

HOVEDPROSJEKT:

TITTEL

HEVN-J

Hvem Er hVor Når Java

FORFATTER(E):

Frode Olsen
Rolf Lien
Sven Erik Espeland
Erik Saghaug

Dato:

23.05.2002

SAMMENDRAG AV HOVEDPROSJEKT

Tittel:	HEVN-J Hvem Er hVor Når - Java	Nr. : 3 Dato : 23.05.02
Deltaker(e):	Frode Olsen Rolf Lien Sven Erik Espeland Erik Saghaug	
Veileder(e):	Frode Haug	
Oppdragsgiver:	Høgskolen i Gjøvik	
Kontaktperson:	Frode Haug	
Stikkord (4 stk)	Timeplan, Database, Java, WWW	
Antall sider:144	Antall bilag:11	Tilgjengelighet (åpen/konfidensiell): Åpen
Kort beskrivelse av hovedprosjektet: Videreutvikling av en eksisterende applikasjon som holder oversikt over ansatte i en stor bedrift. Dette blir gjort gjennom å lese innlagt informasjon fra en database. Denne informasjonen kan vises i en plattformuavhengig applikasjon og i en webside. Det er kun i applikasjonen det er mulig å legge inn informasjon. Det benyttes to egenutviklede verktøy for å konfigurere databasen og applikasjonen.		

FORORD

Hovedprosjektet HEVN-J (Hvem Er hVor Når - Java) er et avsluttende prosjekt ved den treårige høyskoleingeniørutdanning på Høgskolen i Gjøvik våren 2002. Prosjektets deltakere har vært Frode Olsen, Rolf Lien, Sven Erik Espeland og Erik Saghaug.

Oppgaven ble valgt etter en presentasjon på skolen, hvor mulige hovedprosjekter ble presenter i desember 2001. Noen av gruppens deltakere fattet da interesse for denne oppgaven, og etter samtaler med de fremtidige gruppe-medlemmene, ble det bestemt at dette var en oppgave for oss.

Siden dette valget ble tatt, har vi mange lange arbeidsdager med prosjektet. Ikke alle dagene har vært like morsomme, noen har rett og slett vært frustrerende og håpløse. Allikevel, når vi nå gjør opp status, er det de morsomme og interessante dagene, med pizza og brus, vi husker best. Når både programmet og rapporten har blitt ferdige, og slik vi ønsket at de skulle bli, kan vi ikke være annet enn fornøyde.

Vi vil takke følgende personer for hjelp og støtte under hovedprosjektet:

- Veileder og kundekontakt Frode Haug (for veiledning, hjelp og innspill).
- Øyvind Kolloen (for teknisk Java-veiledning)
- Testgruppen: Erik Hjelmås, Ivar Farup, Øyvind Kolloen (for mye testing, kreative løsningsforslag og god tålmodighet)
- Alle andre testbrukere (for testing og feilrapportering)
- Prosjektgruppen F@T (for testing og faglige innspill)
- Vaktmesterne ved HiG (for administrering tilrettelegging av grupperom)

Gjøvik 23.05.2002

Rolf Lien

Frode Olsen

Sven Erik Espeland

Erik Saghaug

1. INNLEDNING	4
1.1. PROBLEMOMRÅDE	4
1.2. AVGRENSNING	4
1.3. OPPGAVEBESKRIVELSE	5
1.4. MÅLGRUPPE	5
1.5. FORMÅL MED PROSJEKTOPPGAVEN	6
1.6. EGEN BAKGRUNN OG KOMPETANSE	6
1.7. ARBEIDSFORMER	7
1.8. DOKUMENTSTANDARD OG ORGANISERING AV RAPPORTEN	7
1.9. TERMINOLOGI	8
2. KRAVSPESIFIKASJON	9
2.1. INTRODUKSJON	9
2.2. BRUKERBESKRIVELSE	10
2.2.1. OMGIVELSER	10
2.2.2. SYSTEMETS BRUKERE	10
2.2.3. FUNKSJON	11
2.2.4. OPERASJON	11
2.2.5. ASPEKTER OMKRING LIVSSYKLUS	12
2.2.6. BEGRENSNINGER	12
2.3. FUNKSJONELL SPESIFIKASJON	13
2.3.1. FUNKSJONELL STRUKTUR OG TVERRELASJONER	13
2.3.2. DATA SPESIFIKASJON OG DATAORDLISTE	14
2.3.2.1. DATA RAMMEVERK	14
2.3.2.2. DATA INPUT	14
2.3.2.3. DATA OUTPUT	14
2.3.2.4. TVERR-FUNKSJONELLE DATA DEFININISJONER	15
2.3.3. OVERORDNEDE OPERASJONELLE SYSTEMKRAV	15
2.3.3.1. NORMAL OPERASJON	15
2.3.3.2. OPERASJON I FEILSITUASJONER	16
2.3.4. FUNKSJONELLE KRAV	16
2.3.4.1. FUNKSJONELLE KRAV FOR HEVN-J	16
2.3.4.2. ANDRE FUNKSJONELLE KRAV	23
2.4. SOFTWARE DESIGN BEGRENSNINGER	25
2.4.1. SOFTWARE STANDARDER OG SPRÅK	25
2.4.2. SOFTWARE PAKKER/VERKTØY	25
2.4.3. SOFTWARE KOMMUNIKASJONSSTANDARDER OG GRENSESNIITT	25
2.4.4. OPERATIVSYSTEM	25
2.5. ASPEKTER OMKRING LIVSSYKLUS	26
2.5.1. DOKUMENTASJON	26
2.5.2. MODUL OG INTEGRASJONSTESTING	26
2.5.3. KONFIGURASJON OG VERSJONSSTYRING	27
2.5.4. KRAV TIL SUPPORT, SERVICE OG VEDLIKEHOLD	27

2.6.	<i>ASPEKTER OMKRING INSTALLASJON</i>	27
2.6.1.	SOFTWARE INSTALLASJON	27
2.6.2.	OVERGANG/OMLEGGING	28
2.6.3.	OPPLÆRING	28
2.7.	<i>PROSJEKTSTYRING OG KVALITETSSIKRING</i>	28
3.	ANALYSE OG DESIGN	29
3.1.	<i>HVA SKAL SYSTEMET GJØRE</i>	29
3.1.1.	HEVN-J APPLIKASJONENS GRENSESNIITT	29
3.1.1.1.	”JEG ER...”	30
3.1.1.2.	”MELDINGER”	31
3.1.1.3.	”ANDRE...”	31
3.1.1.4.	”ROM/UTSTYR”	32
3.1.1.5.	”PERSONALIA”	33
3.1.1.6.	”OPPSETT”	33
3.1.1.7.	”HJELP”	34
3.1.2.	WEBAPPLIKASJON	34
3.1.3.	DATABASE	34
3.1.4.	KONFIGURASJONSPROGRAM	35
3.1.5.	DATABASEKONFIGURASJONSVERKTØY	35
3.2.	<i>HVORDAN LØSTE VI DET</i>	36
3.2.1.	ARKITEKTUR	36
3.2.2.	HEVN-J APLLIKASJONENS GRENSESNIITT	36
3.2.2.1.	”JEG ER...”	37
3.2.2.2.	”MELDINGER”	40
3.2.2.3.	”ANDRE...”	42
3.2.2.4.	”ROM/UTSTYR”	44
3.2.2.5.	”PERSONALIA”	46
3.2.2.6.	”OPPSETT”	47
3.2.2.7.	”HJELP”	49
3.2.3.	KONFIGURASJONSVERKTØY	50
3.2.4.	DATABASEKONFIGURASJONSVERKTØY	52
3.2.5.	WEBAPPLIKASJON	53
3.2.6.	DATABASE	55
3.2.6.1.	BESKRIVELSE AV DATABASEN	56
4.	KODING	60
4.1.	<i>VALG AV VERKTØY</i>	60
4.1.1.	HOVEDPROGRAMMET	60
4.1.2.	WEB-GRENSESNIITTET	60
4.1.3.	DATABASE	60
4.1.4.	INSTALLASJONSPROGRAM	61
4.2.	<i>BRUK AV XP</i>	61
4.2.1.	HVA HAR VI LAGT VEKT PÅ VED XP	61
4.3.	<i>PRINSIPPER VED KODING</i>	63
4.3.1.	KLASSEINNDELING	64
4.3.2.	KOMMENTARER	65
4.3.3.	KODEFORDELING	66

4.4.	<i>EGNE KOMPONENTER</i>	67
4.4.1.	<i>TIMETABLE-KOMPONENTEN</i>	67
5.	TESTING OG KVALITETSSIKRING	69
5.1.	<i>TESTING</i>	69
5.2.	<i>METODER FOR Å SIKRE KVALITET – PROSESSEN</i>	70
5.3.	<i>METODER FOR Å SIKRE KVALITET – PRODUKTET</i>	71
6.	INSTALLASJON	72
6.1.	<i>BRUKERAKSEPTANSE</i>	72
6.2.	<i>KONFIGURASJON</i>	72
6.3.	<i>INSTALLERING</i>	72
7.	KONKLUSJON	73
7.1.	<i>DRØFTING AV VALG OG RESULTATER</i>	73
7.1.1.	<i>VALG AV SYSTEMUTVIKLINGSMETODE</i>	73
7.1.2.	<i>TIDSPLANER</i>	74
7.1.2.1.	<i>GANTSKJEMA</i>	74
7.1.2.2.	<i>MILEPÆLER</i>	77
7.2.	<i>KRITIKK AV OPPGAVEN</i>	78
7.2.1.	<i>PROSESSEN</i>	78
7.2.2.	<i>PRODUKTET</i>	78
7.3.	<i>VIDERE ARBEID</i>	79
7.4.	<i>HOVEDKONKLUSJON</i>	80
7.5.	<i>EVALUERING AV GRUPPENS ARBEID</i>	81
7.5.1.	<i>INNLEDNING</i>	81
7.5.2.	<i>ORGANISERING</i>	81
7.5.3.	<i>FORDELING AV ARBEIDET</i>	81
7.5.4.	<i>PROSJEKT SOM ARBEIDSFORM</i>	81
7.5.5.	<i>SUBJEKTIV OPPLEVELSE AV HOVEDPROSJEKTET</i>	82
8.	LITTERATURLISTE	84
9.	VEDLEGG	85

1. INNLEDNING

1.1. PROBLEMMOMRÅDE

Hvem er hvor og når? Og hvordan får du enklest kontakt med dem? Dette er viktig å ha oversikt over i alle organisasjoner, uavhengig av type og størrelse. Forskjellige mer eller mindre manuelle systemer, noen heller usystematiske, har blitt brukt og brukes fortsatt for å finne ut hvor medarbeidere og ansatte befinner seg nå og de nærmeste dagene.

En mulig løsning på slike problemer er å la alle ansatte opprette sin egen timeplan som enkelt kan holdes oppdatert. Deretter må disse "legges på ett felles sted" hvor man kan lese hverandres timeplaner. På det samme stedet vil det være fornuftig å samle kontaktopplysninger som telefonnumre og adresser, på arbeid så vel som privat. Kan dette "felles sted" også være en naturlig møteplass for å utveksle meldinger, er de fleste problemene som er beskrevet innledningsvis løst.

I en organisasjon hvor nesten alle ansatte har tilgang til en PC koblet i nettverk, er det naturlig å gjøre timeplanen elektronisk, og å samle alle timeplaner, kontaktopplysninger og andre personalia i en database. Nettverket gjør det også mulig å sende meldinger til hverandre, og øyeblikkelig få tilbakemelding når den har blitt lest av mottakeren.

Slike løsninger finnes selvsagt allerede. Ett velkjent eksempel på en kommersiell tidsplanlegger med innsikt i andres timeplaner og med muligheter for meldingsutveksling er Microsoft Outlook, og det finnes flere andre med liknende funksjonalitet. Felles for de fleste er ett stadig voksende antall funksjoner og derav et stadig mer komplisert brukergrensesnitt og ikke minst ett utall konfigurasjonsmuligheter. Dette blir gjerne tunge applikasjoner, hvor bare de færreste bruker mer enn en brøkdel av tilgjengelig funksjonalitet. I tillegg er de ofte kostbare i innkjøp og kompleksiteten gjør de dyre å sette i drift og vedlikeholde.

1.2. AVGRENSNING

Programmet "Hvem Er hVor Når" (HEVN) som har vært i bruk på Høgskolen i Gjøvik har begrenset funksjonalitet i forhold til de fleste tidsplanleggere. Dette har medvirket til å gjøre det til et brukervennlig program hvor så å si all funksjonalitet ligger åpent, uten nedtrekksmenyer og behov for kompliserte innstillinger. I løpet av den tiden HEVN har vært i bruk har det kommet en rekke ønsker om ny funksjonalitet. Noen av disse ønskene kan det være aktuelt å ta med i en ny versjon, men vi føler at det hele tiden må gjøres en avveining mot et enkelt og brukervennlig grensesnitt. Dermed vil også den nye versjonen først og fremst være en tidsplanlegger for ukedager, på personnivå, med muligheter for utveksling av korte meldinger. I tillegg vil den nye versjonen i likhet med den gamle inneholde en personaliadel med adresser, telefonnumre, CV og liknende.

Den viktigste grunnen til å utvikle "Hvem Er hVor Når - Java" med den eksisterende HEVN som utgangspunkt, er muligheten til å få en plattformuavhengig løsning som kan kjøres på de fleste operativsystemer. Den eksisterende versjonen av HEVN kan bare kjøres på Windows-plattformen. Dette er lite tilfredstillende for skolens Unix/Linux- og Mac-brukere som ikke får kjørt HEVN. Teknologien som er valgt for å oppnå plattformuavhengighet er Java.

1.3. OPPGAVEBESKRIVELSE

Våren 1998 var det et svært så suksessfullt hovedprosjekt ("Hvem-Er-hVor-Når") som laget et Windows-program som de fleste av skolens ansatte kjører på sin PC.

Dette programmet ønsket HiG å få laget i en ny Java-versjon. Vår oppgave ble å lage en ny applikasjon som skulle ha det samme brukergrensesnittet og den samme funksjonaliteten som den gamle Delphi-versjonen hadde. I tillegg var det ønskelig med noe ny funksjonalitet, og enkelte forbedringer.

For oss virket dette som en interessant problemstilling, med mange nye og utfordrende oppgaver. Kravspesifikasjonen var mer eller mindre ferdig definert gjennom den eksisterende versjonen av HEVN. Prosjektet har derfor hovedsaklig bestått av utvikling i verktøyet Borland JBuilder 6.

Å videreutvikle systemer som allerede er i drift, var for oss en interessant oppgave. Her hadde vi muligheten til å bruke mange av de disiplinene vi har lært gjennom utdannelsen. At systemet allerede hadde mange brukere, var også med på å gjøre det hele mer realistisk med tanke på vår fremtidige arbeidskarriere. Vi følte hele tiden at det var forventninger fra mange brukere, og alle innspillene som kom i utviklingsfasen inspirerte oss under arbeidet.

1.4. MÅLGRUPPE

HEVN-J er utviklet for de ansatte ved HiG, det er derfor naturlig at denne rapporten har disse som målgruppe. Likevel har rapporten nødvendigvis enkelte tekniske deler som for noen av systemets brukere ikke er av interesse. Disse delene av rapporten er derimot beregnet for fremtidige administratorer og driftsansvarlige ved HiG og andre interesserte som trenger mer teknisk informasjon om HEVN-J. Dette vil typisk være for fremtidige studentprosjekter og lærere ved skolen. Rapporten er også en dokumentasjon for oppdragsgiver på det arbeidet vi har utført. Sensor og veileder på prosjektet legger denne rapporten til grunn for sin karaktersetning av vårt arbeid, og det er derfor viktig at også denne målgruppen blir tatt hensyn til.

Rapporten er ikke ment som en brukerveiledning eller instruksjonsbok for vår applikasjon, da dette er dekket av applikasjonens hjelpemeny.

1.5. FORMÅL MED PROSJEKTOPPGAVEN

HiG ønsket en plattformuavhengig utgave av den allerede eksisterende versjonen av HEVN. Bakgrunnen for dette ønsket var at antall bruker av andre operativsystemer enn Windows, hadde økt. Uten en slik plattformuavhengig løsning, ville det bli vanskelig å få alle ansatte til å benytte HEVN. Gjennom flere år i drift hadde HEVN vist seg å være et enkelt og oversiktlig system for å holde orden på lærerne og deres daglige/ukentlige *aktiviteter*. Det var derfor et sterkt ønske om fortsatt å kunne benytte et liknende system, som alle ansatte aktivt kunne benytte, uten at det ble satt krav til valg av operativsystem. Etter fire års aktiv bruk av HEVN, hadde det også dukket opp enkelte ønsker om forbedret/ny funksjonalitet.

Ved å lansere denne problemstillingen som en hovedprosjektoppgave håpet høgskolen å få løst problemet med plattformuavhengighet.

1.6. EGEN BAKGRUNN OG KOMPETANSE

Deltakerne på dette hovedprosjektet har relativt ulik bakgrunn, felles for alle er at vi har gjennomført tre år på HiG. Tre av oss har gått systemutviklingslinja det siste året, mens fjerdemann har gått driftslinja. Disse ulike linjevalgene har gjort at gruppa har hatt kompetanse på flere felt, noe som kommet godt med under arbeidet med prosjektet.

Vi har tidligere på HiG gjennomført et prosjekt sammen i faget ”Økonomi og prosjektstyring”, noe som gjorde at vi var kjent med hverandre fra før. Dette har gjort prosjektstyringsdelen av hovedprosjektet mye lettere enn hva det kunne ha blitt uten kjennskap til hverandre.

Faglig har det vist seg at fellesfag som ”Objektorientert programmering” og ”Algoritmiske metoder I” har vært gode å ha. Da vi startet prosjektarbeidet ved juletid, var det kun to av gruppens deltakere som hadde programmert i Java. Dette førte til at noe tid i starten av prosjektet, gikk med til å lære seg å jobbe med dette språket. Ulike kunnskaper og interesser har også ført til at deltakerne har arbeidet med ulike deler av prosjektet.

Det er relativt stor aldersforskjell på gruppens deltakere, yngste er 23 år og eldste 40 år. Dette har gitt gruppen et mer nyansert syn på flere problemstillinger, og forhåpentligvis vært med på å gjøre applikasjonen vår bedre.

Enkelte av gruppen deltakere har erfaring fra arbeidslivet, men ikke med noe som tilsvarer det arbeid vi har gjort i forbindelse med hovedprosjektet.

1.7. ARBEIDSFORMER

Gruppen har fått tildelt et grupperom i kjelleren på A-bygget, og arbeidet med hovedprosjektet har i hovedsak foregått der. Gruppen har selv stått for maskinvare og utstyr, slik at vi har kunnet styre vårt arbeid selv. Behovet for å tilegne oss nye kunnskaper innen Java-programmering førte til at det ble brukt noe tid på biblioteket med å finne oppdatert litteratur om emnet [1, 2, 3]. Det var mye nytt å ta stilling til i begynnelsen av prosjektet, og mye av arbeidet den første tiden ble gjort i plenum. Senere har det vært naturlig å dele arbeidet mer opp. Vi har allikevel etterstrebet å jobbe sammen i par, som er en av forutsetningene når man jobber etter den utviklingsmodellen som ble valgt for vårt prosjekt.

Skriving av rapporter og store deler av programmeringsarbeidet har blitt gjort enkeltvis, for å spare tid. Da vi har sittet sammen på et grupperom, har mye av arbeidet likevel ofte blitt gjort i felleskap. Dette er en prioritering som er gjort, men vi tror ikke dette har ført til dårligere kvalitet på vårt arbeid.

Vi har hele tiden integrert nye deler i applikasjonen vår, og testing i forbindelse med dette, har vært et viktig arbeide. Denne arbeidsmetoden, har fungert bra for oss som gruppe, og har gitt en bra applikasjon.

Rapporter og dokumenter som er levert til oppdragsgiver, har alltid blitt lest og gjennomgått i plenum. Vi har benyttet veileder til å hjelpe oss med utforming av disse, ved behov.

1.8. DOKUMENTSTANDARD OG ORGANISERING AV RAPPORTEN

Denne hovedprosjektrapporten følger standarden for skrifttyper fastsatt i kravspesifikasjonen i kapittel 2.5.1. Nedenfor har vi forklart de andre metodene som har blitt brukt for å skille ulike deler av rapporten fra hverandre.

Der vi har referert til ulike kapitler i rapporten er kapittelnummeret benyttet. En referanse til dette kapitlet blir da som følger, se kapittel 1.8. Referanser til litteraturlisten er gjort på følgende måte [1]. For å vise til vedleggene i rapporten har vi brukt, se vedlegg [J].

Kildekode gjengitt i rapporten, er skrevet med fonten Courier New, skriftstørrelse 8.

Første forekomsten av et ord, som er forklart i rapporten er skrevet i kursiv, og forklart i vedlegg [A].

Figurer og bilder i rapporten er nummerert hver for seg, fortløpende i hvert kapittel. Første bilde i kapittel tre, vil ha følgende nummerering; Bilde 3-1.

De ulike kortene i HEVN-J applikasjonen er skrevet med stor forbokstav, og omsluttet med hermetegn. Navn på tabeller i databasen er skrevet med stor forbokstav, mens feltene er skrevet med små bokstaver og omsluttet med hermetegn.

Denne rapporten er delt inn i 9 kapitler. Innholdet i disse kapitlene er som følger:

Kapittel 1 gir en kort introduksjon til rapporten. Her blir dokumentstandard og terminologi forklart.

Kapittel 2 inneholder kravspesifikasjonen. Her blir kravene til systemet detaljert forklart.

Kapittel 3 inneholder analyse og design for prosjektet. Her er det forklart hva systemet skulle gjøre, og hvordan vi valgte å gjøre det.

Kapittel 4 forklarer hvordan arbeidet med koden er løst i prosjektet. Valg av utviklingsverktøy for de ulike delene av systemet blir forklart, og hvordan valgt systemutviklingsmodellen ble brukt under dette arbeidet.

Kapittel 5 handler om hvordan arbeidet med testing og kvalitetssikring har blitt gjort under prosjektet.

Kapittel 6 forklarer aspekter omkring installasjon av systemets ulike deler.

Kapittel 7 er konklusjonen på dette prosjektarbeidet. Her drøftes valg og resultater av prosjektarbeidet. Kritikkk av oppgaven, videre arbeid og en hovedkonklusjon avslutter dette kapittelet.

Kapittel 8 inneholder en litteraturliste for rapporten.

Kapittel 9 er en oversikt over alle vedleggene til denne rapporten

1.9. TERMINOLOGI

Vi har benyttet oss av en del ord og uttrykk som er engelsk, eller av en slik karakter at de trenger en nærmere forklaring. Forskjellige definisjoner er samlet i vedlegg [A]. Vi har forsøkt å benytte norske ord der dette har vært naturlig, selv om mye av litteraturen benyttet under prosjektet er skrevet på engelsk.

2. KRAVSPESIFIKASJON

2.1. INTRODUKSJON

Da vi startet arbeidet med hovedprosjektet, var utgangspunkt fra oppdragsgivers side at kravspesifikasjonen var mer eller mindre ferdig definert. Dette var et arbeid som var utført tidligere, og som oppdragsgiver ikke mente det var nødvendig å bruke tid på. Denne kravspesifikasjonen er å finne i rapporten laget av hovedprosjektgruppen HEVN [6]

HEVN hadde vært i drift i fire år, og hadde fungert meget tilfredstillende hos brukerne. Ut ifra dette ble det enighet om at arbeidet som var gjort med HEVN sin kravspesifikasjonen, var et solid fundament å jobbe videre med.

Da det i forbindelse med vår fornyelse av programmet var ønskelig med noe ny funksjonalitet, var det på dette område vi trengte å spesifisere nye krav. Dette arbeidet ble gjort i tett samarbeid med oppdragsgiver, og alle spesifiseringer og avgjørelser ble tatt i samråd med han. I arbeidet med å gjenskape HEVN i Java, ble det jobbet både med den eksisterende kravspesifikasjonen og den eksisterende HEVN applikasjonen slik den fremstod. Vi benyttet da dette som en kravspesifikasjon for vårt arbeid.

Kunden har benyttet en brukergruppe i arbeidet med å definere nye krav til systemet. Gjennom de fire år HEVN har vært i bruk, har det også dukket opp enkelte ønsker om forbedringer/endringer. Disse ønskene har vi gått gjennom med kunden og spesifisert nye krav der dette har vært nødvendig.

Det er et ønske fra oppdragsgiver at det blir laget en webapplikasjon som viser deler av databasens innhold. Denne webapplikasjonen skal inneholde de ansattes timeplaner og timeplanene for rom og utstyr ved skolen.

Vi har arbeidet med utviklingsmodellen "eXtreme Programming" (XP) på dette prosjektet. (For mer om XP se kapittel 4.2 og vedlegg [C]). Arbeid med kravspesifikasjon skal i tradisjonelle utviklingsmodeller gjøres tilnærmet ferdig før man starter arbeidet med koding. I XP er det ikke lagt spesielt vekt på å skrive en fullstendig kravspesifikasjon. I stedet er det lagt opp til at krav fra brukeren skal taes opp og eventuelt legges inn i applikasjonen underveis i utviklingsfasen. Under store deler av utviklingsarbeidet, er det da også jobbet på denne måten. Vi føler at det derfor ikke er helt riktig å skrive en fullstendig kravspesifikasjon i vårt prosjekt. Bakgrunnen for at det allikevel har blitt skrevet en relativt grundig kravspesifikasjon, er med tanke på leserne av rapporten og hva som er tradisjon for rapportskrivning ved hovedprosjekter.

Selv om vi ikke har jobbet mye med tradisjonell kravspesifisering, er mye benyttet av det arbeidet som ble gjort av HEVN gruppen. Dette arbeidet har vært et rammeverk som vi har benyttet under vår kravspesifisering.

2.2. BRUKERBESKRIVELSE

2.2.1. OMGIVELSER

Systemet består av flere deler som opererer i forskjellige omgivelser. Databasen blir administrert av IT-tjenesten, og ligger på en av deres servere. Denne serveren står på et server-rom, og ingen uvedkomne har tilgang til dette rommet.

Selve HEVN-J applikasjonen ligger på den ansattes maskin. Disse maskinene, er i utgangspunktet begrenset til den som disponerer kontoret. Det skal allikevel være mulig å logge seg på andre maskiner enn den man vanligvis bruker. Det er en forutsetning at alle disse maskinene kan koble seg til databaseserveren.

Konfigurasjonsverktøyet for HEVN-J skal også kunne kjøres på alle skolens maskiner, men det må finnes en begrensning for hvem som skal kunne bruke dette.

Web-grensesnittet er ment benyttet utenfor skolens nettverk. Maskinene må ha nettverkstilgang og nødvendig programvare installert. Se kapittel 2.2.6 for mer detaljerte systemkrav.

2.2.2. SYSTEMETS BRUKERE

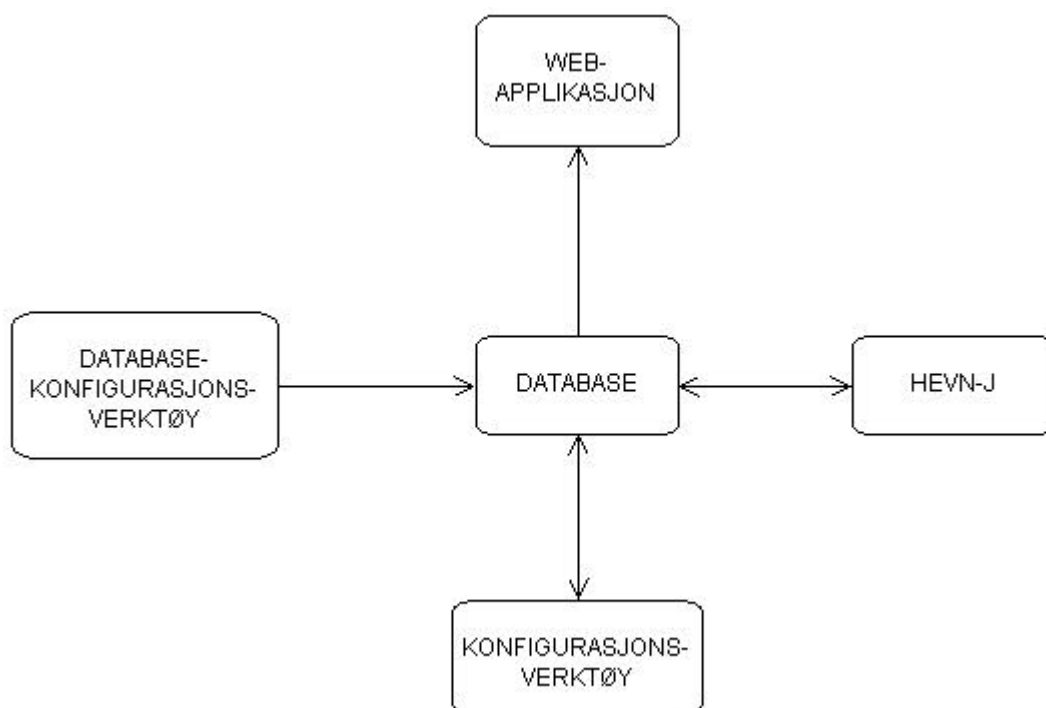
HEVN-J skal brukes av alle ansatte på skolen. Ulike bakgrunnskunnskaper fører til at vi har en sammensatt brukergruppe å forholde oss til. Applikasjonen må derfor være enkel å forstå og lett å bruke for både nye og mer erfarne brukere. Da mange av skolen ansatte tidligere har benyttet HEVN, er det viktig at vi lager en applikasjon som er funksjonelt lik denne. Ny funksjonalitet som blir laget, må ha det samme brukergrensesnittet som det den eksisterende funksjonaliteten har. Det vil fortsatt ikke være noe funksjonalitet som *kun* er tilgjengelig ved høyreklikking på musa. Likeledes er ingen funksjonalitet skjult i nedtrekksmenyer. For å tilfredstille de mer erfarne brukerne, vil det være mulig å benytte tastaturet ved betjening av applikasjonen. Det vil være mulig å benytte blant annet piltaster, del-tast og esc-tast for å betjene applikasjonen.

Web-applikasjonen for HEVN-J vil ikke skille mellom studenter og ansatte, og kun grunnleggende kjennskap til nettlesere er nødvendig for å benytte denne.

Installasjonsprogrammet for databasen, stiller krav om teknisk innsikt, og er beregnet for de ansatte ved skolens IT-tjeneste. Administrasjonsverktøyet for HEVN-J er også beregnet for erfarne databrukere, og det er derfor ikke lagt like mye arbeid i brukergrensesnittet i disse to applikasjonene.

2.2.3. FUNKSJON

Systemet skal bestå av totalt 6 deler; database, HEVN-J applikasjon, webapplikasjon, konfigurasjonsverktøy, databasekonfigurasjonsverktøy og installasjonsverktøy. Figur 2-1 viser de ulike delene av systemet. Installasjonsverktøyet er ikke tatt med på figuren, da dette kun blir brukt ved installasjon. Kommunikasjon mellom de ulike delene av systemet skjer ved at applikasjonene og konfigurasjonsverktøyene sender *SQL* spørringer til databasen. I HEVN-J har brukeren mulighet til å se og endre informasjon på timeplan for personer og rom/utstyr. Hente ut informasjon om personalia og sende meldinger seg imellom. Administrasjonsverktøyet gir mulighet til å endre systemspesifikke parametere for HEVN-J applikasjonen. Databasekonfigurasjonsverktøyet brukes til å endre på databasespesifikke parametere.



Figur 2-1 Interaksjon mellom ulike deler av systemet.

2.2.4. OPERASJON

HEVN-J applikasjonen skal kjøres på maskiner som er tilkoblet skolens netteværk. Dette vil bli gjort ved at applikasjonen automatisk starter ved oppstart av maskinene. Applikasjonen skal ikke avsluttes uten at brukeren spesielt ber om det. Stabiliteten til applikasjonen vil avhenge av operativsystemet, *Java VM*, nettverket og databasens stabilitet. Ulike feiltilstander som kan oppstå under drift, er nærmere beskrevet i kapittel 2.3.3.2.

Webapplikasjonen vil bli kjørt i maskinens valgte nettleser, og vil dermed være avhengig av at denne er stabil. Stabiliteten til denne applikasjonen vil avhenge av nettleseren, operativsystemet, nettverkstilkoblingen til skolens webserver og databasen.

Databasen kjøres på en av skolens databaseservere tilkoblet skolens nettverk. Det er en forutsetning at denne serveren er i kontinuerlig drift for at systemet skal kunne fungere. Oppdateringer og annet vedlikehold, av denne serveren, bør derfor skje utenom vanlig skoletid, slik at brukerne av HEVN-J ikke blir unødvendig rammet.

Systemet har ingen egne *backup*-rutiner, og det er en forutsetning at slike finnes på serverne.

2.2.5. ASPEKTER OMKRING LIVSSYKLUS

HEVN-J utvikles som HEVN i hovedsak for Høgskolen i Gjøvik. Da også andre skoler og institusjoner har vist sin interesse for HEVN, skal det utvikles med tanke på at systemet skal kunne flyttes.

Ved en eventuell flytting av systemet, skal det lages funksjonalitet i databasekonfigurasjonsverktøyet og i konfigurasjonsverktøyet slik at nødvendige parametere kan bli skiftet. Dette er nærmere beskrevet i kapittel 3.1.4 og 3.1.5.

Det skal ikke legges spesielt vekt på å utvikle generelle komponenter. Kalenderkomponenten bør allikevel kunne benyttes i andre applikasjoner. Dette bør være aktuelt for webapplikasjon.. For nærmere beskrivelse av kalenderkomponenten, se kapittel 4.4.1.

HEVN-J applikasjonen skal dokumenteres for brukergruppen ved hjelp av egne hjelpkort i selve applikasjonen. Det vil bli laget en egen kortfattet dokumentasjon til databasekonfigurasjonsverktøyet.

Det er skrevet en kontrakt mellom prosjektgruppen og HiG. Denne avtalen regulerer forholdet rundt videresalg, oppgradering og bruk av egenutviklede komponenter i fremtidige applikasjoner. Denne avtalen finnes som vedlegg [J].

2.2.6. BEGRENSNINGER

Systemkrav til alle maskiner som skal kjøre HEVN-J applikasjonen:

- Ingen krav til operativsystem, men må ha Java VM minimum versjon 1.4.0
- 40 MB ledig diskplass
- 64 MB RAM (Anbefalt 128 MB RAM)
- Riktig innstilt systemklokke
- Installert nettverkskort
- Skjerm med minimum oppløsning på 800*600
- Lydkort er anbefalt, men ikke påkrevd

Hardware og software krav til andre maskiner som blir involvert i systemet.

Databaseserveren:

- *MSSQL* server 2000
- Server collation skal være "SQL_Scandinavian_CP850_CI_AS"

- Spesifisert *TCP/IP* port

Web-serveren:

- *Apache* webserver
- Støtte for *Java-servlets (Tomcat)*
- Nettverkstilgang til databaseserveren
- Klokke må gå riktig

Web-klientene:

- Tilgang til Internett.
- Nettleser.

2.3. FUNKSJONELL SPESIFIKASJON

2.3.1. FUNKSJONELL STRUKTUR OG TVERRELASJONER

Applikasjonen består av et hovedvindu. Dette hovedvinduet er igjen delt inn i seks ulike undersider hvor applikasjonens funksjonalitet ligger. Disse sidene er skilt med bokmerker navngitt etter denne sidens funksjonalitet. Disse sidene heter ”Jeg er...”, ”Meldinger”, ”Andre...”, ”Rom/Utstyr”, ”Personalia” og ”Oppsett”. Nærmere forklaring av de ulike sidene finnes i kapittel 2.3.4.

Webapplikasjonen skal speile databasens informasjonen om hva som finnes i ”Andre...” og ”Rom/Utstyr” kortene. Det vil ikke bli gitt muligheter til å oppdatere informasjon om brukere eller rom/utstyr via Internett.

Databasen er en relasjonsdatabase som inneholder en rekke ulike tabeller. Det er denne databasen alle systemets deler henter informasjon fra.

Konfigurasjonsverktøyet skal brukes til å sette opp databasen og til å vedlikeholde denne. Dette er et selvstendig program, og skal kunne kjøres uavhengig av HEVN-J.

Databasekonfigurasjonsverktøyet skal brukes til å sette opp databasen for første gang, og for å generere en konfigurasjonsfil for HEVN-J.

Figur 2-1 viser samspillet mellom de ulike deler av applikasjonen.

2.3.2. DATA SPESIFIKKASJON OG DATAORDLISTE

2.3.2.1. DATA RAMMEVERK

Systemet skal håndtere data fra ulike deler av applikasjonen. Disse dataene kommer fra systeminnstillinger, brukerinnstillinger, personalia, meldinger og timeplandata. Timeplandata, er litt spesielt, da dette er lagrede aktiviteter for både personer og rom/utstyr.

2.3.2.2. DATA INPUT

Liste over alle data input feltene i HEVN-J er å finne som vedlegg [E].

2.3.2.3. DATA OUTPUT

HEVN-J gir output til skjerm i form av timeplaner for personer og rom/utstyr. Avlesning av registrerte brukeres personalia, meldinger fra disse brukerne og ved varsling av systemhendelser.

Applikasjonen skal ha en utskriftsfunksjonalitet som gir output i form av

- a. Timeplan for en ansatt
- b. Timeplan for et rom
- c. Liste over ansatte i alle avdelinger eller i en valgt avdeling.
- d. Liste over andre relevante telefonnummer for skolen.

Denne utskriftslisten over ansatte kan komponeres av brukeren. Følgende attributter er valgt som standardinnstillinger:

- Etternavn
- Fornavn
- Jobb tlf
- Rom
- Avdeling

I tillegg har brukeren mulighet til å velge følgende attributter:

- Tittel
- Stilling
- E-post
- Adresse
- Privat tlf
- Mobil tlf
- Fax

Utskrift av timeplaner, kan gjøres på ulike måter. Det skal være mulig å skrive timeplanene ut med følgende parametere:

- Alle aktiviteter denne uken med ukenummer
- Alle aktiviteter denne uken uten ukenummer
- Kun faste aktiviteter denne uken, uten ukenummer

2.3.2.4. TVERR-FUNKSJONELLE DATA DEFININISJONER

Informasjonsutvekslingen mellom ulike deler av applikasjonen skal foregå på to ulike måter. Det er SQL spørringer og *resultset*. SQL spørringene sendes til databasen fra applikasjon, databasen sender svar på denne spørringen tilbake i form av et resultset. Dette resultsettet kan bestå av mye data, eller kun en bekreftelse på at SQL spørringen er utført. En slik bekreftelse kommer typisk når det blir bedt om å endre eller slette data. Databasen svarer da enten at det gikk bra, eller at noe ble feil.

2.3.3. OVERORDNEDE OPERASJONELLE SYSTEMKRAV

2.3.3.1. NORMAL OPERASJON

Modus og kontroll

Systemet lages etter klient/server modellen. Denne modellen har en database som mange kliner foretar spørringer mot. Databasen må derfor ha samtidighetskontroll på forespørsler fra klientene, og kunne sørge for at alle integritetsregler overholdes.

Ytelse

Begrensningene for ytelse på vårt system vil være nettverkskapasiteten og databaseserveren. Ved HiG er begge disse faktorene godt ivaretatt, og ytelsen for vårt system er derfor bra.

Sikkerhet

Det forutsettes at alle servere som blir brukt av vårt system blir plassert på et rom med adgang kun for systemansvarlige personer. HEVN-J baserer seg på sikkerheten ved innlogging på maskinen, men for Linux-brukere vil *autentiseringen* skje mot en *pop3* server. For oppkobling mot databasen må riktig *dbconfig* fil brukes. Denne filen sørger for at ikke feil personer kan hente informasjon ut av databasen..

Oppstart og nedtagning

Oppstart og installasjon er beskrevet i vedlegg [K].

Tilgjengelighet

Systemet skal ha i teorien ha 100 prosent *oppetid*. Skulle databasen mot formodning gå ned, må HEVN-J restarteres.

Innebygde tester

Systemet skal utføre tester på alle inputfelter. Det skal testes på om noe er endret før det gis mulighet til å lagre.

2.3.3.2. OPERASJON I FEILSITUASJONER

Alle feilmeldinger, hvor det er mulig, skal gis fra vår applikasjon.

Databaseserveren

Dersom forbindelsen mellom klientene og databaseserver blir borte, skal brukeren gis beskjed om dette.

Ugyldig bruker

Hvis det blir forsøkt logget inn på system med en bruker som ikke er definert i databasen, skal det gis beskjed om dette.

Feil i dbconfig fil

Dersom det er feil i denne filen, skal det gis en feilmeldingen. Systemadministrator må kopiere inn riktig versjon av dbconfig filen for at systemet skal fungere.

Utskrift

Dersom utskrift fra applikasjonen ikke lykkes, skal det gis en feilmelding fra operativsystemet. Selve applikasjonen bør ikke påvirkes av dette.

2.3.4. FUNKSJONELLE KRAV

2.3.4.1. FUNKSJONELLE KRAV FOR HEVN-J

HEVN-J applikasjonen skal fungere som, og være visuelt lik HEVN. Det skal likevel lages noe ny funksjonalitet, samt at noe gammel funksjonalitet skal fjernes. Antall kort i applikasjonen skal økes til seks, for å få plass til funksjonalitet for reservasjon av rom og utstyr.

Vi har nedenfor forklart hva applikasjonen i grove trekk skal gjøre, samt vist bilde av hvordan HEVN ser ut.

For skjermbilder av hvordan vår applikasjon ble, se kapittel 3.2.2 .

”Jeg er...”

Dette kortet skal vise brukerens timeplan for gjeldene uke. Ukedagene skal vises fra klokken 00.00 til klokken 24.00. Det skal være mulig å bla seg frem og tilbake i ubegrenset tid ved hjelp av navigasjonsknapper. Ved å markere et område i timeplanen, skal det være mulig å registrere aktiviteter. Disse aktivitetene skal kunne defineres for flere uker. Alle aktiviteter skal kunne lages som private aktiviteter, som brukeren kun selv skal kunne se. Det skal være mulig å hoppe tre år frem i tid, og to år tilbake ved valg i en nedtrekksmeny. I tillegg skal det være mulig å hoppe mellom år og måneder i egne nedtrekksmenyer. Ukentlige timeplaner skal det være mulig å skrive ut. Det skal finnes funksjonalitet for enkelt å registrere at brukeren er ute en kort tur, uten å måtte markere dette som en aktivitet på timeplanen.

The screenshot shows a software window titled "Hvem-Er-Hvor-Når - versjon 0.4.0.45 (Release Candidate)". The main area displays a weekly schedule grid for the week of May 11-15, 1998. The days of the week are: Mandag (11.5.1998), Tirsdag (12.5.1998), Onsdag (13.5.1998), Torsdag (14.5.1998), and Fredag (15.5.1998). The time slots range from 8:00 to 16:00. Activities are listed in the grid cells, such as "Undervisning" and "Møte". A "Generell kommentar" field at the bottom contains the text: "A bus station is where a bus stop. A train station is where a train stop. I have a workstation.....".

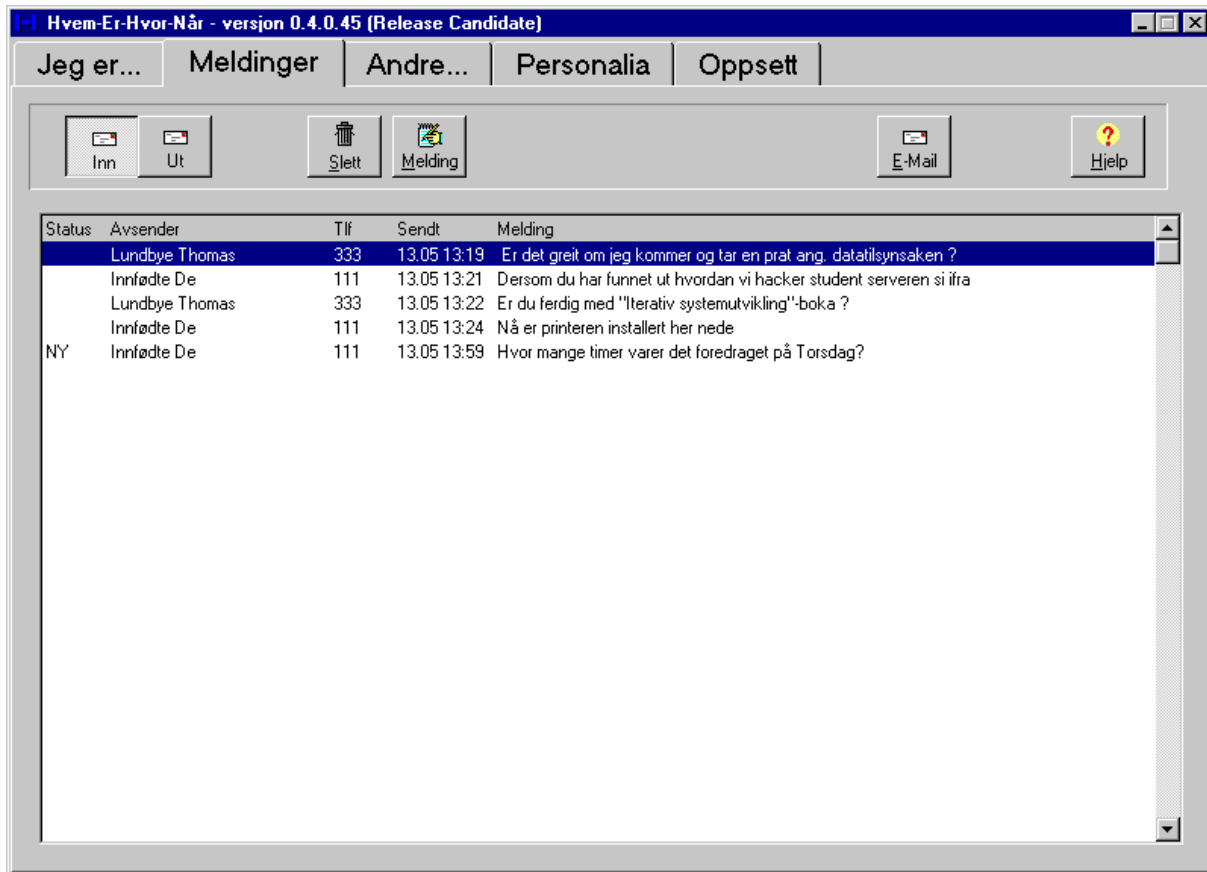
	Mandag 11.5.1998	Tirsdag 12.5.1998	Onsdag 13.5.1998	Torsdag 14.5.1998	Fredag 15.5.1998
8:00					
9:00	Undervisning Programmering mot WWW i C007				Tannlegen
10:00				Undervisning Elektronisk publisering i Eureka	
11:00					
12:00	Lunsj	Lunsj	Lunsj	Lunsj	Lunsj
13:00		Undervisning Programmering mot WWW i C007	Møte A122		
14:00	Møte fast veiledningsmøte				Avspasering
15:00					
16:00	Gått for dagen	Gått for dagen	Gått for dagen	Gått for dagen	

Generell kommentar: A bus station is where a bus stop. A train station is where a train stop. I have a workstation.....

Bilde 2-1 “Jeg er...” kortet i HEVN

”Meldinger”

Det kortet skal benyttes til å sende meldinger mellom brukerne av systemet. Meldingene skal ligge i henholdsvis en inn- og en utboks. Dette kortet skal også benyttes i forbindelse med registrering av reserverasjoner. Meldinger skal markes etter statusen på den. ”Ny” og ”Ulest” for vanlige meldinger, og ”J”, ”N” og ”Vent” for meldinger fra ”Rom/Utstyr” kortet. Det skal ikke være mulighet for å sende e-post fra denne delen av applikasjonen.



Bilde 2-2 “Melding” kortet i HEVN

”Andre...”

Dette kortet brukes til å se andre brukeres timeplaner og deres personalia. Ved å benytte en liste over alle systemets brukere, skal man kunne bla mellom brukerne. Når brukeren har valgt en person i listen, skal denne personens timeplan og personinformasjon være tilgjengelig. Hvis en person er valgt i listen, skal det være mulighet for å sende melding til denne. Administratorer av systemet, skal her ha mulighet til å endre opplysninger om ansatte. I tillegg skal det finnes en funksjon for å finne felles tid for et valgt antall ansatte. På samme måte skal det kunne sendes en felles melding til flere ansatte på en gang. Det skal fra dette kortet være mulig å skrive ut en liste med personinformasjon, av typen navn, kontor og telefonnumre. Mulighet for å søke på lagrede personaldata i databasen, skal også finnes. I tillegg til navigasjonsknapper som på kortet ”Jeg er...”, skal det også her være mulig å hoppe rask frem i tid ved hjelp av en nedtrekksmeny. E-post funksjonen som er i HEVN skal fjernes.

Klikk på første bokstav i etternavnet

	Mandag 18.5.1998	Tirsdag 19.5.1998	Onsdag 20.5.1998	Torsdag 21.5.1998	Fredag 22.5.1998
8:00			Hovedprosjek.. HEVN	Kristi himmelfarts..	
9:00	Undervisning Prøve med MAMM	Eksamen ledelse			
10:00	Hovedprosjek.. Statusmøte HEVN rom A122				Stud Ass Storlabben
11:00					
12:00			Lunsj		
13:00	Undervisning Ledelse				
14:00					Kurs/Seminar Oslo
15:00		Hovedprosjek.. Møte med FrodeH			
16:00					

Sist utlogget: 13.05.98 16:22:07

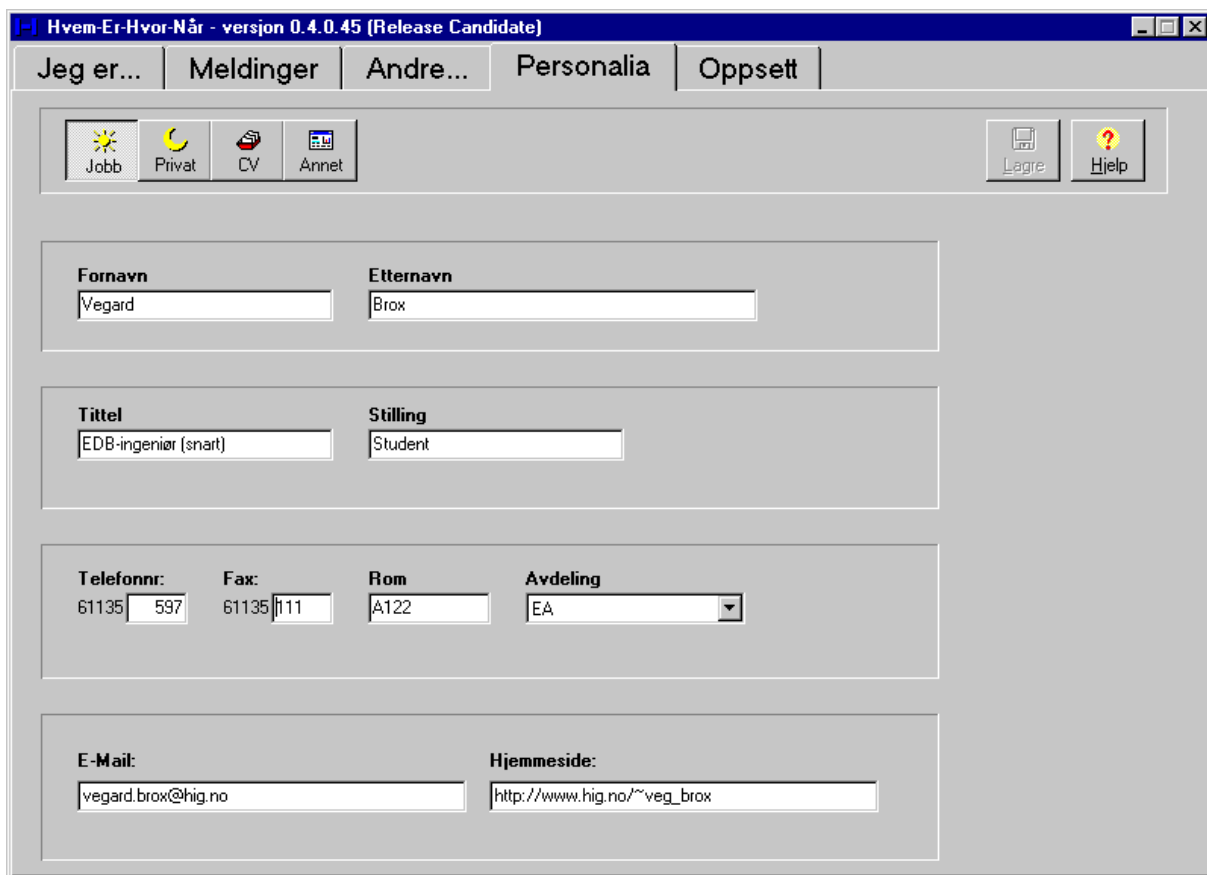
Generell kommentar: 24.04.98 11:18

Bilde 2-3 "Andre..." kortet i HEVN

”Personalia”

Dette kortet skal brukes for å registrere personalia om brukeren. Dataene er delt inn i fire kategorier:

- ”Jobb”, skal inneholde tittel, stilling, avdeling, rom, jobbtelefon, fax, avdeling, e-post og hjemmeside
- ”Privat”, skal inneholde adresse, privattelefon, mobiltelefon, e-post og hjemmeside.
- ”CV” skal inneholde utdanning og jobb/erfaring.
- ”Annet” skal inneholde annen personlig informasjon, som tillitsverv og liknende.



Hvem-Er-Hvor-Når - versjon 0.4.0.45 (Release Candidate)

Jeg er... Meldinger Andre... **Personalia** Oppsett

Jobb Privat CV Annet Lagre Help

Fornavn Vegard **Etternavn** Brox

Tittel EDB-ingeniør (snart) **Stilling** Student

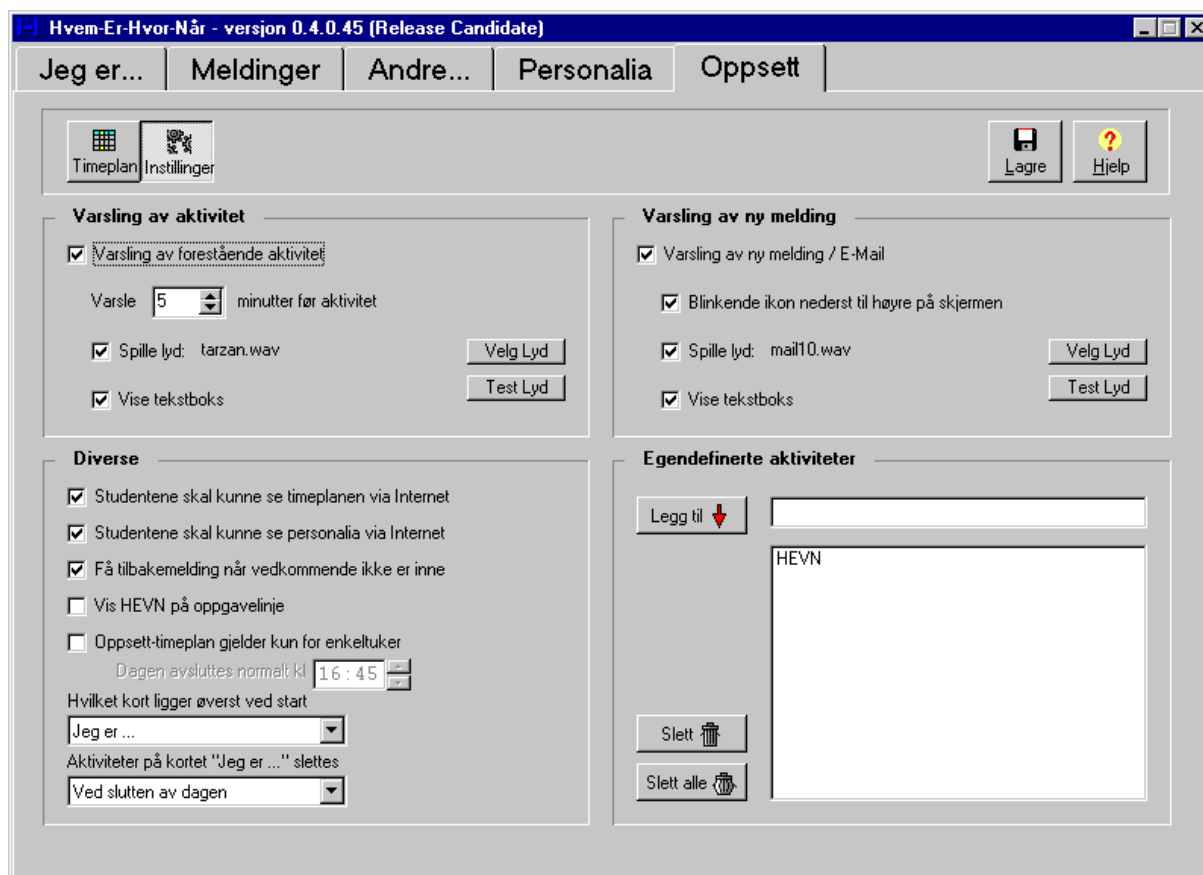
Telefonnr: 61135 597 **Fax:** 61135 111 **Rom** A122 **Avdeling** EA

E-Mail: vegard.brox@hig.no **Hjemmeside:** http://www.hig.no/~veg_brox

Bilde 2-4 “Personalia” kortet i HEVN

”Oppsett”

Dette kortet skal benyttes for brukerinnstillinger av HEVN-J. Her skal hver bruker kunne sette opp hvordan man ønsker at applikasjonen skal reagere på ulike systemhendelser. Hendelser som kan spesifiseres skal være ”Varsling av aktivitet”, ”Varsling av ny melding”, ”Diverse” og ”Egendefinerte aktiviteter”. Under ”Diverse skal det være mulig å begrense hva som skal vises i webapplikasjonen, samt å velge om applikasjonen skal gi tilbakemelding på om meldingsmottaker er opptatt ved sending av melding. Det skal være mulig å velge hvilket kort som skal ligge øverst ved start. Lagringstiden for aktiviteter på kortet ”Jeg er...” skal kunne settes her. E-postprogram og nettleser for Linux skal kunne angis på dette kortet. Muligheten for å velge om HEVN skal vises på oppgavelinjen skal fjernes. Det samme gjelder mulighet for å velge ”Oppsett-timeplanen gjelder kun for enkeltturer”.



Bilde 2-5 ”Oppsett” kortet i HEVN

”Rom/Utstyr”

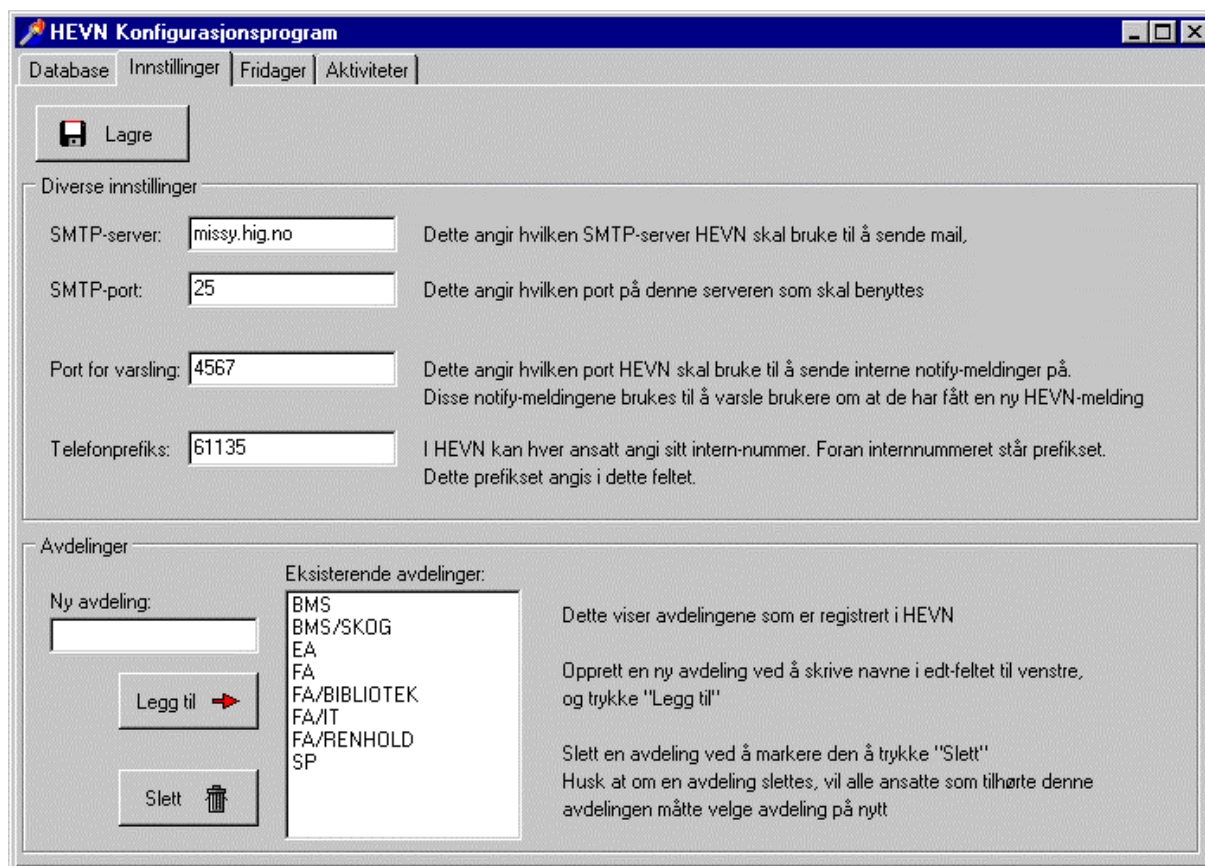
På dette kortet, skal brukerne av systemet kunne reservere rom og utstyr på skolen. Det skal finnes funksjonalitet for å lage et nytt, endre eksisterende eller slette rom og utstyr. Lagrede rom og utstyr, skal vises i egne lister. Ved markering av ett rom eller utstyr, skal den tilhørende timeplanen for disse *ressursene* vises. Det skal benyttes samme manøvreringsmetode for timeplanen på dette kortet, som den som blir benyttet på kortet ”Jeg er...”. Det skal være mulig å sette ulike rettigheter på de ulike ressursene. Eier, administrator og vanlig bruker skal kunne håndteres og gis ulike rettigheter. Avhengig av hvilke begrensinger som er satt, skal systemets brukere kunne markere ønsket tid for reserveringer timeplanen. Det skal være mulig å begrense ressursen til kun å være tilgjengelig for reservering i en tidsperiode. Bestilling og bekreftelse på reserveringer skal kunne foregå med meldingsfunksjonaliteten på kortet ”Melding”. Markering av reserveringstidspunkt i timeplanen, skal gjøres etter samme mønster som på kortet ”Jeg er...”.

Dette er et nytt kort, og det finnes derfor ikke et skjermbilde fra HEVN

2.3.4.2. ANDRE FUNKSJONELLE KRAV

”Konfigurasjonsverktøy”

Dette verktøyet skal brukes til å konfigurere HEVN-J. Opprettelse og sletting av brukere skal være mulig. Det skal også kunne genereres faste offisielle fridager i databasen og finnes funksjonalitet for å legge inn egendefinerte helligdager. Listen over postnummer og poststeder skal kunne oppdateres fra Posten sin offisielle postliste. Sletting av ressurser fra rom/utstyr-delen av applikasjonen og omdefinering av eiere skal også være mulig fra dette verktøyet. Det skal finnes funksjonalitet for endring og sletting av aktiviteter som skal være tilgjengelige i HEVN-J. En liste over ulike avdelinger på skolen skal kunne holdes oppdatert gjennom verktøyet.



Bilde 2-6 Konfigurasjonsprogram for HEVN

”Webapplikasjonen”

Webapplikasjonen skal kunne vise informasjonen i databasen for kortene ”Andre...” og ”Rom/Utstyr”. Liste over ansatte og rom/utstyr skal fungere som på de nevnte kortene. Denne applikasjonen skal kun vise informasjon, og trenger ikke å kunne ta imot datainput fra brukerne. Det er ønskelig å kunne benytte timeplankomponenten fra hovedapplikasjonen som en *Java-applet*, og å få vist innholdet på samme måte som i HEVN-J.

”Databasekonfigurasjonsverktøy”

Dette verktøyet skal kunne brukes til å sette opp databasen for første gang. Det skal kunne generere konfigurasjonsfil for HEVN-J med de riktige databaseparameterne. Konfigurasjonsfilen må kunne brukes i både HEVN-J og i Konfigurasjonsverktøyet. Det er ønske om at det skal kunne settes opp rutiner for autentisering av Linux brukere fra dette verktøyet.

2.4. SOFTWARE DESIGN BEGRENSNINGER

2.4.1. SOFTWARE STANDARDER OG SPRÅK

Vi har valgt å benytte Java som programmeringsspråk for vår applikasjon, dette for å dekke oppgavens grunnleggende krav om en operativsystemuavhengig løsning.

Det er laget en navnekonvensjon som skal benyttes under koding av applikasjonen. Det var naturlig for oss å benytte hovedtrekkene fra HEVN, men vi har valgt å endre denne på noen områder der dette var nødvendig. For å få en gjennomført navnsetting, mente vi det var lettere om vi benyttet et mønster som var kjent fra tidligere.

Følgende hovedregel ble valgt:

Prefikset til en komponent på tre små bokstaver beskriver hvilken type komponent det er snakk om. Deretter en bokstav som angir hvilket kort komponenten befinner seg på, og eventuelt en eller flere store bokstaver til for å spesifisere eventuelle underpanel på dette kortet. Til slutt kommer navnet på komponenten, slik det er naturlig å navnsette denne.

Et eksempel på dette er btnRSlett. Dette er da en knapp (btn), denne befinner seg på hovedkortet Rom/Utstyr (R) og den skal benyttes til å slette ett rom.

Liste over komplett navnekonvensjon er vedlegg [D] til rapporten.

2.4.2. SOFTWARE PAKKER/VERKTØY

IT-tjenesten hos oppdragsgiver krevde at det skulle benyttes MSSQL 2000 som database. Kunden krevde videre at det skulle lages en operativsystemuavhengig løsning. Web-utgaven skal kunne kjøres på nyere nettlesere.

2.4.3. SOFTWARE KOMMUNIKASJONSSTANDARDER OG GRENSESNIITT

HEVN-J krever bruk av TCP/IP. Webapplikasjonen krever bruk av TCP/IP og *HTTP* protokollene.

2.4.4. OPERATIVSYSTEM

Det er ingen krav til spesielle operativsystemer for å kjøre HEVN-J applikasjonen utover det som er beskrevet i kapittel 2.2.6. For å kunne arbeide direkte mot databasen, er det nødvendig å benytte verktøyene ”Enterprise Manager” og ”Query Analyzer” i Microsoft SQL Server.

2.5. ASPEKTER OMKRING LIVSSYKLUS

2.5.1. DOKUMENTASJON

Under skriving av dokumentasjon under hovedprosjektet har vi valgt å bruke følgende standard:

Stil	Font	Størrelse	Attributter
Overskrift a	Arial	18	Fet skrift, store bokstaver
Overskrift b	Arial	16	Fet skrift, store bokstaver
Overskrift c	Arial	14	Fet skrift, store bokstaver
Overskrift d	Arial	12	Fet skrift, store bokstaver
Normal	Times New Roman	12	Ingen

Innleveringen av sluttrapporten skal skje den 23 mai 2002. Det skal leveres statusrapporter i uke 10 og i uke 14. Disse statusrapportene er å finne som vedlegg [I]. Det er også avtalt at det skal foreligge en installasjon/brukerveiledning for databasekonfigurasjonsverktøyet til bruk av IT-tjenesten. Denne brukerveiledningen er lagt ved rapporten som vedlegg [K]

2.5.2. MODUL OG INTEGRASJONSTESTING

Under prosjektarbeidet skal vi benytte XP sine metoder for testing. Det vil si at det vil bli kontinuerlige tester av applikasjonen underveis i arbeidet. HEVN vil bli benyttet til å teste mot. Målsetningen vår er at funksjonaliteten og brukergrensesnittet skal være likt det HEVN har i dag. Der dette ikke er teknisk mulig, vil det være naturlig at Frode Haug som er vår ”*On site customer*”, tar avgjørelser på vegne av de fremtidige brukerne av systemet. Det vil derfor være viktig at vår kundekontakt har et godt forhold til den fremtidige brukergruppen, slik at hans avgjørelser er representative for hele gruppen.

Vi er også forespeilet å utvikle ny funksjonalitet i vår applikasjon og det er viktig at også denne blir testet riktig. Her vil det være naturlig å lage *prototyper* av brukergrensesnittet, samt å ha ukentlige *releaser* til flere enn vår ”*On site customer*”. Her blir det viktigere å få tilbakemelding fra brukergruppen på den nye funksjonaliteten som blir utviklet. Samtidig er det viktig at kvaliteten på testingen holdes oppe slik at ikke den nye funksjonaliteten, som blir lagt til, vil svekke vår applikasjon. Hovedmålet må være å teste vår applikasjon slik at den fungerer som HEVN, for deretter å legge til ny funksjonalitet.

For å sikre at alle systemets deler fungerer sammen, vil det bli gjennomført integrasjonstester etter hvert som systemets deler blir ferdig. Dette vil det være naturlig å gjøre i forbindelse med releaser til en større testgruppe, på omkring 20 aktuelle brukere. Når arbeidet nærmer seg slutten vil vi forsøke å ha ukentlige releaser til denne brukergruppen, hvor feilretting og utbedring av funksjonalitet vil være satsningsområder. Vi ser for oss at denne periode vil vare fra siste del av april og frem til idriftsetting av applikasjonen den 14.06.02.

2.5.3. KONFIGURASJON OG VERSJONSSTYRING

Når klienten settes i drift, skal den være *bakoverkompatibel*, slik at det ikke er et krav om at alle brukerne må sitte med samme versjon av HEVN-J for at systemet skal virke.

Vi har valg å gi applikasjonen versjonsnummer på formen X.X . Det første tallet vil under utviklingsfasen være en. Dette blir først to, når applikasjonen settes i full drift. Tall to angir om det er en utviklingsversjon eller en release. Dette tallet vil stige kontinuerlig frem til vi setter applikasjonen ut i full drift. Her vil partall angi at det er en release og oddetall angi at det er en utviklingsversjon.

2.5.4. KRAV TIL SUPPORT, SERVICE OG VEDLIKEHOLD

Under oppstart av applikasjonen vil enkelte brukere bli satt til å være administratorer. Disse vil ha muligheten til å endre eksisterende brukerinformasjon. Dette vil typisk være avdelingssekretærer slik at de har muligheten til oppdatere informasjon om ansatte som er syke, ute på reise eller av andre grunner ikke kan oppdatere informasjonen om seg selv.

Konfigurasjonsverktøyet vil bli benyttet av systemansvarlige og de som har ansvar for drift av serverne, til å endre konfigurasjonen av systemet. Support, service og vedlikehold av webservere og databaseserver er IT-tjenesten ved HiG sitt ansvar når applikasjonen blir satt i drift.

Support av applikasjonen etter at den er satt i drift vil være Frode Haug i samarbeid med IT-tjenesten sitt ansvar. Det er derfor viktig at disse aktørene får god kjennskap til hvordan å gjøre dette arbeidet. Vi ser for oss at disse personene vil få denne kjennskapen under utviklingsfasen.

Support, service og vedlikehold etter at systemet er satt i drift anses ikke som en del av dette hovedprosjektet, da alle gruppens deltaker vil være ferdig på HiG da dette eventuelt vil skje. Videre arbeid med applikasjonen vil bli regulert av en avtale mellom prosjektgruppen og HiG se vedlegg [J].

2.6. ASPEKTER OMKRING INSTALLASJON

2.6.1. SOFTWARE INSTALLASJON

Systemet installeres på en databaseserver av en systemansvarlig med databasekonfigurasjonsverktøyet. Webapplikasjonen må installeres på en webserver med Java-servlet støtte.

Brukere i HEVN-J må legges inn ved bruk av konfigurasjonsverktøyet. Dette bør gjøres før man installerer klientene. Klientinstallasjonen kan enten gjøres lokalt, eller gjennom en *pakkedistribusjon*.

2.6.2. OVERGANG/OMLEGGING

Det må finnes en måte å overføre personaldata fra den eksisterende databasen, hvor HEVN ligger i dag til databasen hvor HEVN-J skal ligge. Dette må løses enten ved at det lages et program, eller at vi finner et ferdigutviklet program som kan gjøre denne jobben.

Det er planlagt å legge om fra HEVN til HEVN-J i tidsrommet mellom 23. mai og 10 juni. Dette er en forutsetning for at gruppens deltakere skal kunne delta i arbeidet med overgang til vårt nye system.

2.6.3. OPPLÆRING

HEVN applikasjonen har vært i bruk på HiG i fire år. Vi skal etterstrebe å lage vår applikasjon så lik den forrige som mulig. Allikevel vil vår applikasjon få noen vesentlige endringer. Det er i forbindelse med disse endringene at det er et behov for opplæring. Det vil være fordelaktig om oppdragsgiver har en presentasjon av HEVN-J for sine ansatte, hvor ny funksjonalitet blir vist frem og forklart. Dette var en effektiv måte å introdusere den eksisterende versjonen av HEVN på, og bør også nå kunne benyttes.

Vår versjon vil inneholde et oppdatert hjelpekort for hvert enkelt kort i applikasjonen, på samme måte som i HEVN.

For systemansvarlige og drifter av systemet vil det bli gitt ut dokumentasjon for databasekonfigurasjonsverktøyet og for installeringen av HEVN-J.

2.7. PROSJEKTSTYRING OG KVALITETSSIKRING

Arbeidsmetoder for kvalitetssikring, er beskrevet i kapittel 5.2 og 5.3. Valg av systemutviklingsmodell og prosjektstyring er beskrevet i kapittel 7.1.1.

3. ANALYSE OG DESIGN

3.1. HVA SKAL SYSTEMET GJØRE

HEVN-J skal bygge på det eksisterende versjonen av HEVN. Det er derfor naturlig å si litt om dette, ikke minst fordi det vil bli referert til endringer fra gammel og til den nye versjonen.

HEVN er et system som primært skal være en hjelp for å se om en ansatt er tilgjengelig. Det fungerer slik at hver enkelt kan sitte ved sin egen datamaskin, og fylle inn i en timeplan hvor de er til enhver tid. I tillegg kan de sette opp en standardtimeplan hvor det er meningen at man legger inn faste aktiviteter som undervisning, lunsj og liknende. Systemet inneholder også personopplysninger innenfor kategoriene Jobb, Privat, CV og Annet. Disse er av typen adresse, telefon, kompetanse og lignende. HEVN gir også mulighet for meldingsutveksling internt mellom brukere og ved hjelp av e-post.

Det ble også laget en forenklet versjon som studentene kan bruke via Internett. Denne inneholder omtrent samme funksjonalitet som "Andre..." kortet i fullversjonen. Opplysningene som systemet bruker er organisert i en relasjonsdatabase.

Det er foretatt en rekke endringer fra HEVN til HEVN-J, men bortsett fra at e-postfunksjonen er bestemt tatt bort, er ikke endringene større enn at beskrivelsen rett ovenfor også gjelder for HEVN-J. I tillegg kommer en helt ny funksjonalitet for reservering av rom og utstyr. Her kan brukere gå inn i en timeplan, lik den som brukes for å fylle inn aktiviteter, og reservere rom eller utstyr på tider hvor de er ledige.

3.1.1. HEVN-J APPLIKASJONENS GRENSESNIITT

Brukergrensesnittet bygges opp omkring såkalte "tabs", heretter kalt kort. Disse kan sammenlignes med et manuelt kartoteksystem der små "tapper" stikker opp og for eksempel sier hvilken bokstav i alfabetet kortet gjelder. Ved å bruke dette blir skjermplassen utnyttet på en effektiv måte, og mange komponenter får plass på et lite område uten at det blir rotete. Vi bruker seks slike kort som har fått navn etter bruken. Siden bare navnet på underliggende kort blir synlige er det viktig for brukervennligheten at disse navnene er mest mulig selvforklarende. Selv om programmet deles inn i kort, arbeider kortene tett sammen. Ofte vil endringer på ett kort føre til endringer på ett annet kort. Der dette er tilfellet må det være mest mulig opplagt hvor endringer må legges inn.

Et kjennetegn ved HEVN var minst mulig bruk av modi, ingen nedtrekksmenyer og få tastaturnarveier. All funksjonalitet var gjort mest mulig synlig og intuitiv. Dette var en viktig årsak til den popularitet HEVN raskt oppnådde. Det har vært et sterkt ønske fra oppdragsgiver å videreføre denne brukervennligheten, men også å gjøre HEVN-J visuelt mest mulig likt HEVN. I noen få tilfeller hvor det innføres ny og mer komplisert funksjonalitet står disse to kravene litt i konflikt til hverandre. Programmets hovedvindu kan gjøres litt større, men det må ikke overskride grensen på 800x600 piksler med tanke på brukere som bruker eldre skjermer eller av andre grunner bruker denne oppløsningen. Antall kort gir seg nærmest selv ut ifra ønsket om å samle funksjonalitet, som naturlig hører sammen, på ett kort. HEVN har en hjelpfunksjon basert på skjermbilder koblet direkte til hvert enkelt kort. Dette virker enkelt og oversiktlig og blir videreført i HEVN-J.

Det er et mål å gi programmet et helhetlig design fra kort til kort. Viktigst her er at elementer som er felles for kortene gis lik utforming og plassering. Eksempelvis vil det plasseres en knapperekke som står likt plassert øverst på hvert kort. Et annet virkemiddel er standardisering av dialogbokser slik at dialoger og knapper med samme eller liknende funksjon kommer på samme plass hver gang.

3.1.1.1. "JEG ER..."

I HEVN kunne man her bare legge enkeltaktiviteter inn i timeplanen. Repeterende eller faste aktiviteter ble lagt inn i timeplanen på kortet "Oppsett". Timeplanen i HEVN gikk bare 4 uker frem i tid. I realiteten gjaldt det også timeplanen for faste aktiviteter. Denne ble bare kopiert ytterligere en uke frem i tid ved start av ny uke.

I HEVN-J skal timeplanen ikke ha slike tidsbegrensninger. Dermed blir det naturlig å legge inn alle aktiviteter, både enkeltstående og repeterende, i samme timeplan. Faste aktiviteter i HEVN fortsatte å gjelde til de ble slettet. I HEVN-J skal man kunne legge inn repeterende aktiviteter som bare gjelder for et begrenset tidsrom. Timeplanen i HEVN-J skal også gjelde for hele døgnet. Av hensyn til leseligheten skal bare 9 timer vises som i HEVN. Tiden fra 08.00 til 17.00 skal vises som standard.

Fra HEVN skal det føres videre bruken av navigasjonsknapper for å bla frem og tilbake i kalenderen. Da det ikke lenger skal finnes begrensninger på hvor langt man kan flytte seg i tid, legges det til en knapp for å gå direkte tilbake til inneværende uke. Det vil være felter for direkte valg av måned og år.

I utgangspunktet skal alle typer aktiviteter legges inn etter samme mønster. Man merker av et område i timeplanen med musepekeren, dette gir klokkeslettene for start og slutt på aktiviteten. Vil man merke hele dagen, skal dette gjøres ved å trykke på et felt øverst i timeplanen som angir dag og dato. Man skal så få opp en dialogboks hvor man kan angi type aktivitet, tidsrom den skal gjelde for, og om dette skal være en privat aktivitet. Valg av tidsrom, om aktiviteten skal være repeterende og muligheten for å gjøre en aktivitet privat blir nytt i HEVN-J. Type aktivitet skal velges fra en forhåndsdefinert liste mens et eget felt skal brukes for nærmere beskrivelse av aktiviteten. Aktivitetslisten skal kunne suppleres med egne aktivitetstyper. Valg av tidsrom for en repeterende aktivitet skal skje ved å velge startuke / -år og sluttuke / -år. Ved kollisjoner mellom aktiviteter som blir forsøkt lagt inn, og aktiviteter som allerede ligger lagret, skal det varsles om dette før overskriving skjer. Det skal også finnes felter for å justere start- og sluttid. Alternativt skal man kunne dra i kanten på aktiviteten direkte i timeplanen, etter at dialogboksen er lukket.

I HEVN har flytende og faste aktiviteter forskjellig farge, grønn og grå. Helligdager og andre fridager er markert røde. Dagen i dag markeres med gul farge, aktiviteter på nåværende klokkeslett markeres også med gult. Dette videreføres i HEVN-J, men med rom for mindre justeringer etter testing på forskjellige plattformer og skjermtyper. Private aktiviteter skal ha en egen avvikende farge.

HEVN har felter for å velge tid fra **nå** til ett klokkeslett eller fra **nå** og ett antall timer. Ved disse valgene blir starttidspunktet automatisk satt til nærmeste kvarter. Deretter kommer man videre til dialogen for valg og beskrivelse av aktiviteter som beskrevet ovenfor.

HEVN har ett "Inne/Ute modus" som nås fra *traybar* i Windows. Man får opp en egen dialogboks med knapper for "Inne" og "Ute". Ved å trykke på "Ute" vises man som "Ute på ubestemt tid" for andre brukere. Starttiden skal rundes av til nærmeste kvarter, mens sluttiden automatisk økes med ett kvarter helt til man har trykket på "Inne". Denne funksjonaliteten, som er kalt "HEVN Light", skal videreføres i HEVN-J, men det må finnes en alternativ måte å implementere den på i Java.

Kortet skal også ha et felt kalt "Generell kommentar". Dette brukes for kringkasting til alle brukere av meldinger eller kommentarer som ikke er naturlig å tidfeste. HEVN viste dato for siste redigering av dette feltet, dette skal ikke videreføres i den nye versjonen. Helt nytt blir muligheten for å legge inn egne daglige gjøremål. Dette skal gjøres i form av små huskelapper i egne vindu. Disse gjøremålene er personlige, og skal ikke kunne sees av andre brukere av systemet.

For å hindre brukerne i å avslutte HEVN ved en feiltakelse, ble programmet lagt ned i *traybaren* ved lukking av vinduet. Dette er en Windows-spesifikk løsning som ikke er aktuell for Java-versjonen, men da brukerne er tilvent denne funksjonaliteten bør det finnes en løsning hvor programmet ikke avsluttes uten videre.

Utskriftsfunksjonen i HEVN fungerte ikke tilfredstillende for alle brukere, derfor er det et ønske om at denne funksjonen skal videreutvikles og forbedres i vår versjon.

3.1.1.2. "MELDINGER"

Som i HEVN skal det være mulig å sende korte meldinger mellom brukere av HEVN-J. Når man har sendt en melding legges den i en utboks, mens meldinger man mottar havner i en innboks. Hvis mottakeren har markert en aktivitet i timeplanen som foregår når man sender meldingen, skal det være mulighet for sender å få tilbakemelding på dette. Meldingen skal bli markert som "ULEST" i senders utboks, helt til mottakeren har lest den. Meldinger skal kunne slettes fra både inn- og utboks.

Mottaker skal bli varslet om innkommende melding, slik at uleste meldinger ikke blir glemt. For at det skal være lettere å se hvilke meldinger som er nye, skal mottatte meldinger markeres med "NY" i innboksen helt til man har lest den. Tidspunktet for når meldingen ble sendt skal også vises.

Når man sender en melding skal man kunne velge mottaker fra en liste over brukere. Hvis man allerede har markert en melding, skal avsender av denne automatisk dukke opp som mottaker for meldingen.

HEVN hadde en e-postfunksjon. Denne er bestemt fjernet i vår applikasjon.

3.1.1.3. "ANDRE..."

Som i HEVN brukes dette kortet til å hente opplysninger om andre brukere. Kortet skal inneholde en liste over brukere. I listen vil det vises hvem som har installert HEVN-J, om de er innlogget i øyeblikket og hvis ikke; når de sist avsluttet applikasjonen. Listen skal utstyres med en knapperekke med alfabetet for raskt å kunne hoppe til ønsket posisjon i listen.

Ved å velge en person i denne listen, skal man få se vedkommendes timeplan og kan på denne måten finne ut når vedkommende er tilstede. Det skal kun være mulig å avlese andres timeplan for inneværende uke og fremover, ikke bakover i tid.

Helt nytt i HEVN-J blir muligheten for å merke av flere personer i listen over brukere og finne tidspunkter der alle de valgte har felles ledig tid. I tillegg skal det bli innført mulighet for å sende felles melding til flere avmerkede personer.

Dette kortet skal også inneholde en personaliadel med kategoriene "Jobb", "Hjem", "CV" og "Annet". Her vil man finne opplysninger som adresser, telefonnumre, kompetanse og andre aktuelle opplysninger.

På "Andre..." kortet skal det også finnes en søkefunksjon for alle personalopplysninger i databasen. Det skal være mulig å velge om det skal søkes på et helt søkeuttrykk, eller om treff på enkeltord, skal være søkekriteriet.

Det skal være mulig å skrive ut kontaktinformasjon fra "Jobb" og "Privat". Utskriftsalternativene må være fleksible, fra for eksempel bare navn og kontortelefon, til tilnærmet komplett utskrift fra "Jobb" og "Privat". Det må være mulig å skrive ut informasjon for enkeltavdelinger så vel som for hele skolen under ett.

Enkelte brukere kan ha behov for å redigere andres opplysninger og timeplaner. Hvis en ansatt ringer til sentralbordet og melder seg syk, er det ønskelig at sentralbordet skal kunne legge inn opplysninger i vedkommendes timeplan. For å dekke dette behovet skal enkelte brukere kunne gis administratorrettigheter. Disse vil ha rettigheter til å forandre på alle opplysninger som er lagret om den enkelte ansatte unntatt private aktiviteter.

Dette er rettigheter som er tenkt brukt bare i spesielle tilfeller. For å unngå at andres opplysninger blir endret eller slettet ved et uhell skal rettighetene bare kunne brukes ved å gå over i et eget modus som skiller seg ut fra normalsituasjonen.

3.1.1.4. "ROM/UTSTYR"

Dette kortet med tilhørende funksjonalitet blir helt nytt i HEVN-J versjonen. Brukergrensesnittet tenkes lagt tett opp til det som brukes på "Andre..." kortet. Man skal få opp en liste over rom og utstyr isteden for en liste over brukere. Når man velger en ressurs i denne listen skal man få opp ressursens timeplan med eventuelle reserverasjoner. Innleggelsen av reserverasjoner kan skje på forskjellige måter ut ifra hvilke rettigheter brukeren har og begrensningene på rommet.

Ved oppretting av en ressurs, for eksempel ved at et rom legges inn i systemet, skal det være mulig å definere forskjellige nivåer av rettigheter knyttet til rommet. Aktuelle nivåer er eier av ressursen, administratorer og alle. Aktuelle rettigheter er foreta en reserverasjon, endre en reserverasjon og endre eller slette en ressurs.

Om en bruker uten reserverasjonsrettigheter ønsker å foreta en reserverasjon skal det automatisk sendes forespørsel til ressursens eier. Dette tenkes gjort med meldingsfunksjonen. Brukere med reserverasjonsrettigheter må da kunne foreta reserverasjoner på vegne av andre brukere. Ved kollisjoner med allerede foretatte reserverasjoner skal det gis varsel om dette. Timeplanen for

en ressurs skal kunne skrives ut, på samme måten som timeplanen for en person på ”Jeg er...” kortet.

3.1.1.5. ”PERSONALIA”

På dette kortet skal hver bruker legge inn opplysninger om seg selv. Disse opplysningene vises for andre brukere i personaliadelen på ”Andre...” kortet. ”Personalia” kortet blir derfor bygget opp på samme måte med fire underkategorier. De fire kategoriene blir som følger:

- ”Jobb”, opplysningene er her knyttet til den ansattes stilling, kontor og nummer personen kan nåes på.
- ”Privat”, her legges opplysninger som hjemmeadresse og privatnummer.
- ”CV”, denne delen deles opp i utdanning og jobb/erfaring.
- ”Annet”, dette er opplysninger av forskjellig art. Disse kan være hvilke fag man underviser i, hobbyer, tillitsverv og faglige interesser.

3.1.1.6. ”OPPSETT”

”Oppsett” blir betydelig forenklet i forhold til HEVN, i og med at repeterende aktiviteter skal legges inn på kortet ”Jeg er...”. Dermed blir oppsettkortet reservert for innstilling av hvordan programmet skal oppføre seg.

HEVN-J skal kunne varsle om forestående aktiviteter. Det skal gis mulighet for å velge om du vil ha varsling og hvor lang tid i forveien en eventuell varsling skal skje. Intervallet det kan velges innenfor skal være 1-15 minutter. Det skal også kunne velges type varsling, bare meldingsboks, bare lyd eller en kombinasjon av begge. På samme måte skal HEVN-J kunne varsle om innkommende meldinger. Også her skal det kunne velges om du ønsker denne varslingen og hvordan den eventuelt skal skje. Alternativene skal være bare ikon på oppgavelinje, bare lyd, bare meldingsboks på skjerm eller kombinasjoner av disse tre. Det skal her også være mulig å velge bort ”varsling om opptatt mottaker” ved sending av melding. Valg av lydfiler for lydvarsling skal overlates til hver enkelt bruker.

Det skal være mulighet til å velge hvilket kort som vanligvis kommer øverst ved oppstart. Standardinnstilling skal være ”Jeg er...” kortet. Det skal være mulighet for å velge hvor lenge aktiviteter og gjøremål, lagt inn på ”Jeg Er...” kortet, skal ligge i databasen før de slettes automatisk. Det skal gis et antall alternativer, fra 1 uke og opp til 2 år, med 3 måneder som standardverdi.

Listen med aktivitetstyper som dukker opp ved innlegging av aktiviteter på ”Jeg Er...” kortet skal kunne utvides med egendefinerte aktiviteter. Disse kommer bare opp i aktivitetslisten til vedkommende som har lagt dem til.

Det skal lages en webversjon som skal vise informasjonen fra databasen. Det skal derfor være muligheter på ”Oppsett” kortet for å begrense hvilken informasjon som skal vises i webversjonen. Alternativene skal være; både timeplan og personalia, bare en av delene eller ikke noe.

3.1.1.7. "HJELP"

Hjelp skal komme opp i eget vindu og være basert på skjermbilder fra programmet. Disse bildene kobles direkte til de kortene/dialogene hjelpen gjelder. Det skal ikke være nødvendig å måtte navigere i "Hjelp". Dette for å gjøre hjelpefunksjonen så oversiktlig og enkel som mulig.

3.1.2. WEBAPPLIKASJON

Webversjonen er tenkt utformet på samme måte som "Andre..." kortet. Det skal være mulig å velge ansatte på skolen fra en liste, for deretter å ha mulighet til å se en timeplan med ukens eller kommende ukers aktiviteter. I tillegg skal det være mulig å lese informasjonen som er lagret på den valgte personens personalia.

I tillegg til å vise informasjon om ansatte ved skolen, skal man kunne se timeplanen for et rom eller utstyr. Det er også her tenkt benyttet et liste over innlagt ressurser på samme måte som det blir gjort for ansatte. Det skal ikke være mulig å reservere rom eller utstyr via webapplikasjonen.

For studentene ved skolen er det denne applikasjonen som gir mulighet til å se når skolens ansatte har ledig tid. Denne informasjonen kan være interessant ved planlegging av veiledning eller andre møter med lærerne. Det er derfor viktig at løsningen kan benyttes utenfor skolens intern-nettverk. Løsningen må heller ikke stille for spesifikke krav til nettlesere, da brukerne av denne applikasjonen ikke nødvendigvis bruker samme nettlesere som benyttes på skolen.

Hvis de ansatte av ulike grunner ikke ønsker å vise informasjon om seg selv i denne nettversjonen, skal det finnes mulighet for å begrense dette på kortet "Oppsett". Denne begrensningmuligheten skal ikke finnes for rom og utstyr.

Under utviklingen av applikasjonen, må det taes hensyn til at mange brukere av denne applikasjonen vil benytte oppringt tilkobling til Internett. Det bør derfor legges vekt på å begrense hvor mye data som må overføres ved lasting av siden.

3.1.3. DATABASE

IT-tjenesten ved HiG har satt som krav, at det skal benyttes en MSSQL database. Vi ser for oss at databasen for HEVN-J blir tilnærmet lik den som blir brukt av HEVN. Noen få attributter må fjernes, mens noen tabeller og attributter må legges til. Det er ikke planer om å legge noe funksjonalitet på databasen, da vi ser det som like enkelt å legge dette på klienten. Det bør være en målsetning at databasen lages i 3. normalform.

3.1.4. KONFIGURASJONSPROGRAM

Det skal lages et konfigurasjonsverktøy for HEVN-J. Konfigurasjonsverktøyet skal gjøre det enkelt å konfigurere systemspesifikke parametere og innstillinger, som er felles for alle brukere.

I dette verktøyet vil det finnes mulighet for å opprette alle databasetabeller på nytt. Opprettelse og sletting av brukere skal skje her. Det vil også finnes funksjonalitet for oppsett av administratorrettigheter i dette verktøyet. Telefonprefiks for HEVN-J applikasjonen skal kunne settes, samt passord for bruk ved start av HEVN-J med "login passord.

Det skal kunne legges til, og fjernes avdelinger fra listen over eksisterende avdelinger. Man skal også kunne legge til, og fjerne faste aktiviteter for HEVN-J.

Verktøyet skal også kunne legge inn faste helligdager i databasen for flere år frem i tid. I tillegg til disse faste helligdagene, skal det være mulighet for å legge inn egne spesielle fridager.

Komplett liste over postnummer og tilhørende poststeder, som skal benyttes i applikasjonen, skal kunne editeres fra dette verktøyet.

For rom og utstyr delen av applikasjonen skal det finnes funksjonalitet i konfigurasjonsverktøyet for sletting av rom eller utstyr, og omdefinering av eiere for disse ressursene.

Det skal være mulig å slette aktiviteter, reserverasjoner og gjøremål i databasen, som er eldre enn to år.

3.1.5. DATABASEKONFIGURASJONSVERKTØY

Verktøyet skal kunne generere en Microsoft SQL database med relevante tabeller og views fra bunnen av.

Det skal være mulig å angi om Linux-brukere skal autentisere mot en spesifisert pop3 server.

En konfigurasjonsfil skal genereres for HEVN-J og Konfigurasjonsverktøyet. Denne filen skal inneholde nødvendige databasespesifikke parametere og eventuelt om autentisering av Linux-brukere er ønsket.

3.2. HVORDAN LØSTE VI DET

3.2.1. ARKITEKTUR

Det finnes to vanlige modeller for hvordan et klient/tjener løsning kan lages. Den ene er å bruke tynn klient og tykk server. Mens den andre er tykk klient og tynn server.

Ved tynn klient og tykk server stilles det ikke så høye maskinkrav til klientene. Det er også enklere å oppdatere en server, enn å oppdatere alle klientene. Det trengs en kraftig server, som klarer å håndtere mange brukere samtidig. Består applikasjonen av mye grafikk, vil det derfor kreves en høy båndbredde til serveren for at det skal bli et bra grafisk brukergrensesnitt hos klientene. Dette kan eventuelt løses ved å legge presentasjonslaget på klientene.

For tykke klienter og tynn server er maskinkravet til serveren mindre. Håndtering av grafikk vil også gå raskere, noe som vil føre til et bedre brukergrensesnitt

Da HiG ikke har en terminal server, gav vårt valg seg selv. Vi måtte bruke modellen med en tykk klient og tynn server, hvor hele applikasjonen ligger på klienten og databasen ligger på serveren.

3.2.2. HEVN-J APLLIKASJONENS GRENSESNIITT

Man bør ha lest kapittel 3.1.1 før man leser dette kapittelet. Det er forutsatt at man vet hva som skal kunne gjøres på de forskjellige kortene.

Vi valgte å benytte Java som utviklingsspråk, se kapittel 4.1.1 for begrunnelse av dette valget. Java er det eneste språket som vi har kjennskap til, som kan brukes til å utvikle plattformuavhengige løsninger. Med designfunksjonen i Borland JBuilder 6.0 kan man dessuten konsentrere seg om det visuelle designet og se resultatene fortløpende uten forsinkelse.

Ett annet viktig moment for valg av Borland JBuilder som utviklingsverktøy var å få tilgang til Borlands *layoutmanagers*. Bruken av *XY-layout* gjorde det langt enklere å kopiere utseendet til HEVN enn det ville ha vært med fritt tilgjengelige layoutmanagers. Bruken av *XY-layout* gir full kontroll med størrelse på og plassering av komponenter.

Det er forsøkt å bruke samme fonter i HEVN-J som i HEVN. Dette fungerer fint for Windows og andre operativsystemer som støtter *True Type Fonter* forutsatt at fontene finnes på maskinen HEVN-J kjøres. De fleste Linux-installasjoner støtter foreløpig ikke True Type Fonter. Dette har gitt noen problemer med at de medfølgende fontene i Java tar litt større plass enn de vi i utgangspunktet designet brukergrensesnittet rundt.

Da utviklingen ble gjort i Windows, løste vi dette problemet ved å sette av ekstra plass på tekstfelter og knapper. Ved å velge en slik enkel løsning, var vi sikre på at teksten i applikasjonen ville bli vist riktig også på Linux. Vi har også erfart at ikke alle operativsystemer støtter de samme størrelser og fargeoppløsninger på pekere. Her har vi funnet generelle løsninger som tester på funksjonalitet i operativsystemet heller enn type operativsystem.

Knappene i HEVN-J er gjort større enn i HEVN. Det har blitt brukt ikoner i større grad enn i HEVN. Vi var klar over at slike ikoner, kan gi forskjellige assosiasjoner, men vi mener ikonene er illustrerende for knappenes funksjon. Et viktig hjelpemiddel for å gjøre bruken av programmet intuitivt, var å markere knapper og felter som tydelig inaktive i situasjoner hvor de ikke skal benyttes.

Da det er lagt til en del funksjonalitet på flere av de opprinnelige kortene, har det blitt trangere mellom komponentene på noen kort. Vi har lagt vekt på å gjøre beslutningstrær i grensesnittet så grunne som mulig. I noen tilfeller må brukeren gjennom tre nivåer med dialogbokser for å utføre en operasjon. Men i de fleste tilfeller er det nok med ett eller to nivåer, inklusive rene kvitteringsbokser med enkle ja eller nei beslutninger.

Ved lukking av applikasjonen, avsluttes ikke programmet. I stedet kommer det opp en dialogboks hvor brukeren blir spurt om han virkelig vil avslutte programmet. Svarer brukeren nei vil programmet bli minimert til oppgavelinjen.

3.2.2.1. "JEG ER..."

Dette kortet består i hovedsak av en timeplan. Det er den samme timeplanen som brukes på kortene "Andre..." og "Rom/Utstyr", selv om funksjonaliteten er litt ulik. Timeplanen er selve hjertet i HEVN-J, og det er denne komponenten mesteparten av applikasjonens funksjonalitet er bygd opp rundt. Vi fant ikke noen ferdiglaget komponent med muligheter for innlegging og redigering av aktiviteter. Derfor ble en egen komponent utviklet. Denne komponenten er beskrevet mer i detalj i kapittel 4.4.1.

Timeplanen har fått et utseende som er nært identisk med HEVN. Dagene fra mandag til fredag og tiden 08.00 til 17.00 vises. Klokkeslett før 08.00 og etter 17.00 vises ved hjelp av *scrollbar*. Muligheten for å ta med lørdag og søndag i timeplanen ble diskutert underveis, men ble forkastet på bakgrunn av applikasjonens forventede bruksområde se kapittel 1.2. Tidsoppløsningen er som i HEVN 15 minutter. Dette er satt til minste oppløsning, ikke bare for tegning, men også for valg av tidspunkter i ulike dialogbokser.

Funksjonalitet for å angi opptatt fra nå i ett antall timer eller til ett klokkeslett, er gradert i halve timer. Starttidspunktet blir satt til nærmeste kvarter i forhold til nåtid. Disse feltene er i HEVN-J fjernet fra hovedpanelet øverst på kortet, hvor de lå på HEVN. I vår versjon kommer de opp i en egen dialogboks ved å trykke på en "Ute"-knapp.

Utefunksjonaliteten kan kun nåes fra en knapp på "Jeg er..." kortet. Bakgrunnen for dette er at det å legge ikoner/funksjonalitet i traybaren ikke er mulig ved en plattformuavhengig løsning. I dialogboksen som kommer opp, kan brukeren velge ute. Da vil det komme opp ett nytt rødt dialogvindu, med teksten "UTE". Applikasjonen lager nå en aktivitet på timeplanen for brukeren, som automatisk legger til ett nytt kvarter, helt til brukeren deaktiverer denne funksjonaliteten ved å trykke på knappen merket "Inne igjen". Så lenge brukeren har valgt "UTE", vil andre brukere av systemet på "Andre..." kortet se en aktivitet med teksten "Ute på ubestemt tid". Denne aktiviteten vil forsvinne så snart brukeren gir beskjed om at han er inne igjen.

Feltene for direkte valg av måned og år gir anledning til å gå 3 år frem og 2 år tilbake i tid. Navigasjonsknappene for forrige og neste uke har ingen slike begrensinger. Ved direkte hopp til en måned vises første uke med en virkedag i denne måneden.

Aktiviteter vises i timeplanen på samme måte som i HEVN med noen mindre endringer. Når en aktivitet går over hele dagen, blir denne aktivitetens tekst skrevet klokken 08.00. Dette gjelder også for aktiviteter som starter før klokken 08.00 og slutter etter klokken 09.30, men disse får i tillegg starttiden påskrevet klokken 08.00. Bakgrunnen for at dette ble valgt, var at teksten skulle være synlig i den delen av timeplanen som kommer opp når man går til kortet ”Jeg er...”, ”Andre...” eller ”Rom/Utstyr”. Private aktiviteter har fått en blålig farge som klart skiller seg fra andre farger brukt i timeplanen.

Det oppstod et behov for å kunne håndtere konflikterende aktiviteter. I samråd med ”On site customer” kom vi frem til en løsning for dette problemet. Konflikt mellom aktiviteter som gjelder en uke, ble løst etter samme mønster som i HEVN. Brukeren får valget mellom å skrive over, eller å avbryte innleggingen for hele uken.

Hvis man velger at aktiviteten skal gjelde over flere uker, har man tre valgmuligheter. Man kan overskrive eksisterende aktiviteter (”Ja”), legge inn den nye uten å overskrive eksisterende (”Legg bak”), eller å la være å legge inn aktiviteten denne dagen (”Nei”). Dette vil gjenta seg for hver dag med konflikter.

Etter at denne løsningsmetoden ble brukt en stund av testgruppen, kom det ønske om mulighet for å velge ja til alt, og nei til alt. Dette var basert på tilfellene hvor det oppstod mange konflikter når en aktivitet ble registrert over lang tid. Det kunne da bli konflikter på mange dager, og det ville ta lang tid før brukeren ble ferdig med å behandle disse. Dette problemet ble løst ved å lage funksjonalitet for ”Ja til alt” og ”Nei til alt”. ”Ja til alt” skriver over alle konflikterende aktiviteter, mens ”Nei til alt” legger inn aktiviteten der det er plass.

Det er laget helt ny funksjonalitet for notering av egne gjøremål ved at det er plassert knapper med notatblokkikon i feltene for visning av dag og dato. Disse bryter litt med standarden for utseende og plassering av knapper ellers i programmet, men gir en brukervennlig løsning ved at knappene er koblet direkte til aktuell dag. Et resultat av den valgte løsningen er at slike notater ikke kan gjøres for lørdager og søndager, men det samme gjelder for aktiviteter og gjenspeiler hvordan programmet er tenkt brukt. Notater kan legges inn på helligdager som vises i timeplanen. Ved å trykke på knappen kommer det opp en egen boks for å skrive inn tekst. Boksen kommer i utgangspunktet opp i nedre kant av den aktuelle dagen, men kan flyttes og endres av brukeren, hvis dette er ønskelig. Det er bare en slik boks per dag. Ikonet på denne knappen endrer seg hvis det er lagret informasjon på en dag. Dette er gjort for at brukeren lettere skal kunne se hvor han har lagret informasjon om daglige gjøremål.

Hvem-Er-hVor-Når

Jeg er... Meldinger Andre... Rom/Utstyr Personalia Oppsett

Måned Mai År 02 Uke 19 Forrige Nærværende Neste Skriv ut Ute Hjelp

	Mandag 6.5.2002	Tirsdag 7.5.2002	Onsdag 8.5.2002	Torsdag 9.5.2002	Fredag 10.5.2002
08.00			Reise (04:00) Beskrivelsen vises 08.00 med starttid når en aktivitet begynner tidligere og varer utover 09.30.	Kristi himmelfartsdag	Avspasering Denne aktiviteten foregår hele dagen og starttid vises ikke selv om den starter før 08.00.
09.00	Møte	Møte Dette er en engangsaktivitet. Den blir grønn.			
10.00					
11.00	Kurs/Seminar Denne er merket for redigering av tid eller beskrivelse.	Ikke forstyr! Dette er en privat aktivitet, merk fargen! Beskrivelsen vises bare for deg selv.			
12.00					
13.00			Undervisning Denne er repeterende aktivitet. Den blir grå.		
14.00		Ute en tur			
15.00	Lager hjelpbilder til HEVN-J. Aktivitet som pågår akkurat nå blir gul.		Tid kan endres med klikk og dra.		
16.00					

Generell kommentar: For kringkasting til alle av "meldinger" det ikke er naturlig å tidfeste.

Bilde 3-1 "Jeg er..." kortet i HEVN-J

3.2.2.2. "MELDINGER"

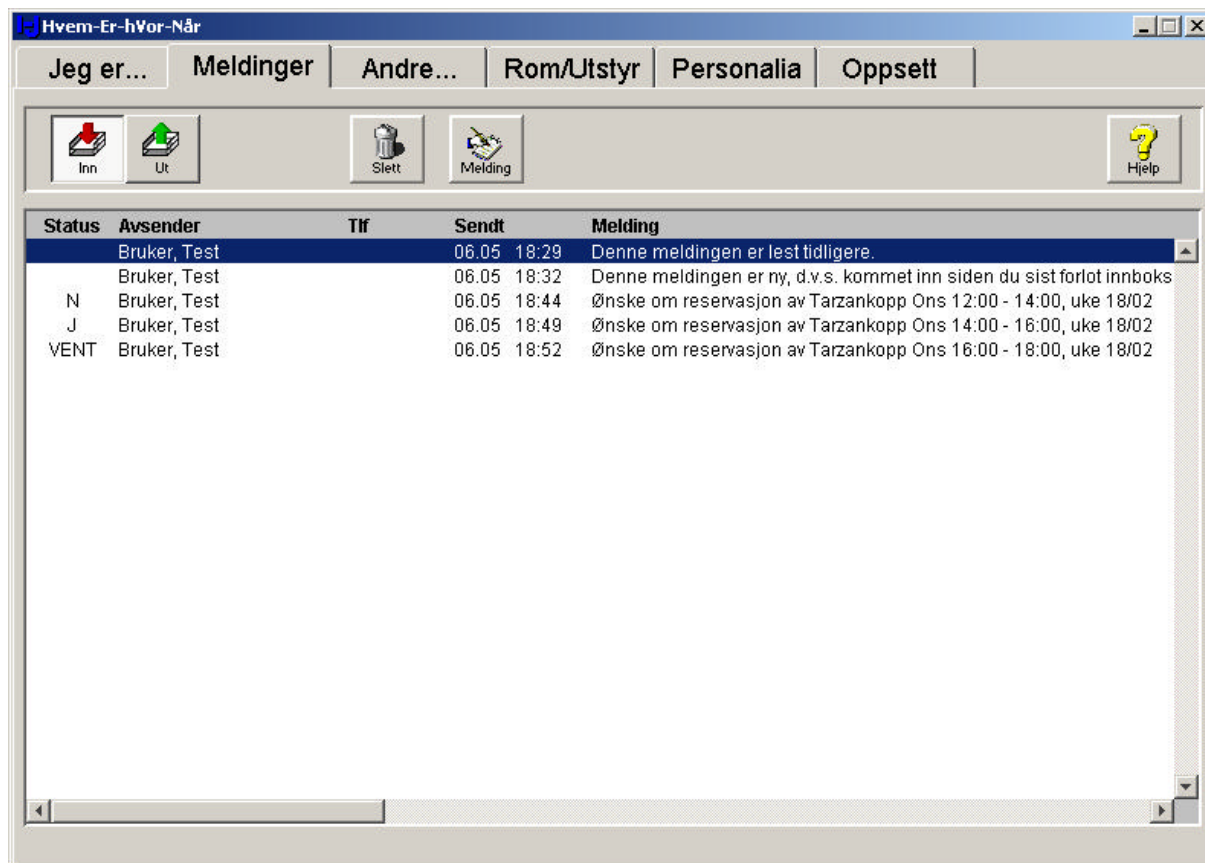
Dette kortet viser bare en liste med mottatte eller sendte meldinger avhengig av om innboks eller utboks er valgt. Felles for begge listene er felter for status, sender/mottaker, intern-telefon, tidspunkt meldingen er sendt og selve meldingen. Meldingens lengde kan maksimalt være 150 tegn.

Listen med feltene Status, Avsender/mottaker, Tlf, Sendt og Melding blir generert av en egendefinert JTable. Alternativet var å bruke en ferdiglaget databasekomponent for å gjøre oppgaven enklere, men disse komponentene virket å være for generelle til vårt bruk.

Statusfeltet viser "NY" i innboksen og "ULEST" i utboksen, inntill mottaker har lest meldingen. Skifte av status fra "NY"/"ULEST" skjer ved at man forlater listen ved å gå til annet kort eller bare ved å skifte mellom ut og innboks.

I forbindelse med automatisk genererte meldinger i "Rom/Utstyr" har det kommet nye funksjoner for statusfeltet. Visningen i statusfeltet gjenspeiler mottakers respons på reservasjonsønsker. "J", "N" eller "Vent". I motsetning til "NY" og "ULEST" forsvinner disse naturligvis ikke.

Ved å trykke på knappen "Melding" kommer det opp en boks med nedtrekksfelt for valg av mottaker og et tekstfelt for å skrive meldingen. Om en mottaker eller avsender er valgt i meldingslisten vil denne personens navn komme opp i nedtrekksfeltet for valg av mottaker. Hvis ingen melding er markert, vil første person i listen over systemets brukere være markert som foreslått mottaker. Knappen "Slett" sletter merket melding. Meldinger kan bare slettes manuelt.



Bilde 3-2 "Melding" kortet i HEVN-J

3.2.2.3. "ANDRE..."

På "Andre..." kortet, kan man se timeplanen og personalia for andre brukere. Brukeren man ønsker informasjon om velger man fra en liste med alle brukerne. Denne listen benyttes også til å vise personer funnet ved bruk av søkefunksjonen. For å vise hvem som har installert HEVN-J og om vedkommende er pålogget for øyeblikket, er det benyttet samme løsningen som i HEVN. Det vises henholdsvis en I (for Installert) og en O (for Online) foran navnet i listen. Dato og klokkeslett for når en bruker sist logget ut, vises under listen.

Timeplanen på dette kortet er en forminskert utgave av timeplanen på "Jeg er..." kortet. Når en person er valgt i listen vises vedkommendes timeplan. Private aktiviteter vises som en vanlig aktivitet, med teksten "(Privat)" uten ytterligere opplysninger.

Funksjonen for å finne felles ledig tid for flere personer er løst med en knapp kalt "Felles tid" som bare blir aktivisert når mer enn en person er valgt i listen. For å velge flere personer i listen, benyttes standard metode for å markere flere linjer i en liste. (Holde Ctrl-tasten nede og markere ønskede brukere i listen med mus eller piltaster). Ved å trykke på "Felles tid" knappen får man så opp en felles timeplan med alle brukernes aktiviteter markert som opptatt. Det ble valgt å markere opptatt tid, da ledig tid ellers i applikasjonen ikke er markert. For at denne funksjonaliteten skal virke som forutsatt, betinger det at brukerne ikke fyller opp timeplanen sin med aktiviteter av typen "Er på kontoret".

"Andre..." kortet har også en knapp for å sende meldinger. Hvis bare en bruker er markert i listen kommer den samme dialogboksen som på meldingskortet. Om derimot flere brukere er avmerket, kommer det opp en dialogboks uten felt for å velge mottaker, men med teksten: "Skriv melding til de merkede i listen". Alle merkede brukere vil da motta den samme meldingen.

Muligheten til å velge flere personer samtidig finnes bare når man har valgt å se timeplanen. Det er ikke mulig å se personalia eller søke når flere personer er markert. Dette er valgt for at ikke brukerne skal tro at søk i utgangspunktet skal kunne begrenses til noen forhåndsvalgte personer. Søkefunksjonen går gjennom alle personaldata i databasen. I dialogboksen som dukker opp ved å trykke på "Søk-knappen", er det brukt radioknapper for å velge om et søk må treffe på alle søkeordene eller bare ett av dem. Ønsket søkeord skrives inn i et tekstfelt. Ved treff vises personene, hvor søkeordene ble funnet, i listen over brukere. Søket gjøres med en vanlig SQL-spørring i programmet. Dette gir noe mer nettrafikk enn å bruke lagrede prosedyrer på serveren, men vi ser ikke dette som et problem med det lave antall søk som forventes.

Det ble valgt å bruke kort som visning av personaldata. Navnene på kortene tilsvarer kategoriene som beskrevet i kapittel 3.1.1.3., "Jobb", "Privat", "CV" og "Annet". For nærmere beskrivelse av innhold på de enkelte kortene vises det til kapittel 3.2.2.5. Personen det skal vises informasjon om, velges i listen på samme måten som for timeplanen.

Ved å trykke på knappen "Skriv ut" vises en dialogboks der man kan velge alle avdelinger, enkelt avdelinger eller en telefonliste. For å velge en enkelt avdelinger benyttes en *nedtrekksliste*. Det er benyttet ett sett med 12 avkrysningsbokser for å velge hvilke felter som skal være med på utskriften. Mulige valg er "Etternavn", "Fornavn", "Tittel", "Stilling", "E-post"(på jobb), "Adresse", "Jobb tlf", "Privat tlf", "Mobil tlf", "Fax", "Rom" og "Avdeling". Det er automatisk krysset av noen alternativer, men disse kan enkelt endres hvis ønskelig.

Utskriften tilpasses automatisk etter antall felter som er valgt, men det er ikke anbefalt å velge mer enn 5 felter på grunn av arkets bredde. Før noe skrives ut, får man opp en forhåndsvisning av utskriften. Det finnes en mulighet for å avbryte, hvis man allikevel ikke er fornøyd med de valgte feltene.

”Andre...” kortet har en knapp merket ”Admin” som bare er synlig for brukere med administratorrettigheter. Denne knappen blir merket rød, når den blir aktivisert. Ved bruk av denne gis det mulighet for å endre den valgte brukers personaldata. Det er nå mulig å endre timeplanen for valgt bruker. Allikevel kan man ikke endre eller se informasjon om private aktiviteter. Eneste mulighet for å fjerne private aktiviteter er å skrive over den ved å opprette en aktivitet over en hel dag.

Nederst på dette kortet er feltet for generell kommentar som beskrevet på ”Jeg er...”.

The screenshot shows the 'Andre...' tab in the HEVN-J software. The interface includes a menu bar with 'Jeg er...', 'Meldinger', 'Andre...', 'Rom/Utstyr', 'Personalia', and 'Oppsett'. Below the menu bar is a toolbar with icons for 'Timeplan', 'Personalia', 'Melding', 'Felles tid', 'Søk', 'Skriv ut', and 'Admin' (highlighted in red). The main area displays a calendar grid for 'Uke 19' (May 6-10, 2002). A list of users is shown on the left, with 'Bruker, Test' selected. The calendar grid shows activities for 'Bruker, Test' on Monday (08:00-16:00) and Tuesday (08:00-10:00). A 'Generell kommentar:' field is at the bottom.

	Mandag 6.5.2002	Tirsdag 7.5.2002	Onsdag 8.5.2002	Torsdag 9.5.2002	Fredag 10.5.2002
08.00		Fravær Kommer senere, uhell med bilen.		Kristi himmelfart..	
09.00					
10.00					
11.00					
12.00					
13.00					
14.00	Kurs/Semin.. Seminar om datasikkerhet, Eureka.				
15.00					
16.00					

Bilde 3-3 "Andre..." kortet i HEVN-J

3.2.2.4. "ROM/UTSTYR"

I større grad enn med de andre kortene startet vi med blanke ark ved utformingen av dette kortet. Både krav til funksjonalitet og brukergrensesnitt er blitt til under prosjektets gang og er endret underveis. Dette er en arbeidsmetode som er anbefalt av XP, og vi har under utviklingen av dette kortet erfart hvor viktig det er å ikke låse seg til en løsningsmetode. De siste endringene på dette kortet har kommet tett opp mot deadline for hele prosjektet, løsningen for dette kortet har likevel blitt bra.

Som planlagt ble grensesnittet tilnærmet identisk med timeplanløsningen for "Andre..." kortet. Ved siden av timeplanen er det en liste for rom og en annen for utstyr. To lister ble valgt, istedenfor en enkelt liste med et statusfelt for å skille mellom rom og utstyr. Under listene er det plassert tre knapper. Man kan lage en ny, endre en eksisterende eller slette en ressurs. Knappen "Slett" er inaktiv for brukere som ikke har rettigheter til å slette en gitt ressurs. Mulighet for å endre en ressurs, avhenger av hvilke rettigheter som er satt opp for rommet eller utstyret. Det er allikevel gitt muligheter for at brukerne skal kunne se hvilke rettigheter en ressurs har. Begrensningsmulighetene for en ressurs, er beskrevet i kapittel 3.1.1.4 .

Det er to forskjellige måter å legge inn nye reserverasjoner på. Har man rettigheter til å legge inn nye reserverasjoner, får man opp en dialogboks hvor man velger reservatør, justerer tidsrom, angir uke intervall og skriver eventuelt en kommentar. En melding vil da gå til reservatøren, som en bekreftelse på reserverasjonen.

Hvis man ikke har rettigheter til å foreta en ny reserverasjon vil man få opp en dialogboks med en liste over hvilke personer som har slike rettigheter. Her kan man justere tidspunkt, angi et ukeintervall, markere for at reserverasjonen kun gjelder annen hver uke og en begrunnelse for hvorfor man vil reservere. En reserverasjonsmelding vil da gå til den markerte brukeren. Mottakeren av reserverasjonsmeldingen kan da klikke på denne i sin innboksen. Det kommer da opp en dialogboks med informasjon om den ønskede reserverasjonen. Han kan da godkjenne, avvise eller legge reserverasjonen på vent.

Det er mulig å overta en annen reserverasjon. Dette skjer ved at man klikker på ønsket reserverasjon. I en dialogboks må man så svare ja eller nei på om man vil overta reserverasjonen. Ved å svare ja kommer det opp en dialogboks, med alle brukerne som har rettighet til å endre reserverasjonen. Hvis det er flere enn bare eieren, som har slike rettigheter, vil reservatøren stå som mottaker av meldingen. Før man kan sende en overtakelsesmelding, må man skrive en begrunnelse. Mottakeren av meldingen vil da få en overtakelsesmelding. Ved å klikke på denne, vil det komme opp en dialogboks med informasjon om reserverasjonen, samt begrunnelsen for overtakelsen.

Hvis en person ikke har tilgang til å endre sin egen reserverasjon, vil han få opp en dialogboks hvor ønsket endring og eventuelt begrunnelsen må angis. Det vil da gå en melding til eieren av rommet. Rommets eier må da selv gå inn og endre dette på "Rom/Utstyr" kortet.

Ved endring og sletting av en reserverasjon, vil det alltid gå en melding til reservatøren, med opplysninger om hva som har skjedd og hvilken reserverasjon dette gjelder.

Hvem-Er-hVor-Når

Jeg er... Meldinger Andre... Rom/Utstyr Personalia Oppsett

Måned Mai År 02

Uke 19
 Forrige Nåværende Neste Skriv ut Hjelp

Rom (ny, endre reservasjon)

- A - Kantina (Eier, Eier)
- A-112 (Eier, Admin)
- A-114A: Ivar Farups kontor (Eier, Eier)
- A-115 (Admin, Admin)
- A-116 (Eier, Eier)
- AAAA (Eier, Eier)
- B-209 (Alle, Admin)
- B-320-Møterom (Eier, Eier)
- B338-Test (Eier, Eier)
- D2 (Eier, Admin)
- grupperom 233 (Eier, Eier)
- K-105 (Admin, Admin)

Utstyr (ny, endre reservasjon)

- 19" Skjerm (Eier, Eier)
- LaserPrinter (Eier, Admin)
- Slegge (Alle, Eier)
- Tarzanopp (Eier, Eier)
- ØL jekk (Alle, Eier)

	Mandag 6.5.2002	Tirsdag 7.5.2002	Onsdag 8.5.2002	Torsdag 9.5.2002	Fredag 10.5.2002
08.00				Kristi himmelfar..	
09.00					
10.00			Bruker, Test		
11.00					
12.00					
13.00					
14.00					
15.00					
16.00					

Nytt Endre Slett

Bilde 3-4 "Rom/Utstyr" kortet i HEVN-J

3.2.2.5. "PERSONALIA"

"Personalia" kortet er delt opp i fire underdeler. Disse er "Jobb", "Privat", "CV" og "Annet", som velges med tilsvarende knapper. Denne løsningen ble valgt istedenfor kort først og fremst for å avvike minst mulig fra HEVN, men også fordi denne løsningen gir mest plass til informasjon, uten å fjerne det øverste panelet som er felles for alle kort. Kortet har en rekke tekstfelder samt en nedtrekksliste for å velge avdeling. Feltet for poststed kan ikke redigeres, innholdet her hentes fra databasen etter at et gyldig postnummer er skrevet inn i postnummerfeltet.

Rent funksjonelt gjør ikke dette kortet annet enn å lese data fra databasen og legge data inn i databasen når "Lagre" knappen trykkes. Denne blir bare aktiv når minst ett felt er endret.

The screenshot shows a web application window titled "Hvem-Er-hvor-Når". The interface has a top navigation bar with tabs: "Jeg er...", "Meldinger", "Andre...", "Rom/Utstyr", "Personalia" (selected), and "Oppsett". Below the navigation bar are four icons for "Jobb", "Privat", "CV", and "Annet", and two buttons for "Lagre" and "Hjelp". The main form area contains several input fields and a dropdown menu:

- Fornavn**: Test
- Etternavn**: Bruker
- Tittel**: Student
- Stilling**: avslappet
- Telefon**: 611 35305
- Fax**: 611 []
- Rom**: 004/1
- Avdeling**: AT (dropdown menu)
- E-post**: tes_bruk@hig.no
- Hjemmeside**: http://studweb.hig.no/tes_bruk/

Bilde 3-5 "Personalia" kortet i HEVN-J

3.2.2.6. "OPPSETT"

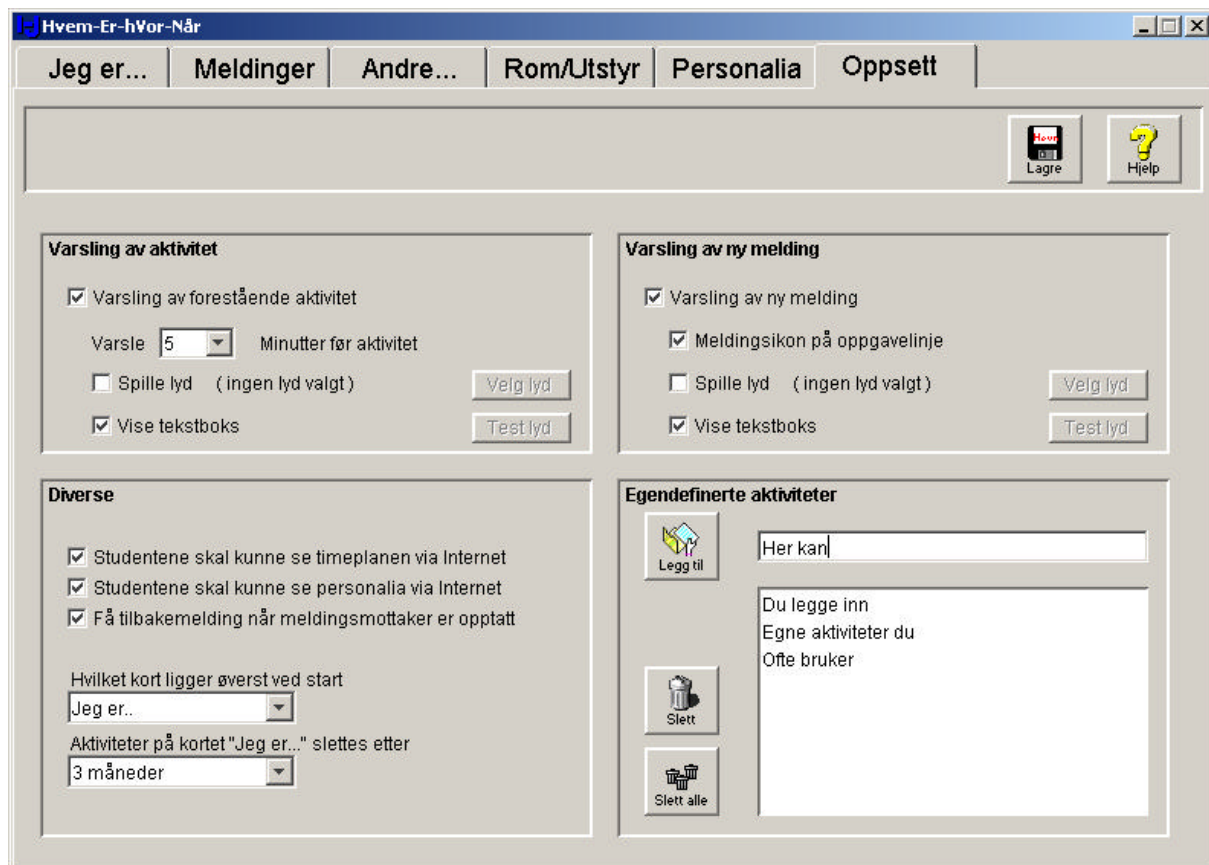
Dette kortet har blitt mye forenklet i forhold til HEVN. Timeplanen på dette kortet er fjernet, og det er nå kun mulig å sette personlige innstillinger for brukeren på dette kortet.

Ved hjelp av avkrysningsbokser, kan brukeren velge, om og på hvilken måte han skal bli varslet ved forestående aktiviteter og innkomne meldinger. Hvis brukeren velger å bli varslet ved forestående aktiviteter, må han angi hvor lenge før aktiviteten varslingen skal finne sted. Det valgbare tidsrommet går fra ett til femten minutter. Brukeren kan velge om han vil bli varslet med lyd eller med en dialogboks. Det er mulig å velge begge alternativene samtidig. Ved valg av lyd som varslig, kommer operativsystemets dialogboks for filvalg opp. Her ligger det et utvalg standardlyder, men brukeren kan fritt velge mellom andre lyder. Begrensning på lydene er maksimalt 22 kHz samplingsfrekvens og anbefalt størrelse på disse er opp til to megabyte. Det er mulig å velge varsling av innkomne meldinger, ved hjelp av et blinkende ikon på oppgavelinja.

Brukeren har mulighet til å begrense hvilken type informasjon som skal vises i webapplikasjonen. Det er mulig å velge at både timeplanen og personalia ikke skal vises. Videre er det mulig å velge hvilket kort som skal komme opp øverst ved oppstart. Det er mulig å velge hvor lenge aktiviteter og gjøremål lagt inn på "Jeg er..." kortet skal ligge lagret i databasen før de slettes automatisk. Maksimaltiden for lagring av aktiviteter og gjøremål er 2 år. Det finnes flere mulige valg, fra 1 uke og opp til maksimaltiden, med 3 måneder som standardverdi.

Brukeren kan legge inn egne aktiviteter som kommer opp i listen over aktiviteter på "Jeg er..." kortet. Det er mulig for brukeren å endre denne egendefinerte listen over aktiviteter.

Det er laget funksjonalitet slik at brukere av Linux kan angi hvilket e-postprogram og nettleser som skal startes fra applikasjonen, når det klikkes på felter for e-post og websider på "Andre..." kortet.



Bilde 3-6 "Oppsett" kortet i HEVN-J

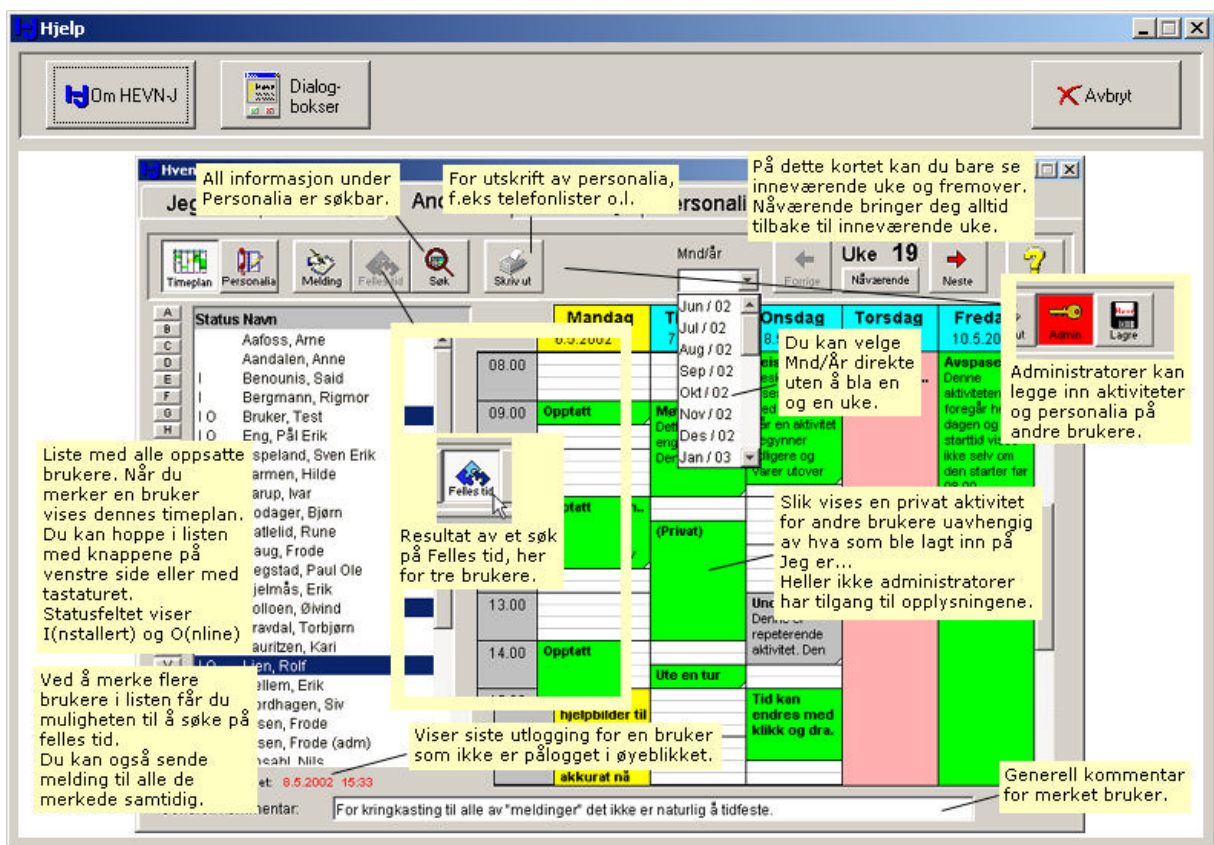
3.2.2.7. "HJELP"

Hjelpesfunksjonaliteten i vår applikasjon er mye lik hjelpen i HEVN. Hvert hovedkort har en egen hjelpeknapp. Ved klikk på denne kommer hjelpen for kortet man står på opp i et eget skjermbilde.

Istedenfor å ha hjelpeknapper i applikasjonens dialogbokser, er alle dialogbokser for et kort samlet i hjelpen. Ved å trykke på knappen for dialogbokser, vil alle aktuelle dialoger med forklaring for dette kortet komme opp.

Selve hjelpen er laget med et skjermbilde av aktuelt kort som bakgrunn. På dette bildet er det "limt" på gule lapper som forklarer ulik funksjonalitet. Ulike tilstander kortet kan være i, er også forklart. Ved å bruke .png bilder, er tiden det tar å laste bildene redusert, dette har ført til at hjelpen kommer raskt opp, uten ventetid for brukeren.

Knappen "Om HEVN-J" åpner et vindu, med informasjon om utviklerne av HEVN-J og av HEVN.



Bilde 3-7 Hjelpesfunksjonaliteten i HEVN-J

3.2.3. KONFIGURASJONSVERKTØY

Det grafiske brukergrensesnittet for konfigurasjonsverktøyet baserer seg på konfigurasjonsverktøyet i HEVN. Det har blitt lagt til en del ny funksjonalitet i verktøyet som er laget. Det har også vært nødvendig å fjerne noe funksjonalitet, dette på grunn av endringer i HEVN-J.

Verktøyet er delt opp i sju ulike kort på samme måte som i HEVN-J. Kortene er "Database", "Innstillinger", "Fridager", "Aktiviteter", "Administratorer", "Sletting" og "Poststeder".

"**Database**" inneholder funksjonalitet for å legge til og fjerne brukere i HEVN-J. For å legge til en bruker lages et fritt valgt brukernavn, med tilhørende fornavn og etternavn for denne brukeren. Kortet har funksjonalitet for å kunne generere hele databasen fra bunnen av, hvis databasen skulle ha blitt korrumpert. Ved generering av ny database fra bunnen av, vil all innlagt informasjon i databasen gå tapt.

"**Innstillinger**" her finnes muligheter for sette telefonprefiks for bruk i HEVN-J på "Personalia" kortet. Det er mulig å sette passord som brukes ved spesiell innlogging til HEVN-J. Avdelinger kan legges til og fjernes på dette kortet.

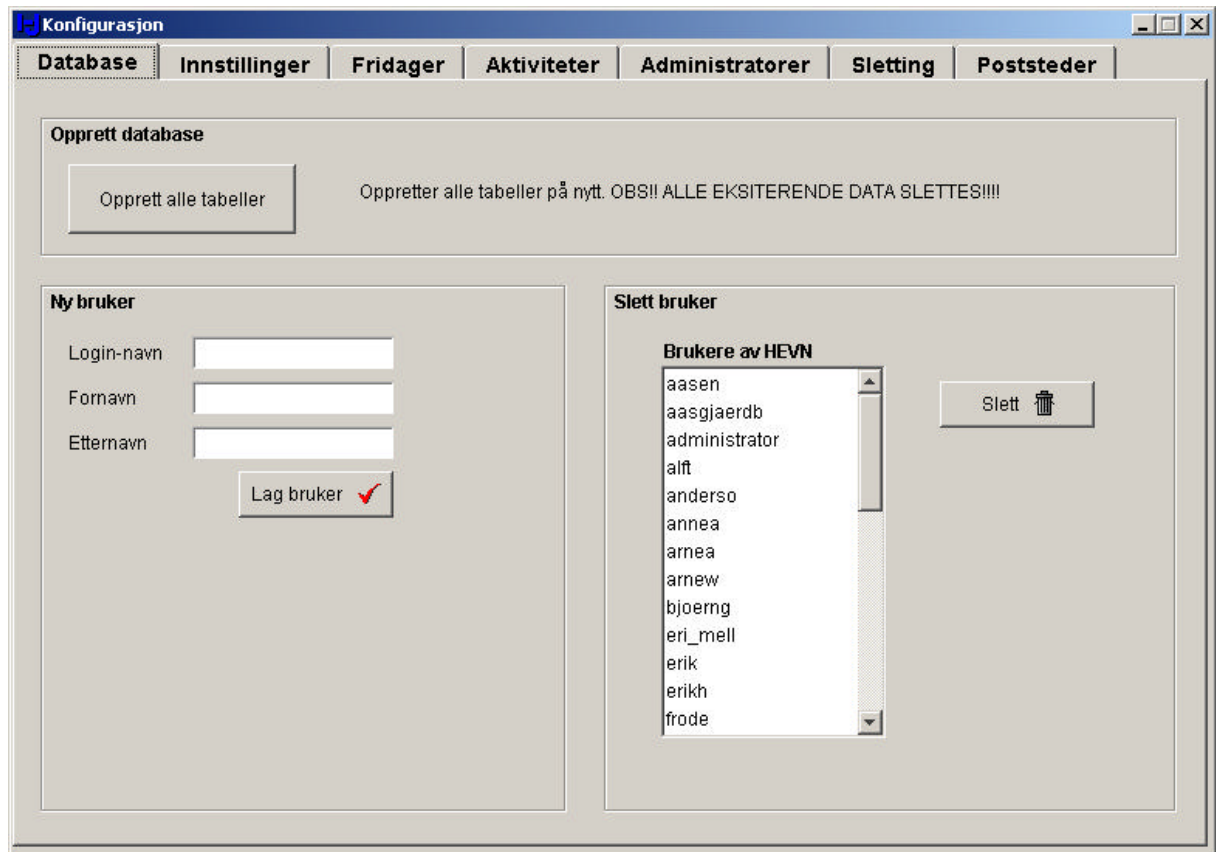
"**Fridager**" gir mulighet for å legge inn egendefinerte fridager. Det finnes funksjonalitet for å generere offisielle fridager. Ved denne genereringen vil det bli lagt inn fridager 6 år frem i tid.

"**Aktiviteter**" inneholder standardaktiviteter som blir vist i HEVN-J ved innlegging av en aktivitet. Her kan HEVN-J administrator legge til og fjerne aktiviteter etter eget ønske.

"**Administrator**" her gis det mulighet for å legge til og fjerne administrator i HEVN-J. Administratorene har tilgang til å endre opplysninger på "Andre..." og "Rom/Utstyr" kortet.

"**Sletting**" inneholder funksjonalitet for å slette og endre eier av Rom/Utstyr. Endre eier av Rom/Utstyr må gjøres når en bruker skal slettes fra systemet og han er eier av et Rom/Utstyr. På dette kortet kan det slettes lagrede aktiviteter for alle brukere. Her kan aktiviteter som er eldre enn 2 år slettes.

"**Poststeder**" har en liste over alle poststeder i hele Norge. Her kan nye poststeder legges til og gamle fjernes. Det er mulig å importere en oppdatert liste over poststeder fra Posten Norge.



Bilde 3-8 Konfigurasjonsverktøy for HEVN-J

3.2.4. DATABASEKONFIGURASJONSVERKTØY

Databasikonfigurasjonsverktøyet ble laget etter krav fra IT-tjenesten ved Høgskolen i Gjøvik. Verktøyet setter opp en Microsoft SQL database fra bunnen av, med tilhørende tabeller og views.

Dette verktøyet har et enkelt grafisk brukergrensesnitt. Grensesnittet er et enkelt skjema for innlegging av databasespesifikke parametere for HEVN-J.

Parameterne i databasikonfigurasjonsverktøyet er som følger:

- IP/Host til SQL-server
- Databasenavn på SQL-server
- MSSQL TCP/IP portnummer
- Brukernavn for SQL-server
- Passord for SQL-server
- POP3-server til å autentisere mot

Ytterligere forklaring av dette verktøyet finnes i vedlegg [K].

Databasekonfigurasjon HEVN-J

Database fil

Her lages dbhevn.cfg filen som brukes av HEVN-J og konfigurasjons verktøyet.

IP/Host SQL server: 128.39.80.22

Databasenavn: hevn

MSSQL Portnr: 1433

Brukernavn: hevnj

Passord: *****

Bruk POP3 for autentisering

POP3 server: pat.hig.no

Lag dbhevn.cfg

Database tabeller

Lag ny database med oppsett ovenfor

Bilde 3-9 Databasekonfigurasjonsverktøy for HEVN-J

3.2.5. WEBAPPLIKASJON

Webapplikasjonen er bygd opp av fire *frames*. Dette gir en enkel og oversikkelig webside. Sidene er en blanding av *HTML*, Java-servlets, *JavaScript*, *CSS* og Java-applet.

Utviklingen av denne applikasjonen hadde ikke topp prioritet fra oppdragsgivers side. Derfor kom utviklingen sent i gang. Dette har allikevel ikke påvirket resultatet i negativ grad. Utviklingsprosessen har tatt kortere tid enn beregnet, da komponenter fra HEVN-J er brukt uten store forandringer. TimeTable-komponenten har blitt brukt som en "applet" i vår webapplikasjon.

Sidene som vises til brukerne blir generert av Java-servlets. Vi så det mest hensiktsmessig å bruke servlets, da HEVN-J har blitt utviklet i Java. Servlet-sidene jobber mot databasen og presenterer dens innhold for brukerne. Alle personalopplysninger i webapplikasjonen vises ved hjelp av skjemaer etter samme mønster som det gjøres i HEVN-J.

Det er mulig å se timeplanen for rom og utstyr. Denne informasjonen blir vist på samme måte som timeplanen for de ansatte.

For timeplandelen ble muligheten for å se fram i tid begrenset til 50 uker. Det er ikke mulig å se tilbake i tid. Dette var et krav fra oppdragsgiver. For at dette skulle bli korrekt, ble klokken på webserveren brukt til å sette tiden. Dette hindrer brukerne i å stille klokken på sin egen maskin, for å kunne se tilbake i tid.

Timeplanen er laget som en Java-applet. Alle aktivitetene på denne timeplanen blir tilsendt som parametere. Dette må gjøres da en applet ikke kan snakke med en databaseserver på en annen maskin.

Det ble lagt stor vekt på at webapplikasjonen skulle kunne vises i alle nettlesere. For å få dette til har mange ulike nettlesere på forskjellige operativsystemer blitt testet. Et resultat av denne testingen var at TimeTable-komponenten måtte konverteres til Java versjon 1.1. Dette gav noe merarbeid, men resultatet ble bra. Et annet moment som det måtte taes hensyn til under utviklingen, var nettlesernes ulike støtte for frames. Dette ble løst ved hjelp av JavaScript.

Høgskolen i Gjøvik - Hvem Er Hvor Når (HEVN-J) - Microsoft Internet Explorer

Address: http://128.39.20.22:8000/servlet/vevd

Sven Erik Espeland Personals

← UKE 20 →

	Mandag 13.5.2002	Tirsdag 14.5.2002	Onsdag 15.5.2002	Torsdag 16.5.2002	Fredag 17.5.2002
09.00					17. mai
10.00					
11.00					
12.00					
13.00					
14.00					
15.00					
16.00					
17.00					
18.00	(Privat)	(Privat)	(Privat)	(Privat)	
19.00	Hovedprosjekt	Hovedprosjekt	Hovedprosjekt	Hovedprosjekt	
20.00			Møte HP med FrodeH		
21.00	Lunsj	Lunsj	Lunsj	Lunsj	
22.00	Hovedprosjekt	Hovedprosjekt	Ute en tur Ute best	Hovedprosjekt	

Left sidebar (vertical list):

- E: Eng, Pål Erik; Espeland, Sven Erik
- F: Fønnes, Hilde; Fong, Ivar
- G: Gundersen, Egon
- H: Hattestad, Rune; Haug, Frode; Hegstad, Paul Ole; Hjeltnes, Erik
- K: Kollum, Dennis; Kravdal, Torbjørn
- L: Lærtengen, Kari; Lina, Rolf
- M: (empty)

Bottom status bar: Applet startSide started | Internet

Bilde 3-10 Webapplikasjon for HEVN-J

3.2.6. DATABASE

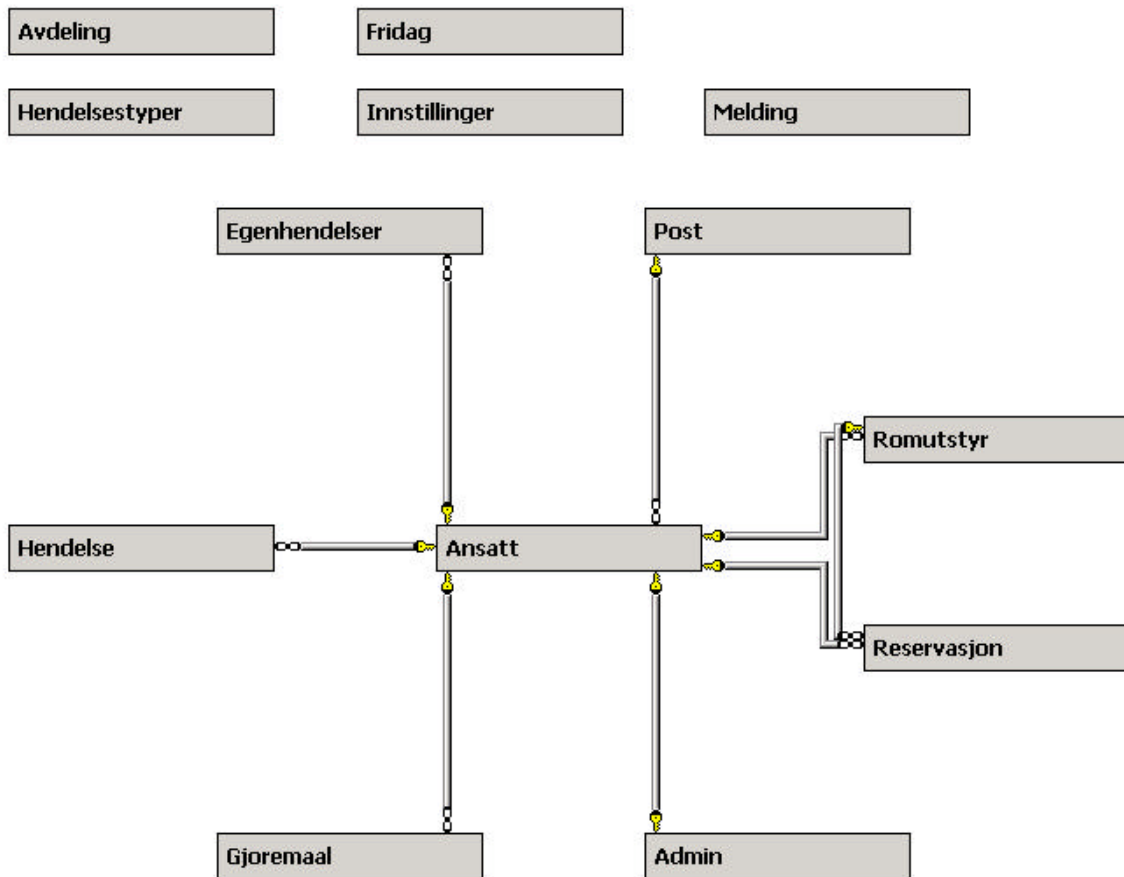
Vi startet arbeidet med å opprette en kopi av den databasen HEVN benyttet. Siden er denne forandret av ulike årsaker. Da noe funksjonalitet i applikasjonen skulle fjernes, førte dette til at noen attributter måtte fjernes i databasen. Ny funksjonalitet underveis i prosjektet skapte behov for å lage nye tabeller og attributter. Tabellene Romutstyr og Reservasjonen kom som et resultat av ny funksjonalitet for å registrere reservasjoner på skolens rom og utstyr. Det samme var tabellen Gjoremaal, som brukes for informasjon om daglige gjøremål i timeplanen. Da det var ønske om å kunne lagre meldingene fra en ansatt i databasen, selv om denne brukeren var slettet, kunne det ikke være noen direkte relasjon mellom melding og ansatt. Det ble derfor laget en meldingstabell uten relasjoner, slik at disse meldingene fortsatt kunne eksistere selv om en ansatt ble fjernet.

Underveis i prosjektet ble det klart at vi trengte å bruke et *characterset* som heter ”Scandinavian dictionary order, case-insensitiv, for use with the 850 (Multilingual) Character Set”, for å kunne bruke Æ, Ø og Å. Dette var et resultat av den *databasedriveren* som ble brukt, for mer om databasedriveren se kapittel 4.1.3.

Valget av MSSQL database og krav til gratis databasedriver, gav oss bare et mulig valg av databasedriver. Det ville vært flere mulige valgalternativer ved bruk av *open source* databaser.

3.2.6.1. BESKRIVELSE AV DATABASEN

Figur 3-1 viser tabeller og relasjonene i HEVN-J. Komplette liste over tabellene og alle attributter finnes som vedlegg [H].



Figur 3-1 Relasjonsmodell for HEVN-J

Ved prosjektslutt hadde databasen følgende tabeller, attributter og nøkler.

Admin

Er en tabell hvor "login" til alle med administrasjonsrettigheter ligger

Primærnøkkel: login
Fremmednøkkel: ingen

Ansatt

Her ligger all informasjon som man kan lagre om en person. Denne tabellen består av en lang liste attributter, som inneholder personalia og det personlige oppsette av HEVN-J.

Primærnøkkel: login
Fremmednøkkel: postnr i tabellen Post

Avdeling

Denne tabellen inneholder kun "avdnavn" for skolens ulike avdelinger.

Primærnøkkel: avdnavn
Fremmednøkkel: ingen

Egenhendelser

I denne tabellen lagres de ulike egne aktiviteter som en bruker kan lage på oppsett kortet. Inneholder kun "login" og "hendelsestype"

Primærnøkkel: login og hendelsestype
Fremmednøkkel: login i tabellen Ansatt.

Fridag

Her lagres informasjonen om årets fridager, slik at disse kan markeres i timeplanen. Alle fridagers "dato" ligger her, sammen med en "kommentar" og en boolsk variabel som sier om dagen er "egendefinert".

Primærnøkkel: dato
Fremmednøkkel: ingen

Gjoremaal

Dette er en ny tabell som inneholder hver dags tidsuavhengige gjøremål. Denne tabellen inneholder "login", "dato" og en "kommentar" som er dagens gjøremål.

Primærnøkkel: login og dato
Fremmednøkkel: login i tabellen Ansatt

Hendelse

Denne tabellen inneholder all informasjon som blir lagret ved registrering av en aktivitet i en av timeplanene. Disse attributtene er "login", "starttid", "sluttid", "hendelsestype", "kommentar", "oppsett" og "erpersonlig". Det nye attributtet "erpersonlig", benyttes for å markere at en aktivitet er privat/personlig i timeplanen. Verdien av oppsett angir om det er en repeterende eller kun en uke denne hendelsen skal gjelde for.

Primærnøkkel: login og starttid
Fremmednøkkel: login i tabellen Ansatt

Hendelsestyper

Tabellen inneholder "hendelsestype", og er en liste over standard hendelser som dukker opp når man registrerer en ny aktivitet på timeplanen. Denne listen kan editeres fra konfigurasjonsverktøyet.

Primærnøkkel: hendelsestype
Fremmednøkkel: ingen

Innstillinger

Denne tabellen holder på fellesinnstillinger for alle brukerne av systemet. Tabellen har "ident" og "value" som attributter. Tabellen har telefonprefiks som "ident" og 611 som "value". I tillegg har den en "ident" til passord og passordet lagres i "value". Denne tabellen brukes kun til dette.

Primærnøkkel: ident
Fremmednøkkel: ingen

Melding

All meldingsinformasjon blir lagret i denne tabellen. "Avsender", "mottaker", "sendt", "status", "tekst", "reservasjon", "vistdialog" og "mldboks" er attributter i denne tabellen. Der "mldboks" angir om meldingen skal ligge i innboksen eller utboksen, "vistdialog" blir satt til 1, hvis det har blitt vist en dialogboks med beskjed om ny melding. For å skille mellom vanlige meldinger mellom brukerne, og reservasjonsmeldinger sendt av applikasjonen, benyttes "reservasjon". "Status" angir om meldingen er lest, og "sendt" er et 15 sifferet tall som angir når meldingen er sendt.

Primærnøkkel: avsender, mottaker, sendt og mldboks
Fremmednøkkel: ingen

Post

Inneholder alle "postnr" og "poststed" i Norge.

Primærnøkkel: postnr
Fremmednøkkel: ingen

Reservasjon

Er en tabell tilvarende hendelse, men benyttes i forbindelse med registrering av rom og utstyr. Tabellen inneholder "login", "starttid", "sluttid", "mode", "romnavn", "kommentar", "annen", "resnavn" og "reslogin". "Resnavn" er navnet på reservatøren. "Annen" angir om rommet skal registreres på noen andre enn de ansatte ved HiG. (Som ikke finnes i navnelista ved registrering). "Mode" brukes til å skille mellom repeterende og enkel aktivitet. "Reslogin" er login til den brukeren som har opprettet reservasjonen.

Primærnøkkel: login, starttid og romnavn

Fremmednøkkel: login i tabellen Ansatt, og romnavn i tabellen Romutstyr.

Romutstyr

Denne tabellen brukes til å lagre informasjon om rom og utstyr som skal registreres. Denne tabellen inneholder "romnavn", "type", "hvem", "endre", "slette", "tidsbegrensning", "fomtid", "tomtid" og "login". "Romnavn" er navnet på rommet eller utstyret. "Type" angir om det er et rom eller utstyr, slik at ressursen kan legges i riktig liste. "Hvem" angir hvilken kategori brukere som kan foreta en reservasjon på rommet. "Endre" angir på samme måte hvem som kan endre reservasjoner, og "slett" angir hvem som kan slette ressursen. "Tidsbegrensning" angir om rommet skal være tidsbegrenset i en periode. "Tomtid" og "fomtid" angir da denne perioden. "Login" er login på den som har registrert ressursen.

Primærnøkkel: romnavn

Fremmednøkkel: ingen

4. KODING

4.1. VALG AV VERKTØY

4.1.1. HOVEDPROGRAMMET

HEVN ble laget i Borland Delphi 3. Problemet med denne versjonen er at den ikke kan kjøre på Unix/Linux og MacOS. Siden HEVN ble laget i 1998, har antall brukere av disse operativsystemene økt, dermed var det ønskelig med en versjon av HEVN som støttet disse. Ved å introdusere en plattformuavhengig løsning vil ingen av skolens ansatte være forhindret i å benytte HEVN.

I arbeidet med å lage en plattformuavhengig versjon av HEVN, var det naturlig å benytte Java som utviklingspråk. Gjennom å benytte Java, var vi sikre på at koden som ble laget, ville kunne kjøres på alle operativsystemer.

Oppdragsgiver hadde ønske om en tilnærmet eksakt kopi av den eksisterende versjonens utseende. Ved valg av utviklingsverktøy, ble det derfor viktig for oss at verktøyet hadde en bra og effektiv måte for å lage brukergrensesnitt. Ved å velge Borland JBuilder 6.0 fikk vi et verktøy som dekket våre behov med tanke på *GUI*. En rask avgjørelse for å bruke Borland JBuilder 6.0 ble tatt. Denne raske avgjørelsen, førte til at videre arbeidet på hovedprosjektet kom raskt i gang. Det var også dette verktøyet oppdragsgiver anbefalte, dermed var alle parter godt fornøyd.

Ved å velge Java som utviklingspråk, er det noen ulemper. Disse ulempene kan allikevel ikke oppveie fordelene med enkelt å få en plattformuavhengig løsning. For å kunne kjøre Java applikasjoner på en maskin, må det være installert en Java VM. Denne tar noe plass, men dette bør ikke være noe problem for skolens maskiner. Dessuten har de fleste ansatte allerede installert en VM, for å kjøre andre Java applikasjoner. *JDK 1.4.0* ble benyttet under utviklingen av vår applikasjon.

4.1.2. WEB-GRENSESNIETTET

Vi valgte å bruke Java-servlet som et bindeledd mellom databaseserveren og en applet. Dette fordi en applet ikke kan jobbe direkte mot en database som står på en ekstern maskin. I utviklingen av webgrensesnittet er det brukt en vanlig teksteditor. Appleten ble utviklet i Borland JBuilder 6.0.

4.1.3. DATABASE

For å kunne koble Java mot en database trengs det en *JDBC* driver. Da applikasjonen skulle jobbe mye mot en database, var det viktig at JDBC driveren som ble valgt fungerte bra. Problemet med disse driverne er at de ofte koster en del kroner i innkjøp. Det var ikke ønskelig fra oppdragsgivers side å bruke penger på dette. Det ble brukt litt tid på å finne en driver som var gratis. Til slutt falt valget på TwFreeTds JDBC 2.0 driver for SQL Server fra ThinWeb. Denne var ikke optimal, men gav oss tilnærmet den støtten applikasjon behøvde. For denne driveren er det ingen garanti eller support.

Vi har løst problemene som denne gratisdriveren har gitt oss, ved å utføre flere SQL spørringer mot databasen og at databasen har måttet settes opp på en spesiell måte. Dette er et arbeid som har vært tungvint og som hadde vært unngått ved å kjøpe en bedre driver.

4.1.4. INSTALLASJONSPROGRAM

Vi hadde valget mellom å lage installasjonsprogrammet selv, eller å finne et gratis verktøy. Da det ville ta lang tid å utvikle et installasjonsprogram selv, samt at dette ikke var en del av oppgaven, ble det valgt å bruke et ferdigutviklet verktøy.

”InstallAnywhere 4.5 Now” ble valgt. Dette er et gratis verktøy, som er spesielt utviklet for installasjon av Java-applikasjoner. Dette verktøyet ble lastet ned fra Internett [9]

Det er laget en egen installasjonsveiledning for alle systemets deler. Denne er å finne som vedlegg [K].

4.2. BRUK AV XP

Ved å benytte eXtreme Programming (XP), har vi hatt ett rammeverk for hvordan arbeidet med koden bør foregå. Denne modellen har gitt oss en måte å arbeide på som vi tror har vært gunstig for resultatet av arbeidet. Gjennom å benytte en slik modell har vi kunnet benytte en allerede ferdigutviklet arbeidsmetode. Dette har spart oss for tid, samt at arbeidet har blitt mer strukturert. Begrunnelse for hvorfor XP ble valgt som utviklingsmetode finnes i kapittel 7.1.1. Det er skrevet et lite notat som sammenfatter historie og arbeidsmetoder for XP. Dette notatet finnes som vedlegg [C].

4.2.1. HVA HAR VI LAGT VEKT PÅ VED XP

Under arbeidet med vårt prosjekt er det lagt spesielt vekt på enkelte deler av arbeidsmetodene XP anbefaler, for mer om XP se [4]. Vi føler at XP har gitt oss flere gode arbeidsrutiner under kodingsarbeidet, og har således vært med på å heve kvaliteten på vårt arbeid.

I XP er det anbefalt å skrive testene for hvordan applikasjonen skal virke før man starter arbeidet med koding. I vårt prosjekt har HEVN blitt brukt for å tilfredstille dette. Ved en slik arbeidsmetode har vi hele tiden hatt klart for oss hva vår applikasjon skal gjøre og hvilke resultater ulike deler av applikasjonen skal gi. Disse testene på forhånd har også ført til at det har vært mulig å produsert det kunden var ute etter, en ny versjon av HEVN utviklet i Java. Vi har heller ikke vært i tvil om hvordan funksjonaliteten i applikasjonen skulle være. Når det gjelder ny funksjonalitet slik som utviklingen av ”Rom/Utstyr” kortet i applikasjonen, var vi nok ikke like flinke til å lage fullstendige tester før arbeidet med kodingen startet. Det ble allikevel forsøkt å lage enkle systemsekvensdiagrammer som beskrev hva som skulle skje ved ulike systemhendelser.

En arbeidsmetode fra XP som er benyttet, er bruk av ”On site customer”. En ”On site customer” er en person fra kunden som er med på utviklingsarbeidet, og tar avgjørelser

underveis i arbeidet. På vårt prosjekt har dette vært Frode Haug. Det hadde vært ønskelig at han hadde vært en del av utviklingsteamet vårt, men dette har ikke vært mulig, fordi han har hatt mange andre oppgaver ved skolen under prosjektperioden. Han har imidlertid vært lett tilgjengelig på sitt kontor underveis i arbeidet. Dette har ført til at han har kunnet ta avgjørelser når vi har stått ovenfor ulike valg av design og funksjonalitet. Da han har hatt mulighet til å ta raske avgjørelser, har det sjelden blitt laget funksjonalitet som kunden ikke har ønsket seg. Vi har også kunnet løse designmessige problemer riktig første gang, dermed har det vært få ting som har måttet blitt gjort om igjen.

All den ulike funksjonaliteten som er utviklet, har blitt delt opp i brukerhistorier, eller ”*User stories*”. Disse historiene gjenspeiler de ønsker og krav kunden har hatt underveis i prosjektet. Arbeidet med disse har blitt prioritert av kunden, slik at det kunden har prioritert høyest har blitt ferdig først. I våre møter med kunden har historiene blitt prioritert, samtidig som det har estimert hvor lang tid det vil ta å få disse realisert. De ferdigstilte historiene er siden integrert i applikasjonen, og kunden har da gitt tilbakemelding på disse. Feilretting og utbedring av ferdigstilt funksjonalitet har blitt satt opp som egne brukerhistorier, hvor prioritering og estimering av tidsforbruk har blitt satt opp på nytt.

Dette har ført til en arbeidsgang hvor det første som ble gjort var å sette opp GUI, videreutvikle den funksjonaliteten som eksisterte fra før, for til slutt å realisere ny funksjonalitet. Prioritering av brukerhistoriene fra kunden har ført til at arbeidet med webapplikasjon startet seint, dette fordi kunden hadde et større ønske om at funksjonalitet og testing av HEVN-J skulle være ferdig til den 23. mai. Arbeidsmetoden hvor kunden har prioritert arbeidsgangen, har vært grei å forholde seg til. Vi føler også at dette har vært med på å redusere muligheten for falske forhåpninger hos kunden, ved at han hele tiden har kunnet vite hvor mye som er utviklet, og hva som er neste arbeidsoppgave for oss som utviklere.

I kodingsfasen av prosjektet, har vi forsøkt å benytte oss av par-programmering. Dette er en metode som går ut på at to og to arbeider sammen på en maskin. Tanken bak dette er at to hoder tenker bedre enn ett. Vi har også erfart at dette er en fin måte å lære av hverandre, samtidig som alle gruppens deltakere har kunnet holde oversikt over statusen på prosjektet. Å benytte par-programmering, har også ført til en følelse av felles eierskap av koden, ved at det ikke er noen som alene, har utviklet deler av applikasjonen uten at noen har vært med i det hele tatt. I siste del av prosjektet, har vi allikevel ikke fullt ut kunnet gjennomføre par-programmering, da arbeidspresset har ført til at arbeidet måtte deles opp i flere ulike oppgaver og fordeles på gruppedeltakerne.

For at gruppens deltakere skal ha lett for å sette seg inn i hva andre deltakere har gjort, er det en fordel å benytte en felles kodestandard. En slik standard gjør koden mer oversiktlig og det blir et mer helhetlig preg over den. En kodestandard er valgt for vårt prosjekt, og den er beskrevet i kapittel 4.3

For å holde høy kvalitet på koden blir det i XP anbefalt å unngå overtidsarbeid. Dette var også noe som ble forsøkt i begynnelsen av prosjektet, men som viste seg å være vanskelig mot slutten av prosjektet. Bakgrunnen for dette, var at kunden hadde mange nye krav som han ville ha implementert, og som vi følte det var vanskelig å si nei til å implementere. Dette ville nok blitt litt annerledes i en virkelig arbeidssituasjon. Da kunne en del av applikasjonen blitt sluppet ved den endelige fristen, for siden å ha implementert den nye funksjonaliteten og sluppet dette som en ny release ved en senere dato.

4.3. PRINSIPPER VED KODING

I startfasen av prosjektet ble det forsøkt å skape gode rutiner for hvordan koden skulle skrives og organiseres. Dette har vi forsøkt å fortsette med, selv om det enkelte ganger har sklidd litt ut. Slik ukontrollert koding er noe som typisk skjer, i iveren etter å få løst et programmeringsproblem ved noen raske endringer i koden. Får man så løst dette problemet, er det lett å fortsette videre uten å strukturere det man har fått til.

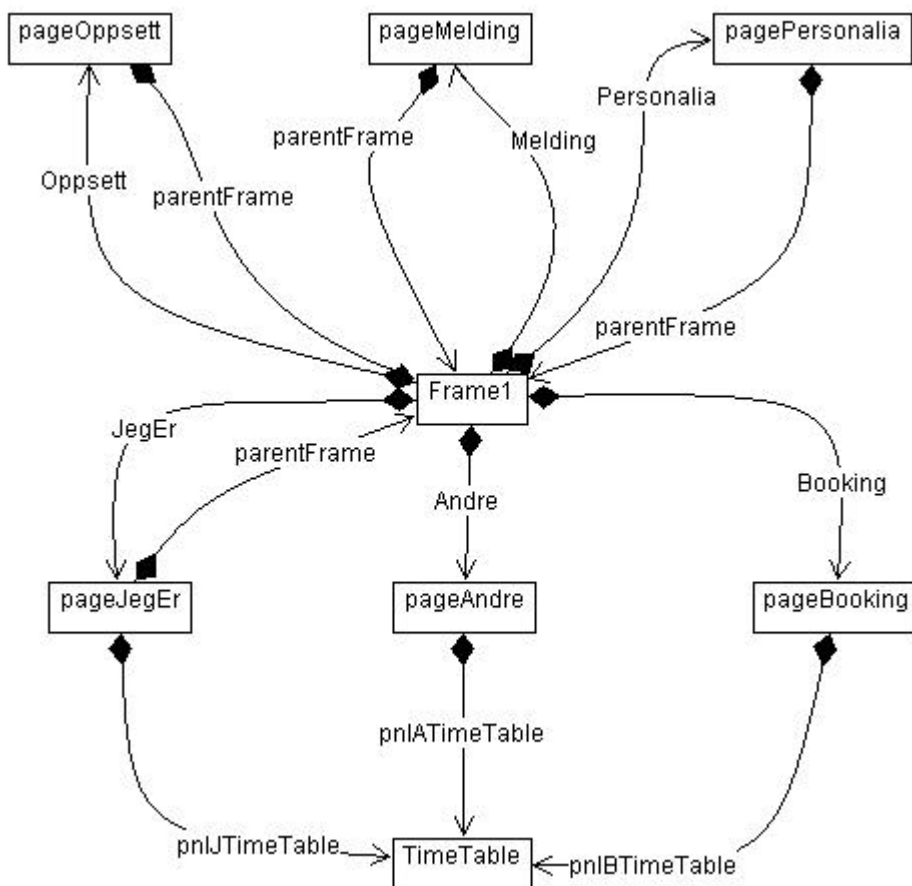
Derfor har det vært forsøkt å bruke noe tid i løpet av uken, på å rette opp slike "grisete" partier i koden. Disse oppryddingene har vært med på å holde koden oversiktlig, noe som har ført til få problemer med vedlikehold og feilretting underveis. Slikt arbeid er en nødvendighet for en så stor applikasjon som vi har arbeidet med. Det ferdige systemet består av 35.000 kodelinjer, og krever at det finnes struktur og system.

4.3.1. KLASSEINDELING

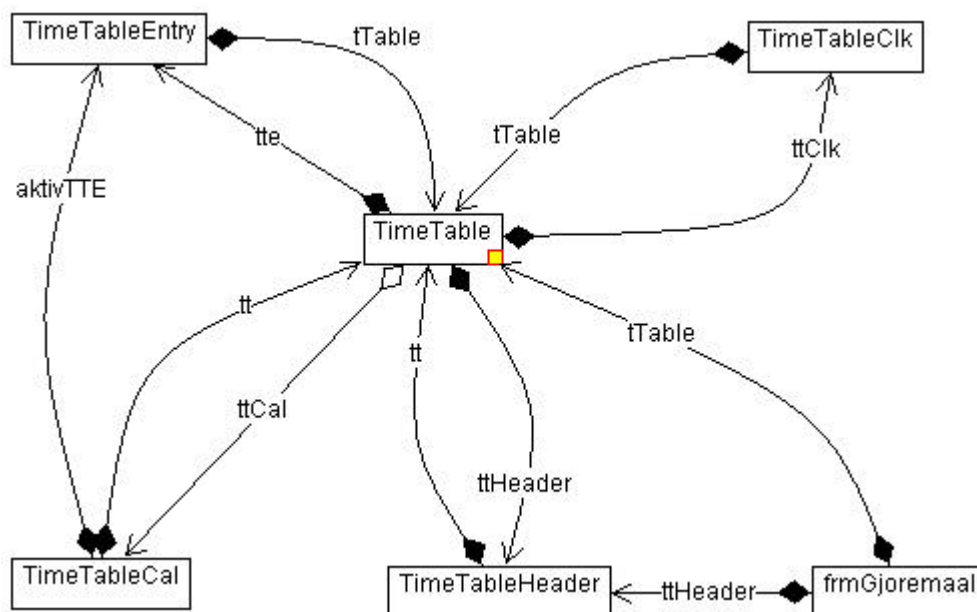
Klasseinndeling av applikasjonen har kommet som en naturlig følge av at programmeringen har blitt gjort i Java, hvor stort sett alt er delt opp i klasser. Derfor ble det naturlig at også vår kode ble delt opp etter samme mønster.

Klassene er i hovedsak delt opp etter applikasjonens åtte komponenter, de seks kortene, timeplankomponenten og databasemotoren. Det har vært behov for mange små klasser, som for eksempel de ulike dialogboksene. Disse er også laget som egne klasser, da de skal vises i egne vinduer.

Det har blitt benyttet et verktøy for "reversed engineering" for å lage enkle klassediagrammer for hovedklassene i vår applikasjon. Disse viser en grei sammenheng mellom systemets ulike deler. Figur 4-1 viser hovedklassene i hele applikasjonen og figur 4-2 viser hovedklassene i TimeTable-komponenten.



Figur 4-1 Klassediagram for hele systemet laget ved hjelp av "Reversed Engineering"



Figur 4-2 Klassediagram for timeplankomponenten laget ved hjelp av "Reversed Engineering"

Det har vært en fordel under arbeidet at vi har hatt en bevist klasseinndeling. Dette har ført til at vi har kunne jobbe med ulike deler av koden uavhengig av hva andre gruppedeltakere har arbeidet med.

Mer om klasseinndeling finnes under kapittel 4.4.1 hvor timeplankomponenten er beskrevet.

4.3.2. KOMMENTARER

I alle programmeringsfagene på skolen har viktighet av å kommentere kode blitt poengtert. I et så stort prosjekt som nå er gjennomført, har vi erfart at å kommentere kode er en nødvendighet. Å gi forklarende og entydige funksjonsnavn, har også vært et viktig virkemiddel for å gjøre koden lett å arbeide med. Da flere personer arbeider på den samme koden, ville det uten bra kommentering og fornuftige funksjonsnavn, blitt umulig å holde oversikten over systemets 35.000 kodelinjer.

Vi har valgt å ikke kommentere linje for linje i koden. Grunnen til dette er at det ville tatt for mye tid å skrive kommentarer til alle kodelinjer som er laget. Det har i stedet blitt lagt vekt på å kommentere i starten på funksjonene og forklart hva disse skal utføre. Denne kommenteringen er i henhold til standarden for automatisk dokumentasjon i Java. Denne Java dokumentasjonen er lagt ved på cd-en som følger med rapporten. Denne kommenteringen i tillegg til fornuftige valg av variabel og funksjonsnavn, har vært med på å gjøre koden vår lettere å forstå.

Under følger et eksempel på hvordan koden er kommentert for at den automatiske genereringen av dokumentasjonen skal kunne skje ved hjelp av Java.

```

/**
 * Tilpasser lagrede aktiviteter.
 * Splitter opp, sletter eller endre start og sluttiden for
 * lagrede aktiviteter.
 * @param ttEntry Et TimeTableEntry objekt
 * @param vt En vektor med alle konflikttidene denne dagen
 */

public void TilpassLagredeAktiviteter(TimeTableEntry ttEntry, Vector vt){

    long konfliktTider[];
    long tteStartTid = ttEntry.ret_bHour(); // starttiden
    long tteSluttTid = ttEntry.ret_EHour(); //sluttiden

    for(Enumeration e = vt.elements(); e.hasMoreElements();){
        konfliktTider = (long[]) e.nextElement();

        if(tteStartTid<=konfliktTider[0] && tteSluttTid>=konfliktTider[1]){
            //skriver over hele
            dbMot.slettTTE(login," " +konfliktTider[0]);
        }
        else if(tteStartTid <=konfliktTider[0] && tteSluttTid > konfliktTider[0]){
            //starter over og slutter midt i
            dbMot.oppdaterTTEStartTid(login, konfliktTider[0], tteSluttTid);
        }
        else if(tteStartTid<konfliktTider[1] && tteSluttTid>= konfliktTider[1]){
            //starter midt i og slutter over
            dbMot.oppdaterTTESluttTid(login,konfliktTider[0], tteStartTid);
        }
        else if(tteStartTid>konfliktTider[0] && tteSluttTid <konfliktTider[1]){
            //starter midt i og slutter midt i
            dbMot.splittTTE(login,konfliktTider[0],tteStartTid,tteSluttTid);
        }
    }
    returnerTTEtilDB(ttEntry); //lagrer TimeTableEntry på databasen

}

```

4.3.3. KODEFORDELING

Det er kun brukt private variable i alle klasser. Dette har ført til at det har blitt mindre forvirring om hvor variablene hører hjemme. Når det allikevel har vært behov for disse variablene i andre deler av koden, er det brukt egne funksjoner i hver av klassene som returnerer disse.

4.4. EGNE KOMPONENTER

4.4.1. TIMETABLE-KOMPONENTEN

TimeTable-komponenten er den sentrale komponenten i vår applikasjon. Denne komponenten ble utviklet, da det var vanskelig å finne en ferdigutviklet komponent, som inneholdt all den funksjonaliteten som var ønsket. Vi fant det ikke hensiktsmessig å bruke koden til timeplankomponenten i HEVN. Grunnen til dette er at det tar lang tid å sette seg inn i andres kode. Da koden var laget i Delphi, var dette heller ikke kode som kunne brukes direkte i vår applikasjon. I tillegg trengtes ny funksjonalitet i forhold til HEVN. Eksempler på dette er 24 timers dager, private aktiviteter og funksjonalitet for "Rom/Utstyr" kortet.

TimeTable-komponenten består av tre visuelle deler; klokken på venstre side, ukedagene med datoer øverst og et rutenett for de fem dagene i sentrum. Da hver dag skal dekke 24 timer, ble det laget et vindu, hvor det var plass til aktiviteter fra 08.00 til 17.00. Vi benyttet derfor en scrollbar for å kunne vise innholdet i timeplanen før klokken 08.00 og etter 17.00. I komponenten står ukedagene og datoene fast øverst, mens man kan scrolle klokken og dagens innhold.

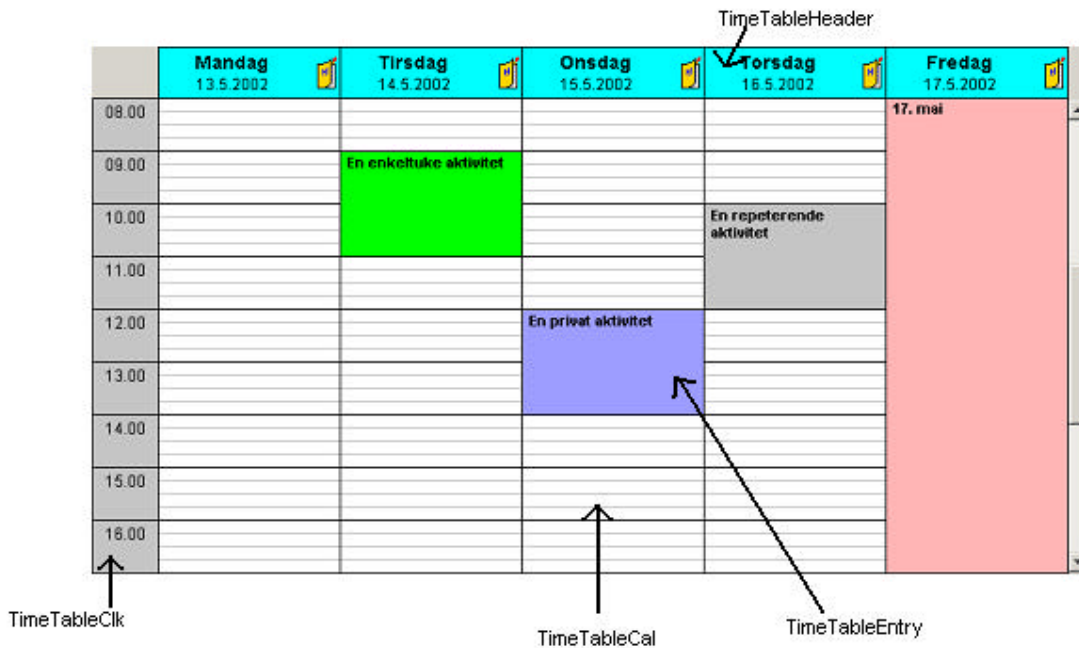


	Mandag 22.4.2002	Tirsdag 23.4.2002	Onsdag 24.4.2002	Torsdag 25.4.2002	Fredag 26.4.2002
08.00					
09.00					
10.00					
11.00					
12.00					
13.00					
14.00					
15.00					
16.00					

Bilde 4-1 Timeplanen i HEVN-J

Det fantes flere måter å få vår komponent lik den som benyttes i HEVN, alle med fordeler og ulemper. Et bilde av HEVN sin timeplan finnes i kapittel 2.3.4.1. Uken ble delt opp i fem dag-paneler, som så aktivitetene ble tegnet på. Å få denne tegningen til å bli riktig, krevde mye arbeid. Dette fordi den skulle kunne håndtere alle de ulike måtene en ny aktivitet skulle kunne opprettes på. En fordel med vår løsningsmetode er at RAM-bruken i applikasjonen er redusert. Java bruker mye RAM, men dette ble noe redusert ved å bruke få underkomponenter. Den største fordel med denne løsningen er at hver av de fem dag-objektene arbeider selvstendig.

Disse fem objektene er av klassen TimeTableCal. Klokkeslettene er et objekt av klassen TimeTableClk, og ukedagene med datoer er et objekt av klassen TimeTableHeader. Disse syv objektene settes så sammen i TimeTable klassen.



Bilde 4-2 Timeplanen med hovedklasser i HEVN-J

TimeTable klassen tar seg av kommunikasjonen mot databasen, gjennom en selvlaget databasemotor. Alle aktivitetene for en dag håndteres av TimeTable klassen, som oppretter de nødvendige objektene av klassen TimeTableEntry. Disse blir sendt til riktig dag-objekt, for opptegning.

Det finnes fire forskjellige typer aktiviteter, disse blir skilt fra hverandre med forskjellig farge. Aktiviteter som kun gjelder en uke har grønn farge. Private aktiviteter er blå. Aktiviteter som går over flere uker er grå og røde aktiviteter er helligdager. Disse aktivitetene blir behandlet likt, med unntak av helligdagene, som ikke kan endres. For å skille ut aktiviteten som foregår "nå", fra andre aktiviteter, blir denne markert med gul farge. Hvis en aktivitet inneholder mer tekst enn hva som kan vises, vil aktiviteten få en hvit trekant nede i høyre hjørne. Man kan da holde musen i ro over denne aktiviteten, og en *hintboks* vil komme fram med hele teksten.

Det blir brukt en TimeTableEntry klasse som har informasjon om hver aktivitet. De viktigste variablene i denne klassen er starttid, sluttid, type aktivitet og hvilken tekst som skal vises. Starttiden og sluttiden brukes i to formater. Et format som angir tidspunktet på formen hh:mm (12:45), mens det andre er et heltall som angir antall millisekund etter 1.1.1970. Det blir brukt to formater fordi dette forenkler ulike regneoperasjoner, samt at all utskrift til skjermen må være på formen hh:mm.

For å legge inn nye aktiviteter markerer man et nytt område i timeplanen. Da kommer det opp en dialogboks, hvor man må spesifisere aktivitetstypen, justere tidsperioden, om det er en privat aktivitet, om den skal gå over flere uker/år og om den skal gjelde annenhver uke. Dersom den innlagte aktiviteten kommer i konflikt med andre allerede innlagte aktiviteter, blir brukeren varslet og det gis mulighet for konflikthåndtering.

Alle aktiviteter kan markeres med musen, for endring og eventuelt sletting i ettertid.

5. TESTING OG KVALITETSSIKRING

Kvalitet på koden som blir produsert er en viktig brikke ved et XP prosjekt. Like viktig som kvalitet på koden, er kvaliteten på det programmeringsarbeidet som blir utført. Hvis programmeringsarbeidet er dårlig, har koden som blir produsert lett for å inneholde feil. For å holde en høy kvalitet på produktet vårt er mye tid brukt på testing av koden, både før, under og etter at den har blitt skrevet. Slike tester, er med på holde antall kodefeil nede, samtidig som de holder oss som programmerere på tå hev.

5.1. TESTING

Testing av applikasjonen har vært et viktig arbeid under utviklingen. Applikasjonen har blitt testet på fire ulike måter. Den første går på testing av utseende i forhold til HEVN. Den andre var testing mot eksisterende funksjonalitet i HEVN. Testing av ny funksjonalitet spesifikk for vår applikasjon. Helhetstesting av hele applikasjonen er den aktiviteten som har vært viktigst og tatt mest tid.

Applikasjon skal kunne kjøres på alle operativsystemer, det har derfor vært viktig for oss å se at dette virkelig fungerer. Det har derfor blitt benyttet en egen maskin som har kjørt Linux. Tester på denne maskinen har gjort oss oppmerksomme på at det finnes enkelte operativsystemspesifikke forskjeller. Eksempel på dette er at Linux trenger større plass til enkelte fonter enn det Windows trenger. Dette er da noe som vi har vært nødt til å ta hensyn til, og aktiv testing har gjort oss oppmerksomme på dette. Det har ikke blitt brukt tid på å teste applikasjonen på Mac OS, da vi ikke har hatt tilgang til slike maskiner. Vi er ikke redde for at applikasjonen ikke skal kunne kjøres på dette operativsystemet, da Java i Mac-OS X vil kunne kjøre med JDK 1.4.0.

Da ulike deler av applikasjonen er laget samtidig, har det vært naturlig å først teste disse delene hver for seg. Senere er det testet at disse delene fungerer sammen og at de kommuniserer riktig med hverandre. Det har vært naturlig at den/de som har utviklet de enkelte delene, har testet sin kode for feil. Senere er disse ulike delene blitt satt sammen til en større enhet, for eksempel et kort, som hele gruppen har brukt tid på å teste. Det er under disse testene de fleste logiske feilene har blitt funnet. Da gruppedeltakere som ikke selv har utviklet en del av applikasjonen, tester på andre kriterier enn det utvikleren av koden selv har gjort, blir mange feil fjernet. Feilsituasjoner som oppstod under disse testene, har blitt vurdert av gruppen og tatt opp med oppdragsgiver.

Det mest ideelle under et XP-prosjekt, er å ha muligheten til å kjøre applikasjonen hos alle brukere fra første release. Da dette av praktiske grunner ofte er umulig å gjennomføre er det isteden vanlig å ha en egen testgruppe. Det er også under vårt arbeid benyttet en slik testgruppe. I utgangspunktet var det noen få testbrukere fra dataavdelingen ved skolen. Dette var en liten gruppe på fire spesielt interesserte brukere. Ved å benytte en slik liten gruppe testere, har vi hatt muligheten til å få tilbakemelding på hvordan vår applikasjon fungerer under vanlig drift. Disse tilbakemeldingene, har blitt benyttet aktivt til å forbedre applikasjonen vår. De har også kommet med gode forslag på hvordan tekniske utfordringer kunne løses. Denne testgruppen er siden utvidet til 20 personer med ulike datakunnskaper. Dette har vært spesielt viktig for å få testet den nye funksjonaliteten. Vi ønsket at den nye

funksjonaliteten skulle integreres med den funksjonaliteten som allerede eksisterte, slik at det ikke ble noe synlig skille mellom gammel og ny funksjonalitet.

5.2. METODER FOR Å SIKRE KVALITET – PROSESSEN

Å miste produsert arbeid, er ikke med på å heve kvaliteten på produktet som skal lages. Derfor bestemte vi oss tidlig for å skape gode backup-rutiner for arbeidet som ble produsert. Vi har tatt med vår egen server på skolen. Denne blir det to ganger om dagen tatt backup av, som blir sendt til Drammen. I tillegg tar JBuilder egen backup av koden hver gang den blir kompilert. Disse rutinene bør være tilstrekkelige for at ikke arbeid skal gå tapt.

Det har ukentlig vært møter med oppdragsgiver/veileder hver onsdag morgen. Her har det kommet tilbakemelding på de ukentlige releasene som er levert, samt at det har blitt diskutert hvilket kort eller funksjonalitet som skulle prioriteres den neste uken. På disse møtene har det blitt avtalt hva som skal være klart til neste ukes release.

For å sikre fremdriften, har det vært daglige møter der vi har snakket om hva det skal jobbes med den dagen. Disse møtene har vært etter XP's mal for "Stand up meetings" [7]. Dette har vært korte møter, der dagens gjøremål har blitt diskutert. Lengre møter, hvor det har vært behov for å diskutere applikasjonsspesifikke problemer, har blitt satt opp som daglige gjøremål hvor alle måtte ta seg tid til å delta. Da arbeidet hele tiden har foregått på et grupperom, har det vært kontinuerlig kommunikasjon mellom gruppens deltakere. Dette har ført til at liten tid har blitt brukt til å ta avgjørelser, og at få spørsmål har blitt glemte.

Vi har under store deler av prosjektet forsøkt å benytte oss av par-programmering. Ved denne arbeidsmetoden, har det hele tiden vært to personer om å ta avgjørelser. På denne måten har løsningsmetodene blitt sett på fra flere vinkler, og logiske feil har blitt luket bort. På slutten av prosjektet har vi måttet gå litt bort fra par-programmering, da ønske om mye ny funksjonalitet førte til et sterkt arbeidspress og behov for arbeid på mer enn to maskiner. Ideelt sett burde ikke dette ha blitt gjort, men mengden av funksjonalitet burde istedenfor blitt redusert, slik at det fortsatt hadde vært større fokus på kvalitet. Oppdragsgiver var veldig interessert i å få mesteparten av ønsket funksjonalitet med i siste release før innleveringen av rapporten. Den økte arbeidsmengden, som dette førte til, kan ha vært med på å øke antall feil i koden. Dette er vanskelig å fastslå før alle deler av applikasjonen er testet over tid i sitt brukermiljø.

5.3. METODER FOR Å SIKRE KVALITET – PRODUKTET

Kvaliteten på en applikasjon, kan måles på ulike måter. Antall feil i koden, antall timer som er brukt på feilretting, gjennomsnitt av svartider og andre faktorer kan måles for å se om disse er tilfredsstillende for kunden. Å være bevist på arbeidet med å gi kunden det han vil ha, kan derimot ofte være vanskelig. Det er derfor viktig at man hele tiden er i dialog med kunden, og at man som gruppe er åpne for innspill. Det har blitt benyttet ulike metoder for å sikre at kunden får det han egentlig vil ha.

De første månedene av prosjektet ble mye tid brukt med oppdragsgiver. Dette ble gjort av ulike grunner. Mye av denne tiden i starten ble brukt til å få det grafiske brukergrensesnittet slik kunden ønsket. Senere i prosjektet gikk mer av arbeidet over til å få funksjonaliteten lik det den var i HEVN. Under dette arbeidet, viste det seg at enkelte komponenter som tidligere var brukt fra Delphi, ikke fantes i Java. Det var derfor viktig at kunden tok avgjørelser på hvilke komponenter og funksjonalitet som skulle erstatte disse. Muligheten til slike raske avgjørelser fra kunden, gav arbeidet god fremgang. Dette sikret også at kundens ønsker ble ivare tatt.

Da arbeidet gikk over til å lage helt ny funksjonalitet, ble det straks viktigere å ha kontakt med en større brukergruppe. Dette for å sikre at funksjonaliteten ikke kun ble utviklet for vår ”On site customer”, men for hele den fremtidige brukergruppen ved skolen. ”On site customer” hadde møter med ulike brukergrupper ved skolen. Disse møtene resulterte i nye brukerhistorier som siden ble realisert helt eller delvis i løpet av en uke. Tilbakemelding fra de ulike brukerne på våre releaser, ble notert på egne feilskjema. Disse skjemaene finnes som vedlegg [G]. På daglige avgjørelser angående utseende og funksjonalitet, var det allikevel kun ”On site customer” som tok avgjørelser. Innspill vedrørende design og funksjonalitet fra andre i brukergruppen gikk i hovedsak gjennom han.

6. INSTALLASJON

6.1. BRUKERAKSEPTANSE

Under hele utviklingsfasen har vår kundekontakt benyttet seg av en testgruppe ved skolen. Denne gruppen, har vært med på å forme applikasjonen slik den fremstår ved den endelige releasen 23. mai. Det er denne testgruppen, sammen med kunden, som vil være med på å ta avgjørelsen på om vår applikasjon blir akseptert og satt i drift.

Da IT-tjenesten ved skolen har mye annet å ta seg av, vil idriftsetting av applikasjonen ikke skje før 10 juni. Selv om oppgaven da er levert, er det naturlig at deler av prosjektgruppen er tilgjengelige en stund etter at applikasjonen er blitt satt i drift, slik at eventuelle feil oppstått etter at applikasjonen har blitt satt i drift kan rettes.

6.2. KONFIGURASJON

Etter mal fra HEVN-gruppen, er det laget et konfigurasjonsprogram for HEVN-J. Dette verktøyet er ment som et konfigurasjonsverktøy for HEVN-J, og det er her muligheter for å endre systemspesifikke parametere. Dette verktøyet gir muligheter for å legge inn nye brukere, telefonprefiks, egne helligdager/fridager, avdelinger ved skolen og felles aktivitetstyper.

Konfigurasjon av webapplikasjonen gjøres ved installasjon av HEVN-J systemet. Dette er dokumentert i vedlegg [K].

6.3. INSTALLERING

For å gjøre installeringen av HEVN-J applikasjonen så enkel som mulig for alle brukerne, ble det valgt å bruke "InstallAnywhere 4.5 Now" fra Zero G. Ved å benytte dette verktøyet, har vi fått en standardisert *installasjonsrutine*, som de fleste brukerne burde kjenne seg igjen i.

Dette installasjonsprogrammet er benyttet under hele utviklingsfasen, og det har fungert meget tilfredstillende. Vi er derfor sikre på at dette vil fungere enkelt og uten problemer for alle fremtidige brukere av HEVN-J.

Installering av webapplikasjonen krever teknisk innsikt i Java og Java-servlets. Derfor har det blitt avtalt med oppdragsgiver at prosjektgruppen skal ta seg av denne installasjonen. Dette vil bli gjort etter levering av hovedprosjektrapporten.

For å sikre at installering, konfigurering og oppstart av systemet skal gå greit, har prosjektgruppen sagt seg villige til å bidra med støtte, også etter den 14. juni.

7. KONKLUSJON

7.1. DRØFTING AV VALG OG RESULTATER

7.1.1. VALG AV SYSTEMUTVIKLINGSMETODE

Når man utvikler en applikasjon for en kunde er det ett overordnet mål; kunden skal bli fornøyd. Det er mange faktorer som er avgjørende for om kunden blir tilfreds med produktet. Noen av de vesentligste momentene er pris, tidsforbruk, funksjonalitet, brukervennlighet og feilprosent/oppetid.

Det er fremmet mange løsningsmetoder for hvordan man best mulig kan ivareta alle disse faktorene. Disse løsningsmetodene har i systemutviklingssammenheng blitt delt inn i forskjellige løsningsmodeller. Disse systemutviklingsmodellene forholder seg ikke bare til hvordan man skal kode applikasjonen man lager, men også hvordan arbeidet med prosjektet bør foregå.

Hvordan man legger opp arbeidsgangen og valg av arbeidsmetodikk er forskjelling fra prosjekt til prosjekt. Videre i dette kapitlet er det forsøkt å forklare og greie ut om hvordan og hvorfor vi har valgt å arbeide med hovedprosjektet på den måten vi har gjort.

Da tre av gruppens deltakere har valgt systemutvikling som retning siste året ved skolen, var dette noe vi hadde gode bakgrunnskunnskaper om. Erfaringene fra disse fagene sa oss at valg av systemutviklingsmodell ofte var fast i en systemutviklingsbedrift. Da vi selv måtte velge hvilken modell vi ønsket å benytte, var det viktig for oss å undersøke hvilke behov vi hadde. De modellene vi hadde best kjennskap til fra før var RUP (Rational Unified Process) og XP (eXtreme Programming). Den førstnevnte eller avarter av denne blir mest brukt i virkelige systemutviklingsprosjekter i dag.

RUP er en tung og relativt godt innarbeidet metode innen systemutvikling, mens XP er en relativt ny utviklingsmodell som har kommet de senere år. Disse to metodene blir av mange karakterisert som ytterpunktene innen dagens systemutviklingsmodeller. RUP legger stor vekt på dokumentasjon og langsiktig planlegging av arbeidet. Ved å benytte denne metoden bruker man relativt lang tid på å definere krav, og på å sette opp modeller og skjemaer for hvordan applikasjonen skal fungere innad og mot brukerne.

Dette systemeringsarbeidet mente vi og oppdragsgiver allerede var utført under arbeidet med HEVN, og at det ikke var nødvendig å gjøre dette på nytt. Da det var noen nye krav og noe ny funksjonalitet som skulle implementeres i vår applikasjon, ble XP benyttet som arbeidsmåte.

En av suksessfaktorene, for vårt prosjekt, var å få det grafiske brukergrensesnittet til å være helt likt den eksisterende versjonen av HEVN. For oss ble det da viktig å kunne gi ut releaser av vår applikasjon med korte mellomrom. Ut i fra disse releasene, måtte vi så ha muligheter til raskt å kunne ta opp nye krav og endringer fra kunden. Da Frode Haug opptrådte som vår "On site customer", var han en person som kunne bistå oss med dette arbeidet, og som var villig til å komme med innspill.

Disse faktorene var med på å gjøre at vi valgte XP som systemutviklingsmodell. Denne utviklingsmodellen gir mulighet for raskt å levere kjørbar kode, uten mye utviklingsarbeid i

forkant. Å raskt kunne gi kunden en kjørbart kode, hvor han fikk se det grafiske brukergrensesnittet, gav han tro på at produktet ville bli som han ønsket.

I XP er kunden i fokus, og ”On site customer” er et viktig begrep. Ved å få kunden til å ta avgjørelser omkring applikasjonens utvikling, er man sikret at applikasjonen blir slik kunden ønsker. Vår oppgave som utviklere blir da å benytte våre tekniske kunnskaper for å guide kunden til å ta de riktige valgene, ut ifra denne kundens behov. I en slik prosess er det viktig at kunden er engasjert og villig til å ta avgjørelser, noe som ikke har vært noe problem i vårt prosjekt.

Fra skolens side, var det krav om at alle hovedprosjektene skulle levere inn dokumenter med planer over prosjektets fremdrift. Det ble da valgt å benytte et *Gantt-skjema* og en Milepælplan. I XP er det ikke lagt opp til å planlegge langt frem i tid, da disse planene allikevel ofte viser seg å bli feil. Prosjektgruppa valgte å levere disse to rapportene som en pekepinn på hva vi trodde ville bli prosjektets fremgang.

7.1.2. TIDSPLANER

7.1.2.1. GANTSKJEMA

Det ble satt opp et Gantt-skjema i uke 4, og dette finnes som vedlegg [B]. Som skrevet i kapittel 7.1.1, har ikke arbeidet med XP gjort det naturlig å arbeide etter denne planen. Vi har ved avslutning av prosjektet satt opp en ny og revidert utgave av Gantt-skjemaet slik det ble ved prosjektslutt. Som forventet ble det noen avvik fra den opprinnelige planen. Allikevel har mange av aktivitetene blitt ferdig til omtrent estimert tid. Det endelige Gantt-skjemaet er lagt med som vedlegg [B].

Videre i dette underkapitlet er det vurdert hvordan arbeidet med Gantt-skjemaets ulike aktiviteter har foregått, og forklart hvorfor det endelige resultatet ble som det ble. Aktivitetene i det opprinnelige Gantt-skjema var som følger:

”Valg av verktøy”

Det var satt av to uker til valg av verktøy. Dette fordi valg av verktøy allerede hadde blitt litt vurdert, og at det ikke var mange ulike valgmuligheter. Denne aktiviteten gikk etter planen.

”Intern opplæring”

Å sette denne aktiviteten til sju uker, var ment å dekke tiden som trengtes for å lære oss å bruke verktøyet, og å sette oss inn i Java-programmeringen. Det viste seg at det ikke lot seg gjøre å lære et så stort utviklingsverktøy som JBuilder på sju hektiske uker. Allikevel hadde vi god forståelse av verktøy etter estimert opplæringstid, og det var da ikke noe problemer å bruke verktøyet aktivt.

”Forprosjektrapport”

At tidsforbruket til denne aktiviteten ble satt til tre uker skyldes at vi hadde en tidsfrist, hvor denne rapporten måtte leveres. Denne fristen ble overholdt, selv om oppdragsgiver kommenterte enkelte punkter i rapporten som måtte endres.

”Kravspesifikasjon”

Tre uker på å lage kravspesifikasjon var tilstrekkelig, da vi i programmeringens første fase konsentrerte oss om kopiere funksjonalitet fra den allerede eksisterende HEVN. Vi kunne da benytte den gamle kravspesifikasjonen. Slik arbeidet i XP har vært, burde egentlig denne aktiviteten også vært like lang som realiseringstiden for applikasjonen, siden kravspesifisering har vært et kontinuerlig arbeid.

”Databasemodulering”

For å få databasen lik den i HEVN, krevde ikke så lang tid som estimert i skjemaet. Likevel kunne vi ikke si oss helt ferdig med databasemoduleringen, da nye krav underveis skapte behov for nye tabeller og attributter.

”Realisering av HEVN-J”

Å estimere realiseringen av HEVN-J til 16 uker, ble gjort fordi innleveringsfristen for prosjektet satte en naturlig stopper for hvor lenge arbeidet med prosjektet kunne pågå. Det som er sikker når dette skrives, er at det vil bli levert en applikasjon som fungerer som HEVN, og at mye ny funksjonalitet har blitt lagt til. Realisering av HEVN-J fortsetter så lenge enten vi, eller noen andre gjør endringer på koden.

”Testing av HEVN-J”

Testingen av HEVN-J startet en uke før realiseringen, dette fordi arbeidet med realiseringen ikke kunne starte før vi hadde testet den eksisterende versjonen og funnet ut hvordan denne virket. I XP er det lagt opp til å skrive testene før man lager koden. Arbeidet med testing ble i hovedsak å bekrefte at vår versjon fungerte på samme måte som HEVN, samt å teste ny funksjonalitet.

”Realisering av administratorverktøy”

Dette verktøyet fikk et annet navn før vi kom så langt som å lage det. Det har siden fått navnet konfigurasjonsverktøy. Det kom raskt på plass en bra versjon av konfigurasjonsverktøyet som kunden var fornøyd med. I dette verktøyet var det heller ikke ønske om mye ny funksjonalitet, derfor kunne dette arbeidet avsluttes tidlig. Det ble brukt seks uker på å realisere denne, en uke mindre enn planlagt. Det ble likevel aktuelt med noen forandringer i dette verktøyet senere i utviklingsfasen, da nye krav i HEVN-J førte til at også konfigurasjonsverktøyet måtte ha ny funksjonalitet.

”Testing av administratorverktøy”

Testing av konfigurasjonsverktøyet ble satt opp til åtte uker, også denne aktiviteten ble ferdig en uke tidligere enn planlagt. Arbeidet med denne aktiviteten foregikk på samme måte som aktiviteten ”Testing av HEVN-J”.

”Webinterface”

Her var det satt av fire ukers arbeid i mars for å få denne delen av systemet klart. Arbeidet med denne aktiviteten har ikke blitt prioritert av oppdragsgiver, i forhold til arbeidet med å realisere HEVN-J. Dette arbeidet startet ikke før i slutten av uke 19.

”Testing av webinterface”

Denne aktiviteten ble ikke påbegynt en uke før aktiviteten ”Webinterface” som planlagt. I stedet har den blitt gjort samtidig med denne. Dette skyldes kundens prioriteringer og tidspress i slutten av prosjektperioden.

”Lage brukerveiledning”

Denne aktiviteten ble fjernet av oppdragsgiver. Det ble foreslått at det skulle legges mer vekt på å lage gode hjelpekort istedenfor. Denne aktiviteten er derfor ikke å finne i det endelige Gantt-skjemaet.

”Opplæring av administratorer”

Denne aktiviteten har blitt gjort litt løpende underveis i prosjektet. Totalt tidsforbruk på denne aktiviteten, er som satt opp i skjemaet.

”Sluttrapport”

Arbeidet med denne aktiviteten var satt til å vare i ni uker frem til innleveringsfristen, arbeidet ble påbegynt to uker etter planen. Det var dermed ikke mer enn sju uker på å skrive ferdig sluttrapporten. Dette viste seg å være akkurat god nok tid til at rapporten ble slik vi ville ha den.

”Forberede fremføring”

Dette er en aktivitet som starter etter at denne sluttrapporten er levert. Det er derfor ikke godt å si hvordan dette arbeidet blir, men også her er det en frist å forholde seg til. Dermed må denne aktiviteten være ferdig i henhold til tidsfristen.

”Feilretting”

Denne aktiviteten er ikke påbegynt, og vil ikke starte før etter at prosjektrapporten er levert ved fristen 23.05.02.

Underveis i prosjektarbeidet har det dukket opp nye aktiviteter som det har vært naturlig å ha med i et Gantt-skjema. Vi har derfor lagt disse inn i et nytt Gantt-skjema, dette finnes som en del av vedlegg [B]. Da arbeidet har foregått etter XP, var det naturlig at det ville dukke opp nye aktiviteter underveis i prosjektet. Disse aktivitetene har blitt prioritert av oppdragsgiver, og det har blitt jobbet med dem etter denne prioriteringen. Vi mener at disse aktivitetene har blitt realisert, og nærmere beskrivelse av disse aktivitetene følger under.

”Implementering av databaseadministrasjonsverktøy”

Det var et krav fra IT-tjenesten om at det skulle være en enkel måte å endre passord og oppsett for databasen, vi var derfor nødt til å legge inn denne aktiviteten. Denne aktiviteten ble gjort litt over tid.

”Lage installasjonsveiledning”

Det var et krav fra IT-tjenesten at det ble laget en installasjonsveiledning for hele systemet. Dette for at vår applikasjon skulle bli mest mulig likt et vanlig produkt som kjøpes. Derfor ble det brukt noen dager på å lage dette.

7.1.2.2. MILEPÆLER

Av de milepælene som ble satt opp, var det tre som var av kritisk karakter for prosjektet. Å levere forprosjekt var vår første kritiske milepæl, som greit ble nådd. De to andre kritiske milepælene var å levere sluttrapport, samt å ha fremføring av prosjektet. Dette er milepæler som må nåes, og som vi tror vi kommer til å stå godt forberedt til å møte. Milepæl nummer to, var å få på plass en endelig databasemodell. Denne milepælen ble nådd i god tid, selv om det skjedde enkelte endringer i databasen senere i prosjektet.

Det ble satt opp tre releaser av systemet som milepæler. Grunnen til dette var at det var kritisk for prosjektet å komme i gang med koden til gitte tider. Å nå den første av disse milepælene, første release av HEVN-J, skapte ingen problemer. Milepæl nummer fire, første release av konfigurasjonsverktøy, ble også nådd til riktig tid.

Den eneste milepælen som ikke ble nådd i løpet av prosjektet var første release av webinterface, dette fordi arbeidet med denne aktiviteten ikke ble prioritert av oppdragsgiver.

I ettertid, må vi innrømme at det ikke ble fokuserte så veldig på å nå alle milepælene som ble satte. At det i ettertid viser seg at mange av milepælene ble nådd til riktig tid, i forhold til hva som var planlagt, tyder på at det har vært arbeidet jevnt med hovedprosjektet og at det har vært progresjon i arbeidet. Det skal også sies at de milepæler som ble satt opp ikke var av den typen det skulle være vanskelige å nå.

7.2. KRITIKK AV OPPGAVEN

7.2.1. PROSESSEN

Arbeidet med prosjektet har blitt styrt av XP som utviklingsmodell. Denne modellen har fungert bra på mange områder. Da vi har liten erfaring fra tidligere prosjekter, er det vanskelig å sette fingeren på hvilke deler av arbeidet som har blitt gjort feil. Vi har merket at enkelte deler av arbeidet ikke har fungert helt tilfredsstillende, men det er vanskelig å avgjøre om arbeidet har vært feil ut ifra valgt metode, eller om det er valgt feil arbeidsmetode.

Et av problemene med å benytte XP ved vårt prosjekt har vært at det har vært en fast tidsfrist, i tillegg til faste krav fra oppdragsgiver. Da vår "On site customer" ikke har hatt mulighet til å være en del av utviklingsteamet, og dermed ikke vært fysisk tilstede under hele utviklingsfasen, har det vært vanskelig å arbeide "etter boka" med XP.

I ettertid, ville det vært naturlig å redusere arbeidsmengden mot slutten av prosjektperioden, i forhold til hva som virkelig er gjort. Den totale arbeidsmengden har vært for stor, slik at vi har vært nødt til å arbeide mye mer enn estimert tidsforbruk for et hovedprosjekt. Vi føler det ville vært naturlig å sluppet en versjon av HEVN-J med den funksjonaliteten som fantes i HEVN den 23. mai. Deretter gjøre ferdig all ny funksjonalitet og deretter gi ut HEVN-J på et senere tidspunkt. Å arbeide 12-13 timer daglig i store deler av de siste månedene, kan ha ført til feil i applikasjonen. Feil som muligens hadde blitt unngått ved å ha brukt mer tid på testing og utvikling.

Det vi føler har gått bra underveis, er å skape et fellesskap i gruppen og derigjennom at alle har tatt ansvar for arbeid som har måttet gjøres. Da det har vært en konstant dialog mellom kunden og oss, har dette ført til at det hele tiden har vært oppgaver å ta fatt på. Dette har ført til en jevn jobbing med prosjektet, og vi har aldri måttet vente på avgjørelser fra kunden for å ha noe å gjøre.

7.2.2. PRODUKTET

HEVN-J og de andre deleapplikasjonene er et resultat av den arbeidsprosessen vi har hatt underveis i prosjektet. Vi synes selv at HEVN-J og tilhørende applikasjoner har blitt meget bra. Høye mål for applikasjonen hadde vi satt oss før vi startet utviklingsarbeidet. Disse målene mener vi nå i ettertid at er nådd.

At vi tidlig i prosjektet fikk en testgruppe til å bruke applikasjonene som ble utviklet, gav oss mange tilbakemeldinger. Dette i kombinasjon med aktiv testing fra "On site customer" sin side, førte til at det aldri ble tomt for brukerhistorier å realisere.

Det ble laget en hintboks for visning av skjult informasjon hver dag. Denne hintboksen blir begrenset av bredden på hver dag. Vi skulle i utgangspunktet ha laget en hintboks funksjon som viser klokkeslettene når musen beveges over timeplanen. Da arbeidet med å kode funksjonalitet for hintboksen startet etter at timeplanen var laget "ferdig", viste det seg at dette ble vanskelig. Problemet var at hintboksen til Java tok kontrollen over musen, slik at de funksjonene vi hadde laget for musen ikke ble brukt. En alternativ løsningsmåte var å tegne denne hintboksen opp manuelt, med det resultat at hele timeplanen ville ha måttet bli tegnet

opp på nytt. Dette ville ført til at timeplanen ville ha flimret mye, noe som var uakseptabelt. Det ble derfor valgt å utelate denne funksjonaliteten. Å velge en annen løsningsmetode for hele timeplanen, ville muligens gjort dette mulig.

Det har ikke vært prioritert å arbeide med å optimalisere koden i applikasjonen. Dette er et arbeid som har blitt gjort underveis, uten å bevist ha brukt tid på dette. Vi tror allikevel ikke hastigheten på applikasjonene har blitt merkbart dårligere av dette.

Det har blitt brukt tid på slutten av prosjektet til å lage en *JavaDoc*. Dette er standarden for dokumentasjon av Java programmer. Denne dokumentasjonen finnes på html-format, og ligger på cd-en som følger rapporten.

Vi ble ikke forespeilet noen spesielle installasjonsrutiner ved starten av prosjektet. Senere viste det seg at IT-tjenesten ved skolen stilte strenge krav for hvordan dette skulle gjøres. Dette har ført til merarbeid, og var noe som vi burde ha blitt informert om ved prosjektstart.

Da kunden ikke prioriterte webapplikasjonen, fikk vi liten tid til å utvikle denne. Resultatet mener vi har blitt bra, men vi skulle gjerne ha hatt mer tid under utviklingen av denne.

Kravet om å bruke MSSQL sammen med en gratis JDBC driver har skapt problemer. Dette fordi det kun finnes en gratis driver for MSSQL på markedet. Det er mulig at innkjøp av en sertifisert driver, ville gjort installasjonen av MSSQL databasen mye enklere.

7.3. VIDERE ARBEID

En applikasjon blir sjelden helt ferdigutviklet, noe som vi heller ikke kan si om HEVN-J. Selv om vi føler at applikasjonen har blitt som både oppdragsgiver og gruppen ønsket, finnes det muligheter for videre arbeid. Hovedprosjektarbeidet som vi har gjort, ble foreslått i en bisetning i HEVN sin rapport over mulig fremtidig arbeid. Dette viser hvor vanskelig det er å forutse hva som kan bli gjort med en applikasjon frem i tid. Under er det foreslått noen mulige utvidelser eller endringer av HEVN-J.

Utvikle for autentisering mot *Kerberos*, slik at brukerne i HEVN-J blir administrert sentralt gjennom Windows domenet. Dette vil gjøre administreringen av HEVN-J enklere for systemadministrator.

Lage applikasjonen slik at den kan støtte flere forskjellige databasetyper på en enkel måte gjennom databasekonfigurasjonsverktøyet. Dette vil være aktuelt ved for eksempel videresalg av applikasjonen.

Lage TimeTable-komponenten slik at den er helt frittstående fra resten av applikasjonen. Slik at denne lettere kan brukes i andre applikasjoner.

Legge til nye funksjonalitetskort hvis dette er ønskelig. For eksempel et diskusjonsforum, dette er spesielt aktuelt hvis applikasjonen utvides til bruk på flere institusjoner. Dette bør kunne gjøres enkelt da vår applikasjon er modulbasert.

Utvide applikasjonen med lørdag, søndag og helligdager, slik at applikasjonen kan brukes mer som en personlig tidsplanlegger.

7.4. HOVEDKONKLUSJON

Vi har arbeidet med å videreutvikle den fire år gamle applikasjonen HEVN. Vårt arbeid har gått ut på å lage denne gamle Delphi-versjonen på nytt i Java. I tillegg har vi utvidet applikasjonen med ny funksjonalitet.

HEVN hadde muligheter for elektronisk å registrere ansattes faste og flytende aktiviteter i en timeplan. I tillegg var det muligheter for å registrere personalopplysninger om de ansatte. Det var utviklet et system som gjorde det mulig å sende meldinger mellom de ansatte.

I tillegg til å ha fått implementert denne funksjonaliteten i Java, har vi lagt til den nye funksjonaliteten som var ønsket fra oppdragsgivers side. Dette var blant annet å kunne registrere alle rom og utleibart utstyr ved skolen i egne timeplaner, slik at de ansatte kunne reservere disse rommene eller utstyret ved behov. Vi har laget mulighet for å kunne finne felles ledig tid for de ansatte, slik at det blir lettere å avtale møter ved skolen.

Vi har laget en webapplikasjon av HEVN-J, slik at man kan se timeplanene og personalia for de ansatte. De ansatte ved HiG, har muligheten til å begrense hvilken type informasjon som skal vises i webapplikasjonen, ved hjelp av funksjonalitet laget i HEVN-J. Det har også blitt laget et konfigurasjonsverktøy for HEVN-J, hvor man kan konfigurere databasen som brukes av HEVN-J. Etter IT-tjenestens ønske har det blitt laget et databasekonfigurasjonsprogram som setter opp databasen for første gang.

Selv om vi som gruppe kunne en del om systemutvikling, databaser og programmering fra før, har gruppens deltaker utviklet sine kunnskaper innen de fleste av disse fagområdene. Det har også vært lærerikt å arbeide i en prosjektgruppe over lengre tid. Vi har for første gang i skolesammenheng arbeidet i en prosjektgruppe med så mye arbeidspress, forventninger og alvor. Disse stressfaktorene, tror vi kan være nyttige å ha med seg ut i arbeidslivet, og er kanskje noe av det viktigste vi har lært under prosjektet.

7.5. EVALUERING AV GRUPPENS ARBEID

7.5.1. INNLEDNING

Gjennom tre år på HiG, har gruppen arbeidet sammen på flere prosjekter. Gruppens medlemmer ble først kjent med hverandre under et prosjekt i faget ”Økonomi og prosjektstyring”. Siden har hele eller deler av gruppen arbeidet sammen i ulike fag. Disse tidligere arbeidene har gitt gode resultater, og vi følte at det var naturlig å arbeide sammen også på hovedprosjektet. Vi kjente hverandres arbeidsmetoder og kompetanse relativt godt før vi startet på prosjektet. Dermed har vi lettere kunnet fordele arbeidet mellom gruppens deltakere, slik at hver enkelt fikk en problemstilling de kunne håndtere og hadde glede av å utføre. Det har vært mindre morsomme arbeidsoppgaver underveis i prosjektet, men også disse oppgavene har vi fått fordelt uten surmuling og klager.

7.5.2. ORGANISERING

Hovedprosjektgruppen har fra starten av hatt en uformell organisering. Dette var naturlig, siden deltakerne kjente hverandre så godt fra før. Da det var et krav om at det skulle være en prosjektleder på alle hovedprosjekter, ble dette valget tatt litt uformelt. Prosjektlederens rolle har i hovedsak gått ut på å motivere de andre deltakerne i tunge arbeidsperioder. Gruppelederen har ikke alene vært gruppens ansikt utad. Møter med eksterne aktører har blitt fordelt etter hvem som sittet med størst kompetanse. Da vi har delt arbeidet inn i ulike User stories/ansvarsområder, har det vært naturlig at deltakerne har representert gruppen innen sitt ansvarsområde.

Vi har ikke hatt noen problemer med vår flate organisasjonsstruktur.

7.5.3. FORDELING AV ARBEIDET

Som det fremgår av prosjektgruppens organisering i kapittel 7.5.2, har det vært en uformell arbeidsgang under prosjektet. Dette har også hatt sin innvirkning på fordelingen av arbeidet. Vi har forsøkt å fordele arbeidet etter den enkelte gruppedeltakers kompetanse og ønsker. Mindre interessante oppgaver, har blitt forsøkt fordelt ut ifra den enkeltes arbeidsmengde og hvor mange slike oppgaver man har utført tidligere. Oppgaver som ingen på gruppen umiddelbart hadde kunnskaper til å løse ble fordelt etter ønske. Da noen av disse oppgavene for enkelte var interessante, fikk man nå mulighet til å sette seg inn i nye fagområder. Hvis ingen umiddelbart påtok seg slike oppgaver ble disse fordelt på to deltaker, slik at ikke en person alene var ansvarlig for å lære noe ”uinteressant”. Denne fordelingen har ført til at prosjektarbeidet hele tiden har hatt interessante problemstillinger det har vært givende å arbeide med.

7.5.4. PROSJEKT SOM ARBEIDSFORM

Vi har nå gjennomført et prosjektarbeid, som i omfang ikke har vært i nærheten av noe vi har gjort tidligere. Selv om vi har hatt flere prosjekter ved skolen tidligere, har ingen av disse på langt nær vært så krevende som dette. Disse prosjektene har allikevel på ulike måter gitt oss erfaringer som vi har benyttet oss av i vårt hovedprosjektarbeid. Den største forskjellen på hovedprosjektet kontra tidligere prosjekter er arbeidsmengden. Vi har nå for første gang

arbeidet fulltid med et prosjekt. Dette har gitt prosjektarbeidet nye dimensjoner. Det å ta hensyn til at folk er slitne, og har gode og dårlige dager, merkes mye bedre i et slik prosjekt enn i de mindre "leke" prosjektene vi har hatt tidligere.

Dette prosjektet har også stilt store krav til at hvert enkelt gruppemedlem har tatt ansvar for sine påtatte arbeidsoppgaver. I et så stort prosjekt som dette har det ikke vært mulig for enkelte av gruppens deltaker å dra lasset alene, noe som ofte kan skje i mindre prosjekter.

Selv om arbeidsmengden har vært stor, har det også vært muligheter får å planlegge arbeidet individuelt. Selv om en av gruppens deltaker ikke har kunnet arbeide på skolen en dag, har det hele tiden vært mulig å gjøre dette arbeidet senere på kvelden eller i helgene.

Prosjektet har vært skånet for interne stridigheter. Dette har vært med på å holde arbeidet i gang, noe som har ført til at vi har kunnet levere et bra prosjekt. Det har selvfølgelig vært uenigheter om når og hvordan vi skulle løse oppgavens mange problemer. Dette tror vi i ettertid bare var sunt ved at vi fikk luftet flere alternativer. Allikevel har ikke disse uenighetene vært verre enn at vi har blitt enige om en løsning som forhåpentligvis har vært til det bedre for alle parter.

7.5.5. SUBJEKTIV OPPLEVELSE AV HOVEDPROSJEKTET

Å gjennomføre et hovedprosjekt er det vi har gjort ved HiG som minner mest om det virkelige arbeidslivet vi møter om kort tid. Selv om vi har hatt mange nyttige fag og prosjekter tidligere, er det ingen av disse som har vist oss så mange sider av IT-bransjen som vi nå har opplevd. Gjennom å jobbe full arbeidsuke fra januar til ut mai, har vi fått følt på kroppen hva det vil si å jobbe fulle arbeidsdager. Selv om det har blitt mye overtid i siste del av mars, april og mai, synes vi arbeidet har vært interessant. Noe som vi håper viser seg i produktet vi nå skal levere.

Da tre av gruppens deltakere hadde hatt mye systemutvikling, var dette et fagområde vi følte oss relativt sikre på. Vi hadde gjennomført et systemutviklingsprosjekt tidligere, men dette benyttet RUP som rammeverk. Derfor var det utfordrende og spennende å ta fatt på et XP prosjekt.

Ved å benytte XP kom vi tidlig i gang med kodingen som var utfordringen i prosjektet vårt. Vi var fra starten klar over at ikke alle gruppens medlemmer hadde like stor interesse for å skrive kode. Derfor ble arbeidsfordelingen i prosjektet styrt av dette. I ettertid ser vi at dette kanskje var like greit, da det har vært mye arbeid med andre aspekter av prosjektet. Design av brukergrensesnitt, tegning av ikoner og skriving av rapporter er arbeid som også har tatt mye tid, og som vi dermed har fått fordelt på en enkel måte.

Vi er glade for at vi fikk tildelt et grupperom ved skolen. Selv om vi har måttet flytte til et annet rom i løpet av prosjektperioden, og at det tok noen uker før vi fikk nettverkstilgang, ser vi ikke helt hvordan vi skulle kunne fått gjennomført dette prosjektet uten et felles rom. Det hadde blitt umulig å gjennomføre vårt prosjekt på en studenthybel. Et annet alternativ kunne vært skolens datarom, men disse er heller ikke spesielt egnet for et XP prosjekt. Det kreves en del diskusjoner og planlegging, og dette er nok ikke spesielt populært på disse rommene.

Samarbeidet med veileder/"On site customer" har fungert meget bra. Frode Haug har vist stor interesse for prosjektet og vært villig til å ta avgjørelser underveis, noe som har vært nødvendig for at prosjektet skulle bli vellykket. Vi har også opplevd de andre ansatte ved skolen som særdeles interesserte i at det skulle komme en ny versjon av HEVN. Dette har vært med på å inspirere oss og gi oss troen på at det vi har arbeidet med virkelig er et ønsket produkt. Denne interessen fra de ansatte har også fått oss til å føle at vi arbeider med et virkelig prosjekt, mye likt hva vi kanskje vil oppleve i en arbeidssituasjon.

Oppdragsgiver har hatt hovedansvaret for å holde kontakt med brukergruppen. Dette er et arbeid som kunne vært interessant også for oss å gjøre i større grad, allikevel føler vi at vårt behov for kundekontakt har blitt dekket gjennom vår "On site customer".

8. LITTERATURLISTE

Bøker og websider som har vært brukt som referanser og oppslag under dette hovedprosjektet:

1. Deitel & Deitel "Java – how to program". Prentice Hall. Third Edition 1999
2. Bruce Eckel "Thinking in Java". Prentice Hall. 1998.
3. Ivor Horton "Beginning Java 2". Wrox Press. 2000
4. Kent Beck & Martin Fowler "Planning extreme programming". Addison-Wesley 2001
5. Craig Larman "Applying UML and Patterns". Prentice Hall. 2002
6. Rune Bjerke, Vegard Brox, Thomas Lundbye og Tormod Rønning "HEVN- Hvem Er Hvor Når" Hovedprosjektrapport HiG. 1998
7. Don Wells. Arbeidsmetoder i XP. 2002
<http://www.extremeprogramming.org/rules.html>
8. Don Wells. Når bruke XP. 2002
<http://www.extremeprogramming.org/when.html>
9. Zero G. Installasjonsprogram. 2002.
<http://www.zerog.com>