdeling :

1. Opprettenyavdeling
2. Slettegammelavdeling
3. Søkettereksisterendeavdeling

KlassenBruker:**Attributter** forklassenBruker :

1. Bruker_ID
2. Etternavn
3. Fornavn

Operasjoner forklassenBruker :

1. Oppretteennybruker
2. Autentiserbrukeroggodkjennedenet ilbrukavsystem
3. Tilknytteetdokumenttilvalgtbilde
4. Søkeetterbruker

KlassenBilde:**Attributter** forklassenBilde :

1. Bilde_ID
2. Forstørrelse
3. Fargedybde
4. Motiv

OperasjonerforklassenBilde:

1. Lagrenyttbilde
2. Visevalgtbilde
3. Slettevalgtbilde
4. Søkeette rbildetviakriterier

KlassenDokument:**Attributter** forklassenDokument :

1. Dokument_ID
2. Tittel
3. LagdDato
4. Forfatter

Operasjoner forklassenDokument :

1. Lagrenytttdokument
2. Viseeksisterendedokument

DatabaseforLagringogAnalyseavMikrostrukturbilder

3. Slettevalgtokument
4. Søkeettertidligerelagretokument

Underfølgerkortfattetoversiktoversystemetsforskjelligekomponenter:

Systemordbok:

Terminologi	Kategori	Kommentar
Prosjekt	Type	Prosjektetsombildettilhører
Prosjekt.prosjekt_ID :Int	Attributt	Etnummersomerentydigogidentifisereret prosjekt
Prosjekt.prosjektnavn :Text	Attributt	Navnetpådenavdelingsombildeter knyttetil
Prosjekt.prosjektperiode :Dato	Attributt	Varaktighetenpåprosjektperioden
Prosjekt.opdragsgiver :Text	Attributt	Navnetpådeneksterne/interne oppdragsgiver
Avdeling	Type	Avdelingensombrukerogprosjekt tilhører
Avdeling.avdelings_ID :Int	Attributt	Etnummersomerentydigogsom identifisererenavdeling
Avdeling.avdelingsnavn :Text	Attributt	Navnetpåavdelingensombrukeransatt i
Avdeling.Fagområde :Text	Attributt	Defagfeltetsomavdelingenoperereri
Dokument	Type	Etdokumentsomskaltilknyttsetbilde
Dokument.dokument_ID :Int	Attributt	Etenydignummersomidentifisererhvertenkeltdokument
Dokument.tittel :Text	Attributt	Tittelenpå hvertenkeltdokument
Dokument.lagd_dato :Dato	Attributt	Datoensomdokumenteterlagd
Dokument.forfatter :Text	Attributt	Forfatterenitlhvertenkeltdokument
Link	Type	Enkoblingmellometbildeogeteller fleredokumenter
Link.lagringsdato :Dato	Attributt	Datoensometbildeblikknyttetopptilet dokumentoglagret
Bruker	Type	Brukerensomereieravetbilde
Bruker.bruker_ID :Int	Attributt	Etenydignummerforhvertenkeltbrukerav basen
Bruker.etternavn	Attributt	Brukerensetternavn

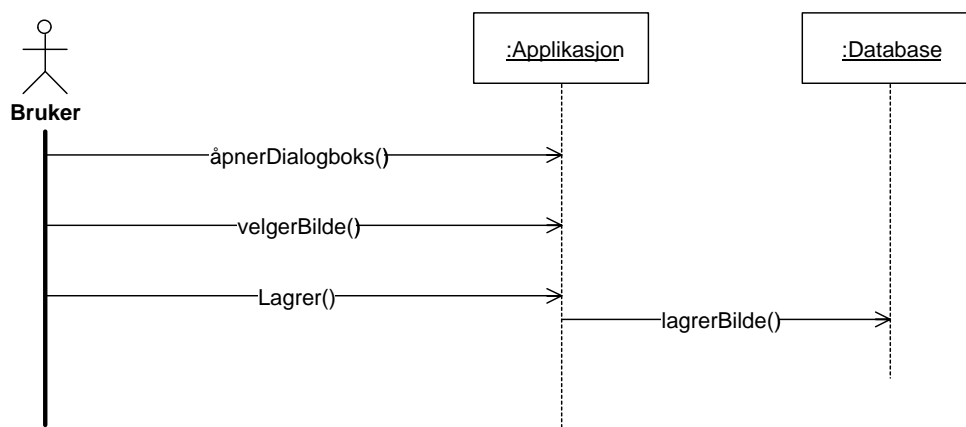
DatabaseforLagringogAnalyseavMikrostrukturbilder

:Text		
Bruker.fornavn :Text	Attributt	Brukerensfornavn
Motiv	Type	Detmotivetsomerpåbildene
Motiv.tittel :Text	Attributt	Etnavnpåhvertenkeltmotiv
Motiv.Legering :Text	Attributt	Navnetpådenspesifikkelegeringsomer avbildet
Bilde	Type	Debildersomb asenskalivareta
<i>Lagrenyttbilde</i>	Usecase	Beskrivelseavprosessennårenbruker lagrerentnyttbildeidatabasen
<i>Søkeetterbilde</i>	Usecase	Beskrivelseavprosessennårenbrukervil finnefremtiltidligerelagredebilder
<i>Navigasjon</i>	Usecase	Beskrivelseavprosessennårenbrukervil navigereseframtiltidligerelagrede bildervedhjelpavengrafisk representasjonavdisse
<i>Visningavbilde</i>	Usecase	Beskrivelseavprosessennårenbrukerhar funnetønsketbildeogsystemetskalvise dette
<i>Slettingavbilde</i>	Usecase	Beskrivelseavprosessennårenbruker ønskeråslettetidligerelagredebilder idatabasen
<i>Autentiseringog registrering</i>	Usecase	Beskrivelseavprosessennårenbruker skalloggesegpåsystemet
<i>Brukeradministrasjon</i>	Usecase	Beskrivelseavprosessennåren administratørskalredigerelisteover tillattebrukereogbrukergrupper
<i>Loggføring</i>	Usecase	Beskrivelseavprosessennårsystemet loggførerhvilke,hvordanoghvemsom sletterfiler

Systemsekvensdiagramme rogakontrakter:

3.4.1.1 Sekvensdiagram og kontrakter for:

"Lagrenyttbilde" -versjon1



Kontrakter for *LagreNyttbilde* Usecase

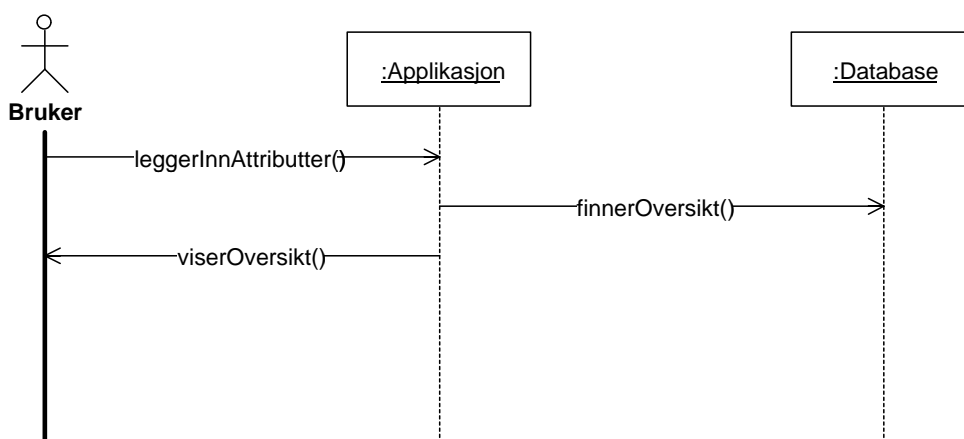
Navn:	åpnerDialogboks
Ansvar:	Sørgerforåfåoppdialogboksmedtilhørendealte rnative nårbrukerklikkerlagreny
Type:	System
Kryssreferanser:	
Notater:	
Unntak:	
Output:	
Pre-tilstand:	Bildeteksistererpålokalharddisk
Post-tilstand:	*Hvisbildeteksisterer,bliradressentilbildetlagtinniform. *Formdannesoger klartilmanipulering

DatabaseforLagringogAnalyseavMikrostrukturbilder

Navn: **velgerBilde**
Ansvaer: Valgtbildelagressammenmedattributterogeventuelt dokumenter
Type: Applikasjon
Kryssreferanser:
Notater: Brukermodulforgjenkjenningavbildetype
Unntak: Hvisikke blantakseptertefiltypernekteslagring
Output:
Pre-tilstand: Bildeteksistereriakseptertformat
Post-tilstand: Bildeterlagretsammenmedtilhørendeattributterog dokumenter

Navn: **lagrer**
Ansvaer: sørgerforåinitiereenlagringsfunksjon
Type: Applikasjon
Kryssreferanser:
Notater:
Unntak:
Output:
Pre-tilstand: Brukertryktlagre
Post-tilstand: InitiererlagreBildemedpåfølgendelagring

Navn: **lagrerBilde**
Ansvaer: Ivaretarlagringinnidennfysiskedatabasen
Type: Applikasjon
Kryssreferanser:
Notater: Foretarensikkerogkonsistentlagringinnmotdatabasen
Unntak: Vedserverfeilellerannetavbruddskaltransaksjonennektes godkjent
Output: Bildetlagres
Pre-tilstand: lagrerinitierer
Post-tilstand: BildetblirlagretinniMySQLbasen

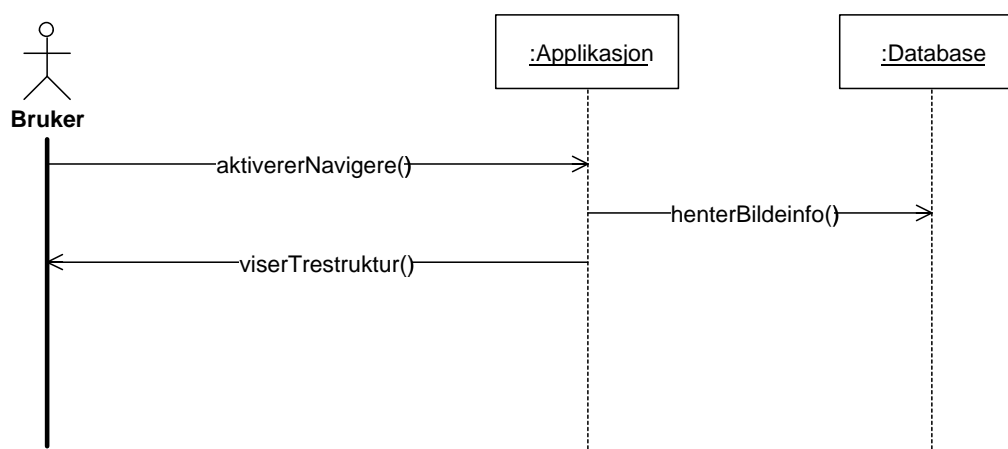
3.4.1.2 Sekvensdiagram og kontrakter for:
”Søkeetterbilde” -versjon1

Kontrakter for *Søkeetterbilde* Usecase

Navn:	leggerInnAttributter
Ansvar:	Ivaretar de attributter som bruker ønsker å søke etter
Type:	Applikasjon
Kryssreferanser:	
Notater:	Denne formen ivaretar de attributter som bruker ønsker å søke etter
Unntak:	Visse felter må være i tråd med forhåndsdefinerte verdier
Output:	Vil vi i denne og neste systemkall få fram oversikt over tidligere lagrede bilder
Pre-tilstand:	Formen er oppgående og applikasjon/server tilsvarende
Post-tilstand:	<i>finnerOversikt</i> blir startet med de verdiene vi har lagt inn

DatabaseforLagringogAnalyseavMikrostrukturbilder

Navn: **finnerOversikt**
Ansva: Sørgerforåskøketterbilderidenfysiskedatabasen
etterdesøkekriterierbrukerenskevinn
leggerInnAttributter
Type: Applikasjon
Kryssreferanser: leggerInnAttributter
Notater: Dennemodulenvilvaretaeattributterbru kerskevinn
ogforetaetsøkmotdatabasenetterdisse
Unntak: Feilattributtergirfeilmelding
Output: Girenoversiktoverdefeltersomsvaretil
søkekriteriene
Pre-tilstand: Verdienesomerblittlagtinni *leggerInnAttributterer*
gyldige
Post-tilstand: Resultatavsøkooverføresoginitierer *viserOversikt*

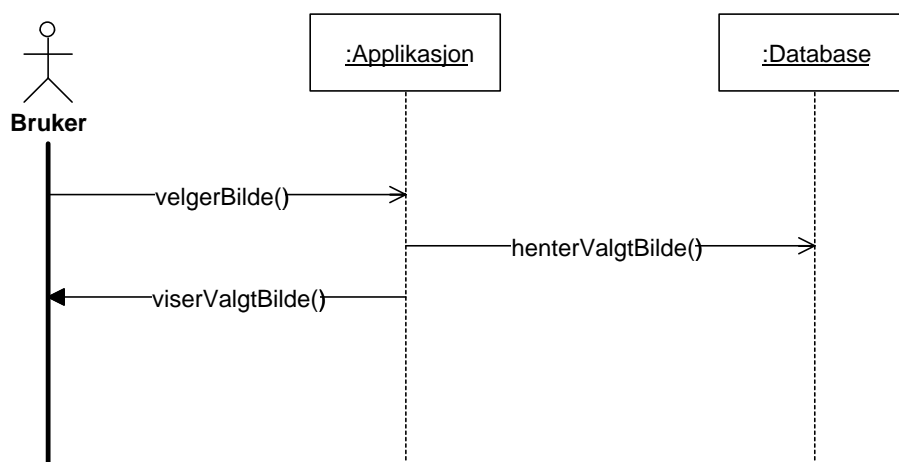
Navn: **viserOversikt**
Ansva: Viserresultateneavsøketterbildeiformaven
tekstrepresentasjonienform
Type: Applikasjon
Kryssreferanser: finnerOversikt
Notater: Resultatavsøkvisesienfasttabelliettbrowservindu
Unntak: Hvisdetikkefinnesnoeunderdesøkskriteriervil
feilmeldingbligitt
Output: Søkeresultater
Pre-tilstand: *finnerOversikterblittfullført*
Post-tilstand: Enformviserres ultatavdesøksomerblittforetatt

3.4.1.3 Sekvensdiagram og kontrakter for:
”Navigasjon”-versjon1

Kontrakter for *Navigasjon* Usecase

Navn:	aktivererNavigere
Ansvar:	Skal ivareta ettsøkbasert på en grafisk trestruktur
Type:	Applikasjon
Kryssreferanser:	
Notater:	forbereder fremvisning av en hierarkisk trestruktur som visertidligere lagrede bilder
Unntak:	Hvis det ikke finnes noen eller avbrudd gis feilmelding
Output:	Trestruktur
Pre-tilstand:	Bruker aktiverer Navigasjon via knappiform
Post-tilstand:	Initierer <i>henterBildeInfo</i>

DatabaseforLagringogAnalyseavMikrostrukturbilder

Navn:	henterBildeInfo
Ansvar:	Henterinfoomdetbildetsomblevalgtunder aktivererNavigere
Type:	Applikasjon
Kryssreferanser:	aktivererNavigere
Notater:	Henterinformasjonogdokumentsomertilknyttetdetbildetvi valgteviatrestrukturen
Unntak:	Vedfeiltrykkellerverdier
Output:	Senderparametretilbrowsersomviserinfoombildet
Pre-tilstand:	aktivererNavigere
Post-tilstand:	Informasjonblirdisplayetivalgtformibrowser
Navn:	viserTrestruktur
Ansvar:	Håndterergrafiskvisningavinformasjonombildetog tilhørendedokument
Type:	Applikasjon
Kryssreferanser:	henterBildeInfo
Notater:	Brukerdataframodulevertilålageengrafisktrekonstruksjon overbilderietdirectory
Unntak:	Hvisparametreerfeil
Output:	Grafisktrestrukturibrowser
Pre-tilstand:	henterBildeInfomåværeinitiert
Post-tilstand:	Enrepresentasjonover dataiformavettredidiagramibrowseren

3.4.1.4 Sekvensdiagram og kontrakter for:
”Visningavbilde” -versjon1:

Kontrakter for *Visningavbilde* Usecase

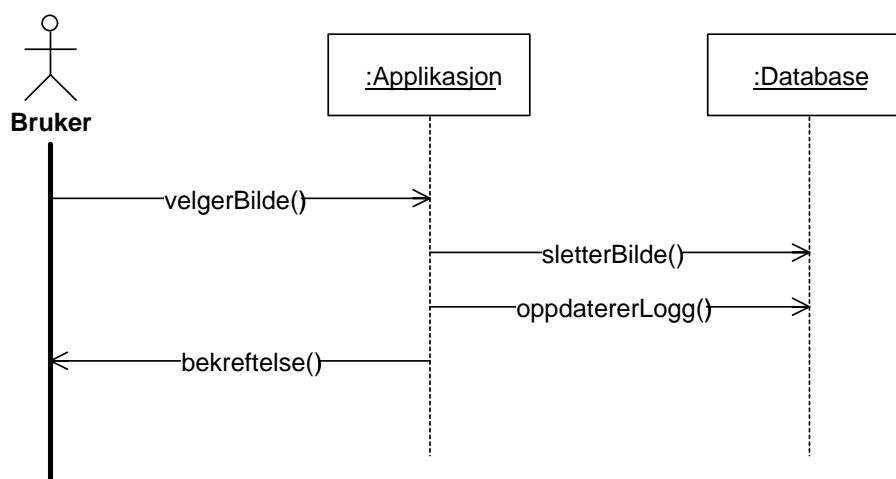
Navn: **velgerBilde**
 Ansvar: Når et bilde skal vises grafisk i browservinduene håndteres dette
 Type: Applikasjon
 Kryssreferanser:
 Notater: Bilde blir vist frem i et eget browservindu
 Unntak: Hvis bildet ikke eksisterer
 Output: Bilde fremvisning
 Pre-tilstand: Et søk eller en navigering gjennomført for å gå til bildet som brukervil vise
 Post-tilstand: Bildet blir vist i et eget vindu

Navn: **henterValgtBilde**
 Ansvar: Foretaren utfører henting av ønsket bilde
 Type: Applikasjon
 Kryssreferanser:
 Notater: Henter ut bildet brukeren har valgt fra den fysiske databasen
 Unntak: Feil under overføring

DatabaseforLagringogAnalyseavMikrostrukturbilder

Output: viserValgtBilde
Pre-tilstand: velgerBilde
Post-tilstand: viserValgtBilde

Navn: **viserValgtBilde**
Ansvar: Viserfremvalgtbildeietegetvinduibrow seren
Type: Applikasjon
Kryssreferanser:
Notater: Denneåpneretegetvinduforsååvisefremønsketbildeidette
Unntak: Hvisdeterfeilvedformat
Output: Bildetibrowser
Pre-tilstand: henterValgtBilde
Post-tilstand: Bildetvisesibrowse rvindu

3.4.1.5 Sekvensdiagram og kontrakter for:
”Slettingavbilde” -versjon1:

Kontrakter for *Slettingavbilde* Usecase

Navn: **velgerBilde**
 Ansvar: Sørgerforåfåmarkertdetbildetsomska Islettes
 Type: Applikasjon
 Kryssreferanser:
 Notater: Ivaretarbrukerensvalgnårdetgjeldervalgavbildesom
 skalslettes
 Unntak: Hvisfeilformatervalgtgisfeilmelding
 Output: sletterBilde
 Pre-tilstand: Slettingavbildeervalgt
 Post-t ilstand: Iverksetter *sletterBilde*medparametrefra *velgerBilde*

Navn: **sletterBilde**
 Ansvar: Besørger slettingavbildeidenfysiskedatabasen
 Type: Applikasjon
 Kryssreferanser: velgerBilde

DatabaseforLagringogAnalyseavMikrostrukturbilder

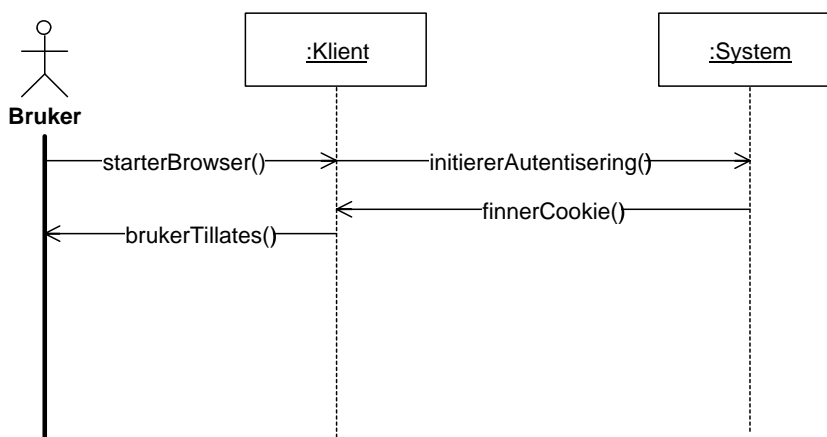
Notater: Dennemodulenskalivaretalet tingutmotdatabasen
Unntak: Bildeterfraværendeellerandredrivermedoperasjonerutmot
sammebildet
Output: Bildetfjernesfradatabase
Pre-tilstand: Valgtbildeterklargjortforsletting
Post-tilstand: Bildetfjernesfradatabaseogforandri ngermedattributter
loggføresvia *oppdatererLogg*

Navn: **oppdatererLogg**
Ansvar: Skalføreoppientekstligrepresentasjonhvilkebilder
somharblittslettetogavhvemognår
Type: Applikasjon
Kryssreferanser:
Notater: Denneskalprodusereenlisteovertidligereeslettedebilder
Unntak: Loggmanglerellerbildetpågrunnavuregelmessigheterved
transaksjonenikkebleslettet
Output: Ajourførtlogg
Pre-tilstand: Loggekspartnerogerinitiertav *sletterBilde*
Post-til stand: Loggblijustertoglagtutinyogrevidertutgave

Navn: **bekreftelse**
Ansvar: Girbrukertilbakemeldingpåomslettingharblittutført
Type: Applikasjon
Kryssreferanser:
Notater: Denneskalgientilbakemeldingtilbr ukerombildethar
Blittslettet
Unntak: Hvisslettingenikkeharblittutført
Output: Enformfordialogboksmedfeedbacktilbruker
Pre-tilstand: *oppdatererLogg*og *sletterBilde*eriverksatt
Post-tilstand: Meldingstilbrukeromatslettinge rutførtellerikke

3.4.1.6 Sekvensdiagram og kontrakter for:

”Autentisering og registrering” -versjon 1:

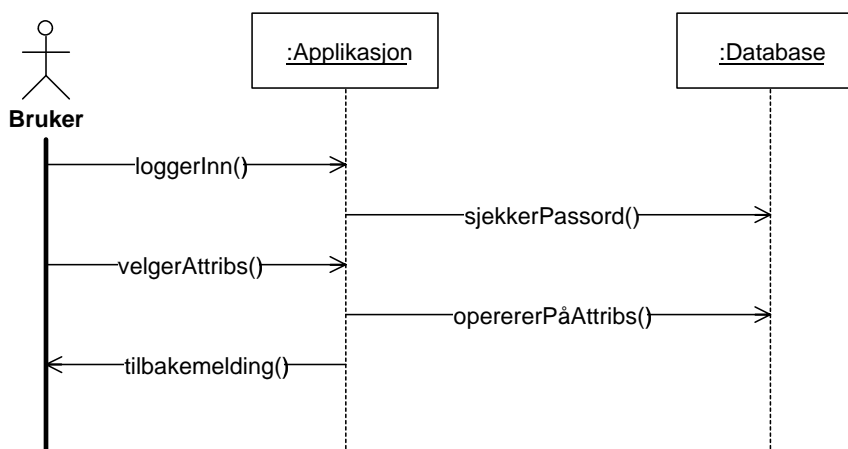


Kontrakter for *Autentisering og reg.avbruker* Usecase

Navn:	starterBrowser
Ansvar:	Initierer på loggingssystemet
Type:	Applikasjon
Kryssreferanser:	
Notater:	Denne tråkkraften når brukeren forsøker å laste ned websiden med de relevante funksjoner
Unntak:	Side finnes ikke eller avbrudd i nett
Output:	Starter initierer autentisering
Pre-tilstand:	Browser er oppgående og riktig adresse blitt tastet inn
Post-tilstand:	Starter <i>initiererAutentisering</i>

DatabaseforLagringogAnalyseavMikrostrukturbilder

Navn:	initiererAutentisering
Ansvar:	Håndtererautentiseringavbruker
Type:	Applikasjon
Kryssreferanser:	starterBrowser
Notater:	Dennemodulenskalsjekkeomenbrukerfinnes registrertfrafør,hvisikkekalnyedataleggesinnog Cookiegenereres
Unntak:	Hvisdetfinnestocookiessomerlikefrafør
Output:	Starter <i>finnerCookie</i>
Pre-tilstand:	starterBrowser
Post-tilstand:	Entenharsystemetfunnetdenbrukersomloggersegpå eksistererfraføriformavencookie,ellersåvildet måttelagesnybasertpåbrukerensdata
Navn:	finnerCookie
Ansvar:	Finnerut omdenbrukersominitiererautentiseringfinnes fraføriformavencookie
Type:	Applikasjon
Kryssreferanser:	
Notater:	Dennesøkeretterbrukerdataiencookiepåklientsom samsvarermedbrukerensprofil
Unntak:	Hviscookieikkefinneselleromde teksistererto identiskepåsammemaskin
Output:	Girmeldingtil <i>initiererAutentiseringomcookieeksisterereller ei</i>
Pre-tilstand:	initiererAutentisering
Post-tilstand:	Parametreomcookieeksistererellerikkeoverførestil initiererAutentisering
Navn:	brukerTillates
Ansvar:	Slipperbrukerinnidatabasennårbrukerregistrert
Type:	Applikasjon
Kryssreferanser:	
Notater:	Nårbrukerentenerfunnetviacookieellerregistrertsomny brukervildennevidereføreb rukereninnidatabasen
Unntak:	Hvisikkebrukerfinnesennå,vedavbrudd
Output:	Brukergistillatsetilvidereaksessidatabasen
Pre-tilstand:	initiererAutentiseringharklargjortbruker
Post-tilstand:	Detgisfulltilgangforbrukertilåbenyt tesegavdatabasen

3.4.1.7 Sekvensdiagram og kontrakter for:
”Brukeradministrasjon”-versjon 1:

Kontrakter for Brukeradministrasjon Usecase

Navn: **loggerInn**
 Ansvar: Ivaretarbrukerdatanår det gjelder innlogging
 Type: Applikasjon
 Kryssreferanser:
 Notater: Dedatasomharblittskrevetinnlagresogivaretas
 Unntak: Hvis ikkedataerkorrekte
 Output: sjekkerPassord
 Pre-tilstand: Klikketbrukeradministrasjonknappen
 Post-tilstand: sjekkerPassord

Navn: **sjekkerPassord**
 Ansvar: Kontrollerer at administratørens passord er korrekt
 Type: Applikasjon
 Kryssreferanser:
 Notater: Kontrollerer om det innstatede passordet og dataene er korrekte
 Unntak: Hvis feilpassord

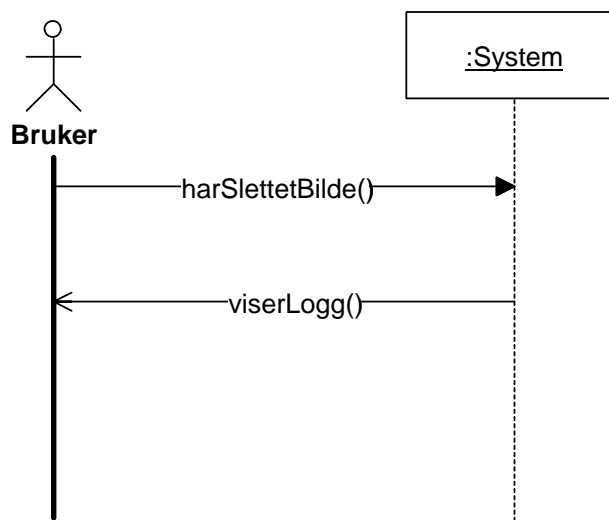
DatabaseforLagringogAnalyseavMikrostrukturbilder

Output: velgerAttribs
Pre-tilstand: loggerInnerinitiert
Post-tilstand: velgerAttribsbliirstartet

Navn: **velgerAttribs**
Ansvar: Ivaretardevalgenadministratorforetar
Type: Applikasjon
Kryssreferanser:
Notater: Skalhåntteredevalgenadmi nistratortarvedrørende funksjoner
Unntak: Hvis *sjekkerPassorder*feil
Output: operererPåAttribs
Pre-tilstand: sjekkerPassordinitiertogkjørt
Post-tilstand: opererePåAttribsverksettes

Navn: **operererPåAttribs**
Ansvar: Hånttereredevalgenadmininstartorønskerå gjennomføre
Type: Applikasjon
Kryssreferanser:
Notater: Devalgenadmintarangåendebrukerfunksjonerutføres
Unntak: Hvisadminvelgerfeil
Output: tilbakemelding
Pre-tilstand: velgerAttribs
Post-tilst and: Tilbakemeldingetteråhautførtdevalgenadministratortar

Navn: **tilbakemelding**
Ansvar: Skalgibrukertilbakemeldingomdefunksjonersomhar blittutført
Type: Applikasjon
Kryssreferanser:
Notater: Girtilbakeme ldingomhvilkeoghvordandeforskjellige tingharblittgjort
Unntak: Hvisfeilunderutførelsenavadministratorvalg
Output: Tilbakemeldingtilbrukeriformavdialogbokslignende melding
Pre-tilstand: opererePåAttribserkjørt
Post-tilstand: Tilbakemeldingerblittgittbruker

3.4.1.8 Sekvensdiagram og kontrakter for:
”Loggføring”-versjon1:

Kontrakter for Loggføring Usecase

Navn:	harSlettetBilde	
Ansvar:	Ivaretar de attributter som gjelder bildets som har blitt slettet	ittslettet
Type:	Applikasjon	
Kryssreferanser:		
Notater:	Denneskal midlertidig lagre den informasjonen som eksisterer om et bilde som har blitt slettet	
Unntak:	Hvis det ikke eksisterer informasjon eller om transaksjonen var ugyltig	
Output:	viserLogg	
Pre-tilstand:	Sletting av bilde modulen har blitt kjørt	

DatabaseforLagringogAnalyseavMikrostrukturbilder

Post-tilstand: viserLogginitiert

Navn: **viserLogg**

Ansvar:Illustrererden tekstlige historiske representasjonen av tidligere slettede bilder

Type: Applikasjon

Kryssreferanser:

Notater: Denne viser fram den historiske oversikten over tidligere slettede bilder som den aktuelle bruker har slettet i basen

Unntak: Hvis.txt fil ikke eksisterer eller transaksjon ikke er komplett

Output: Grafisk visning av slettede bilder

Pre-tilstand: har SlettetBilde

Post-tilstand: Illustrerer debildersom har blitt slettet

4 Begrensninger

4.1 Software design begrensninger

4.1.1 Software standarder og språk

Utgangspunktet for utviklings syklusen vår blir at alle applikasjons logikk skal programmeres i Java. I tråd med det valgte språket ble Java valgt og så bestemt at viskalgå forenjenbruksbasert løsning. Dette vil si at vi vil bruke de komponenter som eksisterer i den grad dette er mulig. Itillegger det også selvfølgelig at det bli en objektorientert utvikling. I og med at dette ble det valgte programmeringsspråket så skjønner vi at dettes systemet ikke skulle by på noen fremtidige problemene når det gjelder å kjøre på ulike operativsystemer. Heller ikke med vekslende databaser. Det vil bli fulgt de regulære standarder som eksisterer idag, som f.eks MySQL og HTML. Noe som også forenkler portabilitet over til andre systemer og plattformer. Angående forklaringer og veiledning så vil dette systemet vårt bli skrevet på norsk. Etter ønske fra IT ansvarlige ved Raufoss ASA og i tråd med gruppens gjennomsyrede gratis-software filosofi, vil det bli brukt Tomcat på webserver.

4.1.2 Software pakker/verktøy/databaser /Operativsystemer

På Raufoss er det kun brukt Windows NT på servere og klienter. Per idag brukes det Oracle til database bruk. Men IT ansvarlige på Raufoss uttrykte at det fremtidens vil bli brukt andre software løsninger. Her under kommer forespørselen om vi ikke kunne bruke Tomcat på webserver. Som nevnt i forrige avsnitt så vil vår Java løsning ikke komme til å skape problemer med implementering på et NTOS. Dette er fullt kompatibel til samarbeidssammensvært bra. Java er også et programmeringsspråk som tetter opp om å kunne legge inn detaljert beskrivelse av hver enkelt komponents funksjon. Dette medfører at nå renny personskalt over og bygge videre på løsningen vår, så vil det være enklere for ham å forstå dets funksjon og oppbygning via detaljerte byggeplaner. JSP vil bli kjørt fra server for å underlette prosessering ute hos hver enkelt klient. Dette da Raufoss ASA er begrenset internt med en linje hastighet på 10 mbit. Itillegget vil bruke Java og JSP vil bli koblingen mellom database og Java applikasjon bli opprettet med JDBC. Dette blir sannsynligvis en form for "pooled" løsning. JSP vil forøvrig generere HTML som vil bli vist i browser ute hos klientene, dette i tråd med underletting av prosessering. MySQL vil bli valgt fortrinnsvis som database program, da denne er gratis og enkle å logge. Det vites at MySQL ikke innehar alle funksjoner som ligger i f.eks Oracle, men det som mangler i denne vil vi prøve å legge ut i applikasjonsdelen. Slik at vi kan kompensere for eventuelle mangler.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Det kan nevnes at selv om Raufoss kjører Windows til daglig, så vil vi bygge opp systemet i Linux miljø.

4.1.3 Toleranser, marginer, muligheter/tilfeller

Programløsningen vil måtte være slik at den i fremtiden kan utvides og implementeres andre moduler. Dette men er vi understøttet at de valg vi har tatt når det gjelder valg av software og utviklingsmiljø.

Når det gjelder diskplass så går sjefen på Raufoss beskjed om at dette ikke er relevant for oss da disk er blitt svært billig. Så implementer pakking eller ekstra funksjoner for å spare plass er egentlig ikke et tema.

Selv om systemet ikke må ta mer enn 20 brukere hvorav 3 samtidig, så kommer vi til å implementere algoritmer som legger ut mye av logikken i RAM. Dette for å øke hastigheten på systemet.

4.2 Hardware design begrensninger

Da it ansvarlig på Raufoss ga oss beskjed om at systemet (webserver) sannsynligvis kommer til å kjøres på en Pentium 3550 mhz med 1,3 gb minne og mer enn nok diskplass, ser vi ikke at hardware kan komme til å legge noen som helst form for begrensninger på systemets funksjonalitet. Samtidig ble det gitt beskjed om at klientene i løpet av vår part kommer til å bli oppgradert til nye stype arbeidsstasjoner. Da de maskinene de allerede hadde der var mer enn nok til å kjøre systemet så kommer heller ikke klientene til å ha noen problemer med systemet.

Angående sikkerhets kopi av billedet i databasen så må dette gjøres manuelt, da dette ikke er understøttet i selve applikasjonslogikken. Det forventes derfor at IT personale opp på Raufoss har gode rutiner når det gjelder slik kopiering av data.

4.3 Bruker design begrensninger

Med tanke på at brukerne på Raufoss ikke har noen formell kompetanse innen fagfeltet data, så vil det bli lagt vekt på at utseendet blir lagd enkelt og intuitivt. Det bør konstrueres på en slik måte at det ikke avviker for mye sett i forhold til den GUI de er vant med fra før. Her under kommer også funksjonalitet og plassering av funksjoner. Selv fargevalg og valg av komponenter bør velges med dette i bakhodet. Bruk av forklarende tekst på skjerm bør velges slik at det er selv forklarende hvordan enskalle gjenbruker data og hvor.

5 Aspekter om kringlivssyklus

Et kriterium fra Raufoss var at koden skal være lagd slik at senere utbygging ikke blir hindret. Dette kom mer vitilåprøve å ivareta ved å dokumentere kildeteksten godt. Det vil heller ikke bli lagt noen begrensning for dette via vårt valg av software løsninger og programmeringstekniske kruspring. Den løsningsomskallages bør ved å tdenne gjenbruksbaserte ikke by på problemer ved rørende utbygging.

5.1 Dokumentasjon

Guimålages slik at det bli enkelt og selv forklarende å bruke. Dette vil da medføre at det neppe skulle trenge så mange tekniske dokumenter som brukermanual. Det vil ved inntalling på Raufoss bli gitt full informasjon om system, slik at brukere umiddelbart kan ta dette i bruk. Men for sikkerhetsskyld vil det bli laget tekstfiler som vil bli lagt ved CD.

5.2 Modul - og integrasjonstesting

Under hele utviklingsprosessen vil hver enkelt modul bli testet under varierende bruk og med varierende input. Samtidig vil komponentenes å fortdeerferdige bli testet i samkjøring og under forhold som gjenspeiler den arbeidsbelastning og variabler de vil bli utsatt for hos Raufoss. Ferdig applikasjon må testes med ulovlige verdier i alle innfyllingsfelter.

5.3 Konfigurasjons - og versjonsstyring

Når det vil bli gjennomført små forandringer ved systemet, så bør systemet oppdateres med en økning på 0.N på versjonsnummer. Hvis forandringen genererer en slik art systemets brukerfunksjoner vil bli om lagt eller større forandringer vil bli gjort så bør versjonsnummer reflektere dette via en økning på første tall N.0. Slik at en mindre justering vil gi en økning fra versjonsnummer 1.1 til 1.2. Større omlegginger vil gi en økning fra f.eks 1.1 til 2.0.

5.4 Kravtilsupport, service og vedlikehold

Det vil under installering bli gitt omfattende beskjed om hvordan systemet skal brukes. På medfølgende CD vil det også bli lagt inn beskrivelse av bruksgangen. Ellers kommer ikke dette systemet til å bli av en slik art at det trenger noen omfattende support. Det vil bli en enkel og intuitiv løsning lagd for alle brukere.

Database for Lagring og Analyse av Mikrostrukturbilder

I Java-kodene vil det bli skrevet innomfattende og detaljerte beskrivelser av handlingssforløpet, så gjennbruket ikke byr på uoverkommelige problemer. Det kan nevnes at gruppe-medlemmer i tillegg vil være åpne for tekniske spørsmål og henvendelser fra brukerne.

5.5 Krav til utvidelser

Da systemet vil bli basert på gjennbrukt så skal fremtidige utvidelser ikke byr på problemer. Dette er et viktig kriterium fra Raufoss sine sider de hele tiden utvikling og legger om og justerer sitt system. Valg av software må heller ikke legge noen restriksjoner på utvidelsesmuligheter.

6Aspekterominstallasjon

6.1Hardwareinstallasjon

Hardwareer allerede oppgående hos Raufoss. Som nevnt tidligere vil dette starten bestå av P3550 mhz med 1,3 gbr og mer enn nok diskplass. Sannsynligvis vil dette også bli forandret i fremtiden.

6.2Overgang/omlegging

Det vil ikke komme til å bli noen omlegging da de i dag ikke har noe system i det hele tatt. Derimot vil det kunne bli en positiv overgang for brukerne å kunne legge bildene sine på et lagringssted i stedet for å spre dem over mange forskjellige lokaliteter.

6.3Opplæring

Det er ønskelig at det lages en manual i form av et tekstdokument på CD som skal leveres av Raufoss. Denne vil inneholde det en brukertrenger for å benytte seg av systemet. Den bør skrives i f.eks. word slik at den kan printes ut senere.

Ved installering vil det også være ønskelig med et innføringskurs for brukerne.

7 Akseptansekrav

Hastigheterogresponstiderisystemetmåholdespåetsliktni våatbrukerikkeføler atdetgårunødvendigmyetidtilventing. Herunderdemaksimumtidsernevent tidligeikravspesifikasjonen. Søkefeltermåakleforskjelligekombinasjonerav tegnoppsett, og gifeilmedinghvissøkerbenyttersegavulovlige kommandoer. Hvishanprøverålagreulovligefilformaternårdetgjelderbildet, såmåsystemgi beskjedomatdetteikkegårighvorfor.

8 Prosjektstyring, inkludert kvalitetssikring

Organiseringinnadigrupperdefinertsomtidligeikravspesifi kasjonen. Prosjektplanenerførsteguidentilgruppennårdetgjelderhvilkenvegprosjektet skalta. Denneerfundamentetfordennekravspesifikasjonenitilleggtildebruker kravsomertilkommetiettertidd. DavalgteutviklingsprosesserRU Pvindettetilsiatvihardeltinn prosjektutviklingenitrefaser. Hvorviihverfaseviltaforossenkeltavusecasene. Deusecasesomharhøyestrangeringifølgetabellenvårvilblitattførst. Slikatusecasenr1og2vilbliutvikletif ase1, ogusecase3og4ifase2. Dettegirossenklarformeningomhvoriløypaviliggersettiforholdtil fremdriftsplanen. Noesomforenklerprosjektoppfølgingen. Nårdetgjelderkvalitetsstyringsåvildetunderogetterhverutviklingssyklus verifiseresomprosjektetgåridenretningviønskerogidetteempoviharsattopp. Testingutføreskontinuerligogmedvarierendebelastningogforholdrundt.

KAPITTEL3DESIGN

3.DESIGN

3.1Introduksjon

Detrefereresistorgrad tilkravspesifikasjonsdokumentetforimplementasjons - spesifikkedetaljerveddesign.

3.2Målogobjektiver

Detoverordnetemåletmedsoftware -arkitekturenerfordetførsteetentogoversiktlig design.Dermedkommerplattformuavhengigetsometsidesti lltmål.

3.3Softwareomgivelser

Hoveddeleneavsoftware -miljøetkanbeskrivesetteren3 -lagsløsning,Detførstelaget erpresentasjonslagetiformavstandardwebklienter.Disseblirbrukttilren presentasjon/gui -grensesnittilrestenavapplika sjonen.Detandrelaget,mellomlaget, eridennesammenhengbruksomenplattformforapplikasjonslogikken.Idenne sammenhengerdenneplattformenbasertpåenweb -serversomkantjenehtml -sider ogkjørejava -kode.

3.4Hovedrestriksjoner/begrensninger

Detvariutgangspunktetsterktønskefraarbeidsgiveromågjørearkitekturenså enkelogliteressurskrevenesommuligutehosklientene.

Detteførtetilvissebegrensingerihvilkekravsomskullestillestildeulikesoftware - lagene.Presentation s-lagetskulleværeenkeltoguniverselt.Meddettemenesatdet skullestillesenklekravtilsoftwarepåklientsiden.Itilleggvaretetkravatklientene skulleutføreminstmuligprosseseringavdata.Proseseringenblederforflyttetover påapplik asjons-serverenogderetterdatabase -serveren.

3.5Datadesign

Demestfremtredendedata -strukturererHashMaps.HashMapserenformfor tonøkkelarrayforålagrekeyogvalue.Dennevisteseåværemegetnyttigforå lagreenkleparamaterefraklient -siden.Denandreformenfordata -strukturerde forskjelligdata -streamenesomertilgjengeligjava -biblioteket.Disse

Databasefor Lagring og Analyse av Mikrostrukturbilder

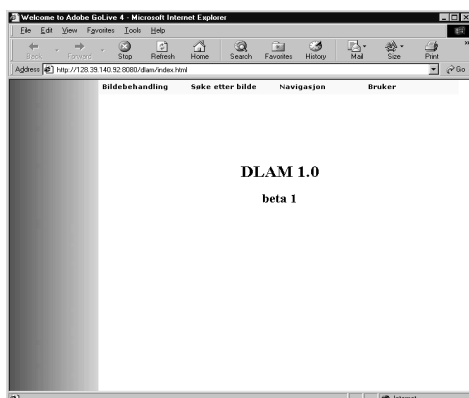
datastrukturene er tilpasset for å lese og skrive ulike typer data, og kunne utveksles mellom objekter, noe som underletter implementasjonens detaljer.

3.6 Beskrivelse av arkitektur og komponenter

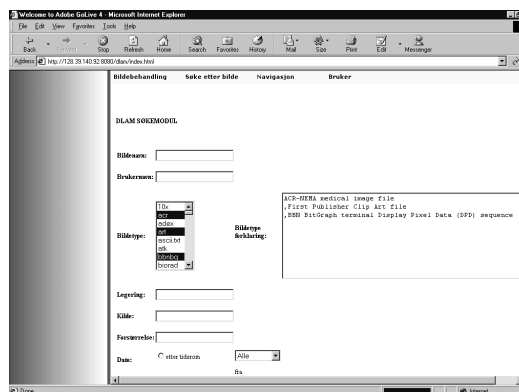
Programstrukturen er i stor grad basert på bruk av objekter og meldinger. Dette innebærer derfor forskjellige stadier for overføring av data. I tillegg er den logikken for klient-kommunikasjon og database-kommunikasjon skilt ut i egne klasser for ryddigere meldings-håndtering.

3.7 Brukergrensesnitt for design

Brukergrensesnittet er designet kan sees ved å starte index.html-siden fra root-katalogen for applikasjonen. Herfra vil lagrings- og registrerings-sidene være mest aktuelle å se på. Et hovedkrav i grensesnittet er å være enkelt og design som mulig slik at brukernes skjelens skulle bli lav.



Figuravåpningsskjerm



Figuravlagringsskjerm

3.8 Begrensninger og restriksjoner

Begrensninger i utviklingen lå først og fremst på bruken av jsp/java-server-pages på applikasjonsserveren. Etter noen tids testing fant vi ut at denne logikken ville bli for trege til vårt bruk. Vi gikk derfor bort fra denne og erstattet den med javascript på klientene. Javascript viste seg i stor grad å fungere tilfredsstillende. Eneste minuset ved bruken av dette var forskjeller i hvordan ulike browserer støttet opp mot despråkspesifikke detaljer som ble brukt.

4.IMPLEMENTERING

Ia nvendelseavverktøyogutviklingsmiljøervardetførstdiskusjonavvalgav plattformstype.DavitrengteenstabilogeffektivplattformfalltvalgetpåLinux. Linuxeretunix -kompatibeltoperativsystemsomerkjentforsinstabilitetog effektivitet. Itilleggtildettevargruppensutvikleregodtkjentmeddenneplattformen fratidligereutviklings -oppgaver.Denneplattformengjordeetogsåenklerefor gruppenåinstallereogkonfigurereenplattformsåhurtigsommulig.Enannetpluss vedvalgetav denneutviklingsplattformenvaratLinuxergratistilgjengeligfra internett.Detteiformavmangeulikedistribusjoner.Valgavtypelinux -distribusjon falltpådenmestdominerendeavde,RedHatLinux.

Nesteskrittvarvalgetavverktøyforåskriv einnkode.Hervargruppenmestvant tilGUI -lignendeutviklingsmiljøer.DissegårunderbetegnelsenIDE -miljøer.IDEstår herfor"IntegratedDevelopmentEnvironment".Ideenmeddisseeråintegrererå mangeulikeaspektervedselveutviklingsgangensommulig.Detsomvarufordelaktig meddettevalgetvarimidlertiddehøyekostnadeneforbundetmedetsåomfattende utviklingsmiljø.Detblederforbestemtåvelgeenkleogmeruniverselle utviklingsverktøyforågjørebrukenavdisseenklereogmindreomfatt ende.Valget falltpåeditorenemacsfraengruppesomheterFreeSoftwareFoundation.

Emacsereneditorsomermegetkjentiunix -miljøerogblirbetraktetsomenavde bedreditorenesomerlagd.Fordelerveddenneeditorenerdemangekraftigeog enklemakroenesomertilgjengeligfrastaturet.

Etviktigvalgvedutviklings -gangenvarvalgavjava -bibliotekerogkompilator.Siden javaeretspråkhorutviklingenforegårietåpentmiljø,finnesdetmange implementasjoneravden grunnlegendejavaAPI (ApplicationProgramming Interface).Daetstabilogrobustbibliotekvarenviktigfaktorvalgteviåtaibruk referansebiblioteketfraSunMicrosystems.Detteeroriginalbiblioteketforjavasom determestutbredtbrukav.Valgetavdettebibliotek etvistesegåværeenbrøløsning dadetfungertetilfredsstillendegjennomheleutvilingsforløpet.Valgetpåkompilator falltiutgangspunktetpåreferansecompilatorenfraSun.Dennevistesegderimot etterhvertåbrukelangtidpåstørrebløkkeravkode e.Vibegynntederforåsetteren raskerekompilatorsomvarstandard -kompatibel.Herfantvifremtilenkompilator vednavnJikes.DenneerenreferansecompilatorfraIBM(InternationBussiness Mashines)somerkjentforsinhurtigekompilering/overs ettelseavjavakode.

Gjeldendepinsipperiforbindelsemedkodingavjava -kodevarstandardretningslinjer fraSunMicrosystems.Disseretningslinjenegikkiveldigstorggradpåriktigbrukav java-bibliotekene.Idissebibliotekene finnesdetmangemet oderogprosedyrer som eksistererkunpågrunnnavbakoverkompabilitet.Dissemetodeneblirmanderimot ikkeoppfordrettilåtaibrukdadevilblifjernetifremtidigeversjoneravjava -

DatabaseforLagringogAnalyseavMikrostrukturbilder

bibliotekene eller erstattet med nye og bedre implementasjoner. Iva Igavdatabase -
verktøy var det viktig for utviklingen å velge et verktøy som ikke var for stort. I tillegg
var det et hovedpunkt å basere databasen på en velutprøvd og robust plattform. Valget
falt derfor på databasene MySQL. MySQL er en database-løsning som er
open source-lisensiert og skrevet av MySQL AB som er et selskap. Denne
plattformen var lett å konfigurere og hadde god dokumentasjon.

5. TESTING

Da prosjektet baserte seg på en evolusjonær utviklingsprosess var testingsentralt i
utvikling av applikasjonen. I utgangspunktet var det tiltenkt å utføre all testing via
prosjekt pc'en som var utleidd av Raufoss ASA. Dette fungerte meget tilfredsstillende i
første del av utviklingsperioden, for å bekrefte grunnleggende funksjonalitet. Denne
testingen ble utført via skolens eget nettverk. Fordeler med dette var at
nettverksmiljøet var grunnleggende det samme som Raufoss ASA hadde sitt
intranett. Med dette menes at nettverket var et svi sjet -ethernetnett på 10 mbit.
Etter hvert vokste det i midlertid frem et behov for å utføre koding ->testing->feilretting
syklusen i en hurtigere takt. Dette på grunn av det etter hvert voksende testtidene som
ble opplevd da applikasjonen ble testet i ytterpunktet for å se om det var robust. Det
ble derfor bestemt å sette opp en eksakt kopi av software -miljøet på en av
gruppemedlemmernes hjemmepc'er. Fordelen med dette var mange. Fordet første var
detten nye nettverket basert på et 100 mbit net. Dette ville teoretisk la overføringene gå
10 ganger raskere.
I tillegg hadde den nye maskinen 512 MBRAM/minne, noe som bedredupliserte
systemet som applikasjonen til slutt var tiltenkt for, som var 1500 mbram. Dette
haddemye å si for overføringstiden ved testing av store datamengder.

5.1 PLANLEGGING

Fra starten av prosjektet var det prioritert å få utstyret som det kunne testes med. I
utgangspunktet hadde vi tenkt å bruke gruppemedlemmernes egne pc'er for dette. Da
dette viste seg å bli alt for vanskelig å koordinere, gikk vi inn for å teste på skolens
utstyr. Etter en forespørsel til skolens IT-tjeneste fikk vi låne en Intel Pentium 75 mhz
maskin. Denne ble da satt opp med riktig operativsystem -plattform og
utviklingsmiljø. Etter testperioden på den nye maskinen ble det derimot oppdaget
svakheter i hvor rask den kunne prosessere data. Det ble derfor tatt valg om å få en
annen testplattform. Vi tok derfor kontakt med oppdragsgiver, Raufoss ASA, og kom
menen henvisning om testplattform til utviklings -formål. 2-3 dagers senere kunne vi
avhente en PII maskin 600 mhz, 256 mbram. Etter konfigurering av den nye maskinen
fant vi ut at innledende utvikling og testing kunne begynne.

5.2 TESTSTRATEGIER

Strategier for testing var at det skulle utføres tett knyttet opp mot selve utviklingen.
Dette innebar å teste ofte, noe som i utviklings -miljøet blir sett på som en fordel.
Snarlig testinger med på å avdekke mest mulig feil slik at de kan bli rettet for test

DatabaseforLagringogAnalyseavMikrostrukturbilder

mulig. Metoden for å utføre denne snarligetestingen var å skrive et maksimum på 100 linjer kode i forskjellige filer før neste uttesting fant sted. Denne metoden er med på å avdekke feil tidlig slik at de kan bli fikset tidlig i utviklings -forløpet.

5.3 GJENNOMFØRING

I de innledende testene ble det oppdaget mange små feil. Disse var grunnet enten feil i java-logikken (som oftest), eller feil ved database -kallo database -oppbygning. Testene ble utført flere ganger under ulike systemforhold. Siden utviklings -miljøet vårt var basert på linux var det litt i bruk av kommando -baserte verktøy i testingen. Dette kontra GUI -verktøysomen måtte ha tatt i bruk under tilsvarende windows miljø. Fordelen med dette var mange, blant annet at den med letthet kunne overvåke ulike deler av applikasjonens oppførsel ved å skrive test og feilmeldinger direkte til konsollet.

I de senere testperiodene hadde verktøyene blitt godt innarbeidet, og bruken av disse gikk nå lettere enn i innledende faser. I senere testperioder hadde derimot koden mengden økt betraktelig og dermed også designet. Dette førte til en grundigere gjennomføring av basistestene for å finne feil som dukket opp. Et annet moment lå i den økende kompleksiteten i feilmeldingenes omblemot at etter de innledende testene. Disse tok i økende grad mer tid å fikse.

Avsluttende test -faser ble påbegynt med den grunnligste testingens så langt. Basis testene gikk i den testfasen for seg uten mye styr, og de feilsomble oppdaget var betydelig færre enn før. Allikevel kan en si at den testfasen var vel så omfattende som de tidlige retest -faser, da kompleksiteten ved oppdaget feil, selv om de var betraktelig færre, hadde blitt mer avanserte. På fagspråket ville en kalt disse feilene for "bugs". "Bugs" er dannelsen av feil ved applikasjonens som nesten unngår utviklere da de er veldig spesielle og dukker opp bare ved underlige omstendigheter. Noen av de feilsomble oppdaget i denne fasen kan velegentlig ikke kategoriseres som feil, men mer som mannlige funksjonalitetens endringer som i utgangspunktet var ønsket.

5.4 OPPDAGEDE FEIL

Innledende testfaser:

I den testfasen ble de mest grunnleggende feilene oppdaget. Eksempel på dette var at innholdet i databasen ikke ble oppdatert da vi trodde det ble utført riktige operasjoner fra applikasjonens innside. Disse feilene viste seg først ved undersøkelse av hva som egentlig var registrert i databasen via manuelle metoder. Disse manuelle metodene baserte seg på bruk av kommando -linje ned i database -klienter. I java delene var det mange små feil med metoder for sjekksum -kalkuleringer og database-overføringer. Testene avslørte mange egenskaper ved java -språket som divergerer fra språket i den samme familien. Disse baserte seg i stor grad på den store graden av threads i java serverlets. Feilene av den typen kreves spesielt i implementasjon av metoder for å løses. Til slutt ble ledet ved å implementere en single thread modell for oppdatering av databasen.

Mellomliggendetestfaser:

Idemellomliggendetestfaserbleblearbeidetmedutvklingogtestingmer rutinepreget. Dette vil si at mange feil dukket opp flere ganger etter hvert om implementasjonen ble forandret. Disse feilene var databedre for stått, da vi hatt sett de før. Derfor var behandlingstiden for disse betydelig kortere enn før. Eksempel på feil som dukket opp i denne testfasen var databasefeil ved tilknytning, og java - implementasjons-spesifikke feil. Disse feilene var nå lettere å ta hånd om da miljøet hadde blitt mer kjent.

Avsluttendetestfaser:

Idette stadiet hadde de fleste kjente feil blitt avklart. Det var nå tid for å avdekke de mer obskure feil med hensyn til integritet og validering. Disse var feil som ikke var synlige i tidligere faser. En litt overraskende egenskap ved disse feilene var at når de først ble oppdaget var de betydelig vanskeligere å behandle. En stor del av tiden ble derfor brukt til å finne løsninger på disse problemene.

6. DISKUSJON AV RESULTATER

I startfasen av prosjektet var det nokså enkle ønskersomble framsatt. Dette fremkommer også av kravspesifikasjonen i form av de grunnleggende kravene til den ønskede applikasjonen.

Av den grunnleggende funksjonaliteten i kravspekmen er vi etter vurdering av endt prosjektå hakomet i havn med tette opp til 100% av disse. Denne grunnleggende funksjonaliteten går da ut på det å kunne lagre, hente fra og sette poster i databasen. Disse har blitt testet mest og ser ut til å fungere opp mot kravspesifikasjonen. Av disse grunnleggende funksjonene er det alle likevell lagringsfunksjonene som fremtrer som de mest avanserte og mest vellykkede. Sett i ettertid var det også denne funksjonaliteten som var mest etter spurt, men også vanskelig å implementere!

KAPITTEL7KONKLUSJON

Brukerne fra Raufoss ønsket seg en mulighet til å ha et sentralt database system som var tilgjengelig fra alle klientene inne på det lokale intranettet. Dette utpekte seg selv som et tilgjengelig til å være en klient-server løsning, og med at det skulle være en sentral ansamling av data. Dette tilsa at det ville bli et system hvor vi hadde en webserver med database og en applikasjonsserver med tilhørende kommunikasjonsut mot områdes klienter.

Et av kriteriene ved oppgaven var at systemet skulle være uavhengig av hvilken plattform det ble kjørt på og at det skulle være bygd opp av standardiserte språk og dermed software. Nødvendigvis måtte dette bety at interfacet ville måtte kjøre si klientenes browsere. Dette rammeverk tilsa at språket til applikasjonen måtte bli java, databasen en form for gratis tilgjengelig software og html ut mot klienten måtte genereres via jsp. Jsp fordi vi da sparte klientene for en nødvendig prosessering. Java ble valgt fordi det er et språk som er rypere og lettere å sette opp gaver. Det er et globalt programmeringsspråk som det finnes utallige gjenbruksbaserte komponenter for. Jsp utpekte seg automatisk som det beste språket i IT avdelingen på Raufoss hadde kompetanse innen. Gruppen ville personlig gå til foren løsning av php, da vi har mer interesse og kunnskaper innen dette. Men det ble bestemt at det skulle tas hensyn til Raufoss sine ønsker.

Når det gjaldt valg av database så falt det naturlig at vi skulle gå for en fritt distribuerbar base. Raufoss bruker fra før Oracle og kunne forsyne oss med lisensiert versjon av denne. Dette ville da ikke gruppen benytte seg av, da vi tilhenger av fritt distribuerbar software. Valget vårt falt da heller på MySQL databasen. Denne er da fritt tilgjengelig for alle som er interesserte ut på nettet. Og med at vi gikk for denne så gikk vi en viss skjebeimøte. Dette fordi MySQL er en ganske enkel base. Mange av funksjonene som finnes i kommersielle databaser eksisterer ikke i denne, og må derfor programmeres rundt via logikk som ligger i applikasjonen. Dette medførte en ny fare, da Raufoss ønsket seg en applikasjon som kunne brukes på forskjellige databaser. Fordi hvis logikk mangler i databasen og dette må legges inn i applikasjon, så vil dette føre til at applikasjonen når den konverteres ikke vil bli kompatibel med den nye databasen. Slik at det er en fordel om den logikken støttes i databasen. Men dette gjordedaaltså ikke MySQL, slik at vi måtte implementere logikken i java applikasjonen. Den logikken utførte vi da på en slik måte at det ikke bør være noe særlig stort bryderi å kjøre den med andre databaser allikevel. Det var altså en bevisst rekke av valg som ble tatt under kodingsfasen.

e

DatabaseforLagringogAnalyseavMikrostrukturbilder

Dadettesystemutviklingsforløpetbleinitiærtviakravspesifikasjonen, blesystemet somenhetetoppdeltiåttmoduler(usecase).Disse dannet grunnlaget for all videre utvikling og var opererå betraktesomdeendeligemålforsystemet. Altså kalte suksessfaktorer.

Disse åtte usecase var følgende moduler, med på følgende prioritet:

*Lagrenyttbilde	Primært
*Søkeetterbilde	Primært
*Navigasjon	Primært
*Visningavbilde	Primært
*Slettingavbilde	Primært
*Autentiseringogregistrering	Sekundært
*Brukeradministrasjon	Tilleggsfunksjon
*Loggføring	Tilleggsfunksjon

Etter usecase navn står det primært, sekundært eller tilleggsfunksjon. Dette forteller oss om de var krav til systemet, om de var ønskelige eller bare tilleggsfunksjoner som kunne programmeres om gruppen fikk tid til overs. Ifremdriftsplanen til gruppen ble RUP prosessen inndelt i tre utviklings sykluser. I disse satte vi opp de 2 første primære usecase i syklus 1, den neste 2 i syklus 2 og den siste primære pluss resterende sekundære og tilleggsfunksjoner i syklus 3. Vi begynte prosjektet med å bygge opp og konstruere databasen i MySQL. Da denne var opppe gåendeså begynte vi på applikasjons biten. Her ble modul "Lagrenyttbilde" og "Visningavbilde" de første som ble programmert.

Dette var forøvrig blant de som tok mest tid. Her måtte vi forstå den logikken som måtte implementeres i applikasjonen for å kunne legge inn og ta ut bilder. Dette valgte vi å gjøre på den måten at vi fjernet header på bildene, og la inn dem i selve den fysiske databasen som binære filer. I likhet med all annen informasjon som kan legges inn i en relasjonsdatabase. Dette førte selvfølgelig til at databasen ble stor, men diskplass var ikke et henseende til å etterdets omble fortalt oss av IT ansvarlig på Raufoss. Dette fordi disk i dag har blitt såpass billige innkjøp.

En slik innlegging av bilder fungerte svært bra og ger å anbefale tilsvarende problemer. "Visningavbilde" programmert i vislikat bildet ble vist i et get browser vindu når det ble aktivert via en standard button.

I syklus to utviklet vi "Slettingavbilde" og "Søkingavbilde".

Slettingavbilde ble en form for "reversed engineering" av lagrenyttbilde.

Her var det å se i et gammelt innlagte bildet via applikasjonen. Dette ble programmert inn og ble fungerende bra. Søkingavbilde ble lagd via standard spørringer ut mot databasen ved hjelp av applikasjonslogikken.

Del 3 av utviklingsstadiene inneholdt først og fremst å få laget "Navigasjon".

DatabaseforLagringogAnalyseavMikrostrukturbilder

Herbruktevidelvisgjennbruktaveksisterendekomponenterutepånett,delvislagde vitilleggselv.Denneblealtsåfungerendepådetvisatdenviseren”explorer lignende”grafiskillustrasjonavfileneidatabasen. Nåhaddevilagda lledekomponentersommåtteværemedogdetvaropptilosså bestemmehvilkesomskulleblimedavdeønskelige. Valgetfaltpå”Autentiseringogregistrering”og”Loggføring”.Dissetovurdertevitil åværedesomvarmestrelevanteforbrukogsikkerh etforsystemet. Autentiseringogregistreringbleenfunksjonsomskalkontrollerehvembrukerer,for sååjobbeoppmotdatabasenforåautentisereatbrukerriktigperson. Dettevilsiatførstegangenbrukerloggersegpåmåhanskriveinnnavn,b rukernavn ogpassord.Dettevilsåblilagreticookiepådennehjemmeområde. Dennesegangbrukergårinnpåhjemmeside,ogforøvrigallekommendeganger,så vilikkehanmerkenoeninnloggingsrutine.Menhanslipperrettinnpåwebsidensom omdetvar enheltvanligside.Menibakgrunnenhervildetforegåenlesingav identifikasjonentilbruker.Detteførertilatnåenbrukerprøveråletteenfilsåvil applikasjonsjekkecookiesdataoppmothvemsomharlagretidenfilahanprøverå slette.På denmåtenfårkuneieravetbildelovtilåsletteiden,noesomvaretkriteri fraRaufossinside.Samtidigfårallelovtilåsepåalleandresbilder. ”Loggføring”blelagdvedålasystemetskrivedatauttilenfilomhvemsomsletter hvaognår.På dennemåtenerdetmuligheterforåsesinhistorieoghvaennhargjort. Etterdennemodulenvarimplementertsåhaddeprosjektetkommetsålangtatviikke syntes”Brukeradministrasjon”varrelevant.Dettevarogsåenmodulsomblelagtpå somenformf ortidsfordrivisistemøtetmedRaufoss.Menvinedprioritertedadenne ogtokhelleretvalgomåbrukemertidogressurserpådeandrekomponentene.Da dissevaravestørreverdiforatsystemetskulleblibra.

Gjennomdetteutviklingsforløpetinns åviatdetfinnesmangefordelermedåbruke kommersiellsoftware.HvisentenkerpåMySQLoppmotOraclesåerdetherstor differanseidenlogikkosomliggiserelvedatabasene.Mendeter,somvifantut,ikke umuligåbrukeenkleløsningerallikevel. Dettedaenkanflyttelogikkentildatabasen overtilapplikasjonen.Somnevnttidligeremedførerdetteenfareforatdetvilbli toleranseproblemer nårdetskaloverføres,menselvdettedetmuligåtahensyntil nårkodingsfasengjennomføres.Dete romågjøreåbrukemestmuligavdende standardersomeksistererutepåmarkedet.Noesomvilsiatenbørholdesegtilden generellesyntaksenistedetforågåpåkommandoersomkunerakseptertpåett produkt.DenkombinasjonvibrukteavMySQL,jav aogjspfungerteegentligmeget braogviseratenikketrengerålisensiereidyredommerforåværeproduktiv.

Iogmedatvigikkforengjenbruksbasertløsning,itrådmedjavafilosofien,såharvi benyttetossavendeleksisterendekomponenter.De tteharmedførtatviredusererden tidsomgårmedtilselvutvikling,menøkerabsorberingavusikkerhet.Fordide komponentervivalgteutsannsynligvisharenvissfeilrate,noesommedførerativi drardennefeileninnivårtprogram.Detteharvidapr øvdåkompensereforvedå testeutjevnliggjennomheleforløpet. Etannetusikkerhetsmomentvilværeatkomponenterhverforsegfungerersværtbra, mendadesettessammenfårvineavdetmotsatteavensynergieffekt.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Det vil si at de separate komponenter fungerer svært bra, mens sammen begynner de å utvikle feil tilstander i systemet.

Via tåke kan det være en mulighet for å unngå dette ved testing. Så alle komponenter har blitt testet separat og sammen gjennom hele utviklingsforløpet.

En av flere innovative vinklinger som har blitt tatt under denne oppgaven er å legge bilder inn i databasen MySQL som en binær fil. Da vil etterløsning på dette utepå internet og i manualer så fant vi ut at det egentlig ikke fantes noe om dette.

Det stod nevnt noen få linjer i manualen til MySQL, men ellers var det ganske dårlig nevnt. Så vi måtte prøve oss fram her. Noe av det vanskeligste ved dette var spørringer som fjernet de tegn som ikke ble godtatt av basen.

Det måtte før innlegging i databasen tashensynt til at header ikke kunne følges med inne i basen. Dette førte til behov for å fjernes spesifikk tegn som MySQL ikke aksepterte. En løsning på dette ble funnet etter hvert, og systemet ble fungerende svært bra.

Gruppene i den oppfatning at det systemet vi har bygd opp, kan brukes andre steder der det er behov for et tilsvarende lagringsstruktur. Uansett om det skulle være et behov for lagring av dokumenter, film, bilder eller annet.

Det er basert på å gjøre bruk og kan utvides og forandres etter ønske. I tillegg til at det er god dokumentasjon og med at kodene er godt kommentert vil det være fullt mulig å sette seg inn i virkemåten.

Da det ikke er lagd noen begrensninger i systemet kan Raufoss i fremtiden ekspandere systemet, eller modifisere det til å kunne omfatte flere funksjoner innen på et nett.

Faglig sett så har vel dette prosjektet vært en av dem mest lærerike delene av våre ved HIG. Forelesninger og studier er vel og bra for å tilegne seg basiskunnskaper, men når en kommer ut i det virkelige arbeidslivets såfå er det helt annet syn på det.

Du må forholde deg til dynamiske problemstillinger og forskjellige virkelighetsoppfatninger. En ser at et problem neppe har ett fasitsvar, men løsningsmetodikk varierer fra situasjon til situasjon.

En fordel ved dette er at en får brukte kunnskaper fra mange forskjellige fag i en situasjon. Dette gjør tilfellet mye mer utfordrende og stiller større krav til hver enkelt individ for å kunne innhente tilleggs kunnskap.

Vi valgte under den nye utviklingen å legge opp en lagplan og så prøve å holde den hele veien. Mange av det tidsrammervis at de oppble trampet på fullstendig, mens andre ble overholdt med god margin. Allikevel så er vi positivt overrasket over at såpass mange av tidsrammen holdt. Dette fordi vi ikke har vært i en slik situasjon hvor vi har måttet analysere tidsforbruk før.

Det kan også si at gruppen faglig sett har fått større forståelse for de forskjellige fagfelt på den måten at vi ser hvor de griper inn i hverandre og overlapper visse kunnskapsområder.

Vi har også fått større forståelse for den viktigheten og den tiden kravspesifisering tar å gjennomføre. Dette skjønnervin å er noe som er helt fundamentalt for at en slik utviklingskallykkes. Uten en slik fundament harkodingsfasen er det tendens til å kunne skli ut i forskjellige retninger.

Koordineringsfunksjonen i innad i gruppen har også oppnådd større status.

DatabaseforLagringogAnalyseavMikrostrukturbilder

Detbleetterhvertydeligatdetåsamkjøreindividuelearbeidsinnsatsene,slikat alle”drarisammeretning”ersværtviktig.Ellerskanenblisittendemedtrepersoner som harprogrammertividtforskjelligeretninger.

Noesomkunnehaværtgjortannerledesunderprosjektforløpetvårteratvikunnelagt opptilåbegynneåkodetidligere,slikatenslappdenhardesteinnspurten. Enmernøyaktigkoordineringinnbyrdesigr oppaheltfrastartenavhaddeogsåvært fordelaktig,dastartenblepregetavlittusikkerhetangåendemål. Vedrørendedetsoftwaremessigekunnevalgtågåttforkommersiellesoftware komponenter,somf.EksOracledatabase.Mendaviutgangspunktet ermotstandere motmangeavsoftwarekolossenesimperium,såvilleviholdeengjennomsyretstil beståendeavfrittdistribuerbareprogrampakker.

Åsetteingangmedetsliktprosjektsometindividiengruppekanværebådeen skremmendeogspennendeoppl evelse.Detågåfraenstatisksituasjonehverdagen bestårilåseoglæretilåskulleutføreestørrearbeidsoppgaveforenoffisiell oppdragsgiverkanvirkekaotisk.Fordetførstekandethendeduhavneriengruppe medpersonerduikkekjenner ellerduhavnerientilsynelatendeukontrollerbar situasjon.

Mensamtidigharfølelsenavåværedeleravenorganisasjonkommetetterhvert. Ekstragledeligerdetdegangerensergruppensenheterbegynneråtainitiativog utfyllerhverandreiarbeidso ppgaver.Dennegruppenkanvisiharfungertsliketter hvertsomtidenhargått.Hverenkeltsstyrkeharblittbraktframoglagtnedi arbeidsmengdensomharfåttfremdetteprosjektet. Fradag1bledetsattoppenfremdriftsplanogmilepæler.Perdags dato harikke gruppenoverskredetenmilepæl,menderimotgikkdetmangetidsfrister. Derutinervisatteoppangåenderapporteringogdendokumenterbaredelenav prosjektetharfungertprikkfritt.

Hvisenikkeharhattdenkunnskatvialleerforskjell igefør,såfårmandenidet manskajobbeienslikgruppe.Istartenhaddegruppenettsvarestrevmedåfåsamlet troppenitidetidspunkterogstedersomblesagt.Menetterhvertsomdeforskjellige individeraksepterteatallehaddeforskjelligeh andlingsmønstre,ogdermedfikkfram etkompromi.Såbegynteindivideneåhandlesomengruppe.Vifikkaltsåenfelles forståelseforhverandresvanerogbegynteåtilpasseoss.Slikatnoenkanskje programmertehjemme,mensandresattpåskolen.Enville kanskjebegynnefra morgen,mensandrevarnattmennesker.Dadenneforståelsenfaltpåplass,bledetmye enklereåfåtilenkoordineringavarbeidsoppgavene.Samtidigfikkviogsåsetthvade forskjelligepassetttiloghvilkestyrkerdehadde.

Altialt såfungerteintegrasjonsprosessenmegetbraigruppen,ogførteossframtillet ferdigproduktmedfullfunksjonalitet!

KAPITTEL8LITTERATURLISTE

Merkatnoenavreferanseneerlistetunderbådebokformogsomonlinemanual.Detteforå
gjøretilgjengelighetenavreferansenelettere.

Bøker:

Callaway,DustinnR.:Insideservlets,Server -SideProgrammingfortheJavaPlatform,
AddisonWesleyLongman,1999,ISBN0 -201-37963-5

Eckel,Bruce:ThinkinginJava,2nded. Revision11,2000 , ISBN0 -13 -659723 -8
(ogsåonline/elektroniskform)

Larman,Craig,"ApplyingUMLandpatterns",ISBN0 -13-748880-7

MarkWutka,"Hackingjava -thjavaprofessionalresourcekit",ISBN0 -7897 -0935 -X

Schildt,Herbert:Java2,FourthEdition,The CompleteReference,Osborne/McGraw -Hill,
2001, ISBN:0 -07-213084-9

Stallman,Richard:GNUEmacsManualThirteenthEditionUpdatedforEmacsVersion20.3,
1998, ISBN1 -882114-06-X
(ogsåsomonline/elektroniskform)

SunMicrosystems,JavaTM2SDKDocum entation,EnterpriseEdition,1.3BetaRelease,
January2001
(ogsåsomonline/elektroniskform)

SunMicrosystems,JavaTM2SDKDocumentation,StandardEdition,Version1.3,2000
(ogsåsomonline/elektroniskform)

TheApacheSoftwareFoundation,JakartaTOMCATVersion3.2.1Documentation,2000
(ogsåsomonline/elektroniskform)

Onlinemanualer/bøker(ielektroniskform/liggerogsåpåvedlagtd -rom):

MySQLAB,MySQLReferenceManualforversion3.23.33,March2001
(internetadresse: <http://www.mysql.com/documentation/mysql/bychapter>)

Database for Lagring og Analyse av Mikrostrukturbilder

Stallman, Richard: GNU Emacs Manual Thirteenth Edition Updated for Emacs Version 20.3,
(internetadresse: http://www.gnu.org/manual/emacs/html_mono/emacs.html)

Internetreferanser (standarder, artikler, web -sider, online -manualer):

"Form-based File Upload in HTML":

<http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1867.txt>

"Improved Performance with a Connection Pool" - Hans Bergsten, web developer journal:

http://www.webdevelopersjournal.com/columns/connection_pool.html

"Code Conventions for the Java™ Programming Language" - Sun Microsystems:

<http://java.sun.com/docs/codeconv/>

"THE JAVA LANGUAGE SPECIFICATION" - Sun Microsystems:

<http://java.sun.com/docs/books/jls/index.html>

"com.oreilly.servlet":

<http://www.servlets.com/cos/>

"klient & server, fagside, HIG":

http://www.hig.no/at/data/client_server/servlet_eksempler.phtml

"A Type IV JDBC driver for MySQL":

<http://mmysql.sourceforge.net/>

"The MySQL website" - MySQL AB:

<http://www.mysql.com/>

"The Sun Java website" - Sun Microsystems:

<http://java.sun.com/>

"The Jakarta Project" - Apache Software Foundation:

<http://jakarta.apache.org/>

Java Developer Connection (sm) (JDC) Tech Tips, April 11, 2000, USING CHECKSUMS

<http://developer.java.sun.com/developer/TechTips/2000>

Center for Innovative Computer Applications, Image File Formats List, 11/21/94

<http://www.cica.indiana.edu/graphics/image.formats.html>