

Examensarbete

Webbapplikation för hantering av driftsdata från fartygsmaskineri

av

Hugo Egerström

LITH-IDA-EX-ING--03/014--SE

2003-09-17

Handledare: Kalle Bohman
Examinator: Kristian Sandahl



TEKNISKA HÖGSKOLAN
LINKÖPINGS UNIVERSITET

Institutionen för datavetenskap

Sammanfattning

”Stena Jutlandica: Operation Data” är namnet på en webbapplikation för att hantera driftsdata från fartyget Stena Jutlandicas maskinrum. Applikationen ska ersätta ett excelblad som används sedan flera år tillbaka. En av anledningarna till att man samlar in driftsdata är att fel kan upptäckas snabbare. Exempel på driftsdata som genereras i maskinrummet är gångtider för olika motorer, nivåer i olika tankar och vattenåtgång. Driftsdata lagras i en databas på en server som står i land på Stena Lines IT-avdelning. Ombord på Stena Jutlandica kommer man åt webbapplikationen genom en webbläsare via fartygets fasta Internetuppkoppling. Den här rapporten beskriver hur utvecklingsarbetet fortgått och hur den färdiga applikationen fungerar.

Abstract

”Stena Jutlandica: Operation Data” is the name of a webb application with the purpose of managing runtime data from the machinery of the ship Stena Jutlandica. The application will be replacing an excel spreadsheet which has been in use for several years. One of the reasons to collect runtime data is that faults can be discovered more quickly. Examples of runtime data generated from the machinery is runhours for different engines, levels in different tanks and water consumption. Runtime data is stored in a databas on a server in Stena Lines IT-department. On board Stena Jutlandica the webb application is accessed thru a webb browser via a sattelite which keeps the ship on line. This report describes how the development has proceeded and how the resulting application works.

Innehållsförteckning

1	Inledning	4
1.1	Bakgrund	4
1.2	Syfte	4
1.3	Avgränsning	5
1.4	Metod	5
2	Förberedande dokument	7
3	Arbetsgång	8
4	Teknisk beskrivning	9
4.1	Webbapplikation eller traditionell klient	9
4.2	Valet av Java/JSP	9
4.3	Säkerhet	9
4.4	JSP-sidorna	10
4.5	Klasser	12
4.6	Inställningsfil	13
4.7	JavaScript	13
4.8	Formatmall (Cascading Style Sheets)	13
4.9	Databas	13
4.10	Övrigt	14
5	Vidareutveckling.....	15
6	Resultat och reflektioner	16
7	Slutsats	17
	Referenslista.....	18
	Termer och förkortningar.....	19
	Bilaga 1. UML-diagram databas.....	20
	Bilaga 2. Specifikation.....	22
	Bilaga 3. Användarhandledning.....	25
	Bilaga 4. Installations- och driftsanvisning.....	33

Tabellförteckning

<i>Tabell 1 Resor</i>	6
<i>Tabell 2 Klasserna i applikationen</i>	12
<i>Tabell 3 Tabellerna i databasen</i>	14

Figurförteckning

<i>Figur 1 JSP-sidornas uppbyggnad</i>	10
<i>Figur 2 Vart skickas data från formulären?</i>	11

1 Inledning

”Stena Jutlandica: Operation Data” är en applikation för att hantera statistik från Stena Jutlandicas maskinrum. Applikationen ersätter delvis det excelblad som nu används med ett webbaserat gränssnitt mot en databas. Stena Jutlandica är ett så kallat Ro/Pax fartyg, alltså ett kombinerat frakt- och passagerarfartyg, det trafikerar sträckan Göteborg – Fredrikshavn. Stena Lines huvudkontor och IT-avdelning finns i Göteborg. Handledare är Kalle Bohman, han arbetar som maskinist ombord. I fortsättningen kommer benämningen "applikationen", att syfta på hela systemet alltså databasen och webbgränssnittet samt samspelet mellan det och användarna.

1.1 Bakgrund

I maskinrummet läses vissa mätare av runt midnatt varje dygn. Mätvärdena skrivs upp i en bok som kallas för maskinkladd. En ny maskinkladd påbörjas varje månad och den gamla stoppas undan och lagras en bestämd tid. För att lättare kunna överblicka de värden som samlas in varje dygn i maskinrummet har Kalle Bohman gjort ett antal excelblad. Det finns ett excelblad för varje månad. Varje dygn som passerar förs in i månadens excelblad. Det är gångtider och smörjoljeförbrukning för motorerna, vattenåtgång för smörjolje- och bränsleseparatorer, nivåer i bränsletankar och sludgetankar, gångtid för värmepanna och kompressorer. Excelbladet beräknar bland annat gångtider per dygn och smörjoljeförbrukning per timme utifrån de inmatade värdena. Varje blad summerar också månadens totala gångtider och förbrukningar. Excelbladen för varje månad är kopplade till en årssammanställning där flera värden summeras och beräknas för att ge en översikt över hela året. Genom att börja använda excelbladen skapades regelbundna rutiner för att granska alla värden i samband med att man fyllde i nya värden för dygnet som gått. Det har gjort att man kunnat se trender och tidigare kunnat upptäcka fel eller läckor i maskineriet. Användandet av excelbladen skapade också en uppföljning av den gångna månaden som inte skett tidigare eftersom den tidigare månadens maskinkladd inte funnits tillgänglig lika enkelt som excelbladen. Excelbladen har använts i flera år och har efterhand utökats för att omfatta fler och fler mätvärden. De flesta värden läses av, eller pejlats, via ett speciellt datorsystem från det danska företaget Lyngsø Marine A/S. Det har ingen koppling till övriga datorsystem ombord utan är bara till för att samla in data från sensorer ute i maskinrummet. I praktiken betyder det att många mätvärden läses av från en skärm och matas in i ett excelblad som finns på en skärm bredvid. En del av värdena samlas dock in från analoga mätinstrument och måste pejlats för hand ute i maskinrummet.

1.2 Syfte

Användningen av excelbladen har flera nackdelar. De kan bara användas av en person i taget vilket gör att åtkomsten begränsas. Det är inget stort problem eftersom excelbladen har så pass få användare ombord. Det finns dock ett visst intresse av att ha tillgång till driftsdata också i land, framförallt hos inköpsavdelningen. Idag skickar varje fartyg in en rapport dit där de anger hur mycket de har förbrukat, det sammanställs till en rapport för alla fartyg. En sådan rapport skulle kunna genereras automatiskt om alla Stena Lines fartyg använde en applikation som den här. Att använda excelbladen på motsvarande sätt är inte praktiskt möjligt. Excelbladen distribueras som delade filer via det lokala windowsnätverket det gör att det inte finns

någon åtkomstkontroll, vem som helst kan gå in och ändra. Det finns inga rutiner för att säkerhetskopiera excelbladen, de kan försvinna genom att någon av misstag tar bort dem eller genom att den hårddisk de lagras på går sönder. Genom att använda en databas i land istället kan säkerhetskopieringen införlivas med existerande rutiner på Stena Lines IT-avdelning. Varje månad påbörjas ett nytt excelblad som måste skapas för hand, det är visserligen bara antalet dagar och månaden som behöver ändras men det är ändå ett extra arbete. Årssammanställning utgör också ett extra arbete eftersom den för hand måste kopplas till varje månads excelblad.

Syftet med applikationen är i första hand att förenkla och minska arbetsbelastningen med den dagliga hanteringen av driftsdata från maskinrummet. I andra hand syftar den också till att visa på de möjligheter som blir tillgängliga när driftsdata lagras i en databas. Med detta i åtanke har en av de bakomliggande tankarna under utvecklingen varit att skapa en så generell arkitektur som möjligt. En arkitektur som gör att applikationen enkelt kan anpassas för användning i ett annat fartygs maskinrum eller byggas ut med ytterligare funktioner när behov uppstår.

1.3 Avgränsning

För att begränsa applikationens omfattning motsvarar den bara vissa delar av excelbladet. De delar av excelbladet som utelämnats är de som finns under flikarna med rubrikerna, ”Temperaturer Hjm”, ”AVGASTEMP” och ”Prop-hylsor”. De här delarna rapporteras inte regelbundet och används bara ibland. I excelbladet går det att koppla en kort notering till vilken ruta som helst. Dessa noteringar används till exempel för att notera att ett visst underhåll har utförts någonstans. Den här funktionen har utelämnats eftersom den sällan används och skulle kräva förhållandevis mycket arbete att implementera. Under planeringsfasen fanns planer på en förenklad variant av noteringar men den utelämnades också efter hand på grund av den erfordrade tidsåtgången.

1.4 Metod

Arbetet har gjorts hemifrån samt under tre resor till Göteborg. Resorna har gett tillfälle att prata med användarna och utvärdera nya versioner av programvaran och dess gränssnitt. Under resorna har undertecknad bott ombord på Stena Jutlandica. Maskinisterna på Stena Jutlandica arbetar i tvåskift en vecka i taget och är sedan lediga i tre veckor. Alla resor har gjort under handledarens arbetstid. I tabell 1 nedan följer en förteckning över resorna med en kort beskrivning av vad som hänt vid varje tillfälle.

Tabell 1 Resor.

Resa	Datum	Beskrivning
1	16-17 april	En prototyp till ett gränssnitt i html togs med, den var en bra grund att diskutera kring. Personalens arbete i maskinrummet observerades, vilka rutiner de hade och hur miljön såg ut.
2	2-4 maj	Den första testversionen är klar och hanteringen av användare fungerar. Tabellen med huvudmaskinerna, ME, fungerar också hjälpligt. Tabellens funktion gav grund för ytterligare samtal.
3	27-28 maj	I den här testversionen fungerar allt förutom årssammanställningen (Annual report). Applikationen tas i provdrift parallellt med excelbladet för att hitta räknepel och dylikt.

En nackdel med att inte arbeta på plats är att kommunikationen med användarna minskar. Löpande kontakt har hållits över telefon med handledaren men det är bättre att ha tillgång till flera användares åsikter och vara på plats. En fördel med besöken ombord är att de ger en naturlig iteration åt utvecklingsarbetet, varje ny version med efterföljande besök och utvärdering har ger nya insikter. Personalen ombord är intresserad och engagerad i utvecklingsarbetet, det har gjort att besöken ombord alltid givit mycket återmatning. Stena Data var till stor hjälp och ställde en server till förfogande och har hjälpt till att installera nya versioner under slutfasen av arbetet, efter det sista besöket.

2 Förberedande dokument

Till att börja med skrevs en specifikation. Det gjordes främst för att samla ihop alla fakta som fanns då. Arbetet med databasens utformning påbörjades också tidigt, till en början genom att göra UML-diagram, senare direkt i databasen när det kom fram att UML-diagram enkelt kunde genereras från databasen. Tabellerna i den färdiga databasen motsvarar i princip de tabeller som finns i gränssnittet. Det betyder att arbetet med databasen lika gärna kunde ha gjorts parallellt med utvecklingen av gränssnittet. Det beror troligen på databasens relativt enkla uppbyggnad, utan relationer eller andra konstruktioner direkt i databasen. I ett större projekt med en mer komplex databas hade det säkert varit användbart att designa databasen för sig. I planeringen ingick också att göra UML-diagram över användningsfallen. Efter det första besöket på Stena Jutlandica drogs dock slutsatsen att det inte var nödvändigt eftersom applikation skulle ersätta en existerande applikation nämligen excelbladet vars roll i arbetet ombord redan var väl känd. För att kunna införa ett nytt system som applikationen så smärtfritt som möjligt måste applikationen i möjligaste mån passas in på excelbladets ställe i systemet. För att göra det gällde det att få användarna att känna igen sig i gränssnittet.

3 Arbetsgång

Först gjordes en studie av excelbladen på Stena Jutlandica och arbetet med dem ombord. På Stena Germanica fanns ett liknande system med excelblad som också studerades. I studien ingick även ett annat examensarbete som också hade målet att utveckla ett datorprogram för att hantera driftsdata. Det första dokumentet som skrevs var specifikationen (se bilaga 2) för att fastställa och avgränsa de krav som ställdes på applikationen och för att bilda en uppfattning om i vilket sammanhang applikationen användes. Specifikationen formulerades medvetet väldigt löst eftersom det inte fanns någon klar bild av vad som skulle göras vid det laget. Efter specifikationen gjordes en prototyp till ett gränssnitt i HTML. Prototypen saknade mycket funktionalitet men var ändå till stor hjälp för att förmedla hur gränssnittet skulle kunna se ut. Det visade sig vara väldigt bra att ha prototypen som utgångspunkt att diskutera omkring. Det går inte att bara fråga användarna vad de vill ha utan att de har något att relatera till. Visserligen fanns excelbladet sedan tidigare men det skiljer sig en hel del från hur en version för webben fungerar. Det första målet var att få ramverket med inloggningen och hanteringen av användare att fungera. Det fungerade i den version som togs med till den andra resan. Tabellen för huvudmaskinerna fungerade delvis också då, där kunde värden bara matas in på sista raden och inga beräkningar gjordes på inmatade värden. Den versionen innehöll också ett förslag till hur ändringar skulle visas och hanteras. Under den andra resan kom flera nya saker fram, bland annat att en deadline var önskvärd. Med en deadline menas att en viss tid efter midnatt när det blir möjligt att rapportera värden för det gångna dygnet så låser applikationen dygnet, dvs före midnatt dygnet därpå. Rapporteras inte det gångna dygnets värden i tid går det inte att rapportera alls. Det framkom också att ändringarna inte är så intressanta att de alltid ska synas. Tillsammans med användarna togs då den lösning fram som finns nu, med en stjärna som indikerar att en post är ändrad. Den version som var med till den tredje resan var fullt fungerande förutom att delar av sammanställningen för hela året (Annual report) saknades. Det fanns också en del mindre problem med layouten och utskriften kvar att lösa. Efter den tredje och sista resan skickades nya versioner av applikationen med e-post till personalen på Stena Data som installerade dem. Den 2 juni var applikationen så mogen att den fick heta version 1.0. Den installerades och började användas på prov parallellt med excelbladet. I version 1.0 fanns all planerad funktionalitet med, dessutom hade ett nytt verktyg lagts till. Det nya verktyget bestod i ett antal formulär som används för att transportera in värdena när man börjar använda applikationen. Det underlättar övergången från excelbladet eftersom mätserierna för den första månadens totala gångtider, förbrukningar och så vidare kan initieras med andra värden istället för noll, det gör att de totala förbrukningarna stämmer. Ytterligare en version, 1.01 togs fram och installerades den 12 juni. I version 1.01 är alla värden i ändringsformuläret ifyllda från början, det sparar en del arbete och förhindrar nya inmatningsfel. I version 1.01 fungerar också utskriften som de ska och bredden på tabellernas rubriker är alltid densamma som bredden på de motsvarande kolumnerna som scollar därunder. Applikationen har använts på prov sedan början av juni och verkar i stort fungera bra. Under utvecklingen har IntelliJ IDEA använts för att skriva Javakoden, Macromedia Dreamweaver har använts för utformningen av gränssnittet och SQL Server Enterprise Manager har använts tillsammans med Access för att bygga databasen. Noteras bör att en hel utvecklingscykel med faserna utveckling, implementation och testning har genomförts i och med att användningen av applikationen kan utvärderas när provperioden är över.

4 Teknisk beskrivning

Applikationen är en webbapplikation. Webbsidorna genereras av en webbserver och fylls med data från databasen. Webbservern som används är den fristående versionen av Tomcat 4.1.24 tillsammans med Java 2 SDK 1.4.1_02 från Sun. Varje webbsida i applikationen är uppbyggd av ett antal Java Server Pages (JSP-sidor). För att kunna dela på kod mellan de olika JSP-sidorna används sex gemensamma klasser. SQL Server 2000 från Microsoft används för att hantera databasen. Alla data skickas, hämtas och manipuleras via SQL-frågor. Valet av SQL Server för databasen kommer av att Stena redan använder SQL Server för andra tillämpningar. Vid installationen av webbapplikationen förutsätts att databasen körs på samma server som webbservern, det är dock inte nödvändigt. Applikationen kan konfigureras för att använda en databas på en annan server. Under hela utvecklingsarbetet har tanken, på att applikationen ska kunna anpassas för andra fartyg, hela tiden funnits. Det har resulterat i att källkoden utgör en bra utgångspunkt och ram för en eventuell anpassning till ett annat fartyg. Applikationens källkod kan, i så fall, till stora delar återanvändas.

4.1 Webbapplikation eller traditionell klient

En traditionell klient, alltså ett vanligt program, ger större möjligheter att fritt utforma gränssnittet. Gränssnittet i applikationen ställer dock inga särskilda krav vilket gör att de ökade möjligheterna inte betyder så mycket. En traditionell klient ger dock ett snabbare gränssnitt med bättre respons. Det är en fördel jämfört med en webbapplikation där responstiden till stor del beror på uppkopplingens latens. Den stora fördelen med en webbapplikation är att den går snabbt och enkelt att distribuera. Valet att göra en webbapplikation gjordes med tanke på att arbetet skulle ske delvis på distans och att uppkopplingens latens till vissa delar skulle påverka också en traditionell klient.

4.2 Valet av Java/JSP

Det fanns främst två skäl till att JSP valdes framför, till exempel, ASP eller PHP. Det första skälet, förtrogenhet med JSP sedan tidigare, motiveras med bristen på tid och med vikten av att få fram en fungerande applikation. Det andra skälet, det fanns en mogen drivrutin för databasen i Java, möjliggör valet av JSP och utesluter flera andra alternativ. Huvudsyftet med arbetet är att utveckla en applikation. Därför har tiden som lagts på att utreda de ingående komponenternas lämplighet begränsats till en kortare utredning.

4.3 Säkerhet

Applikationen är bara tillgänglig i Stena Lines intranät, därför har relativt lite tid lagts på säkerheten. Eftersom inloggningsuppgifterna för databasen kan utläsas på första sidan i applikationen, förutsätts att användarna inte är intresserade av att göra otillbörliga ändringar i databasen. Inloggningsfunktionen är ändå uppbyggd för att inte ge några kryphål där man kan ändra i databasen utan att logga in. Det gör det möjligt att höja säkerheten genom att ta bort inloggningsuppgifterna för databasen. För att höja säkerheten skulle också inmatningskontrollen på serversidan behöva förbättras. I nuläget sker huvuddelen av kontrollen hos klienterna med JavaScript.

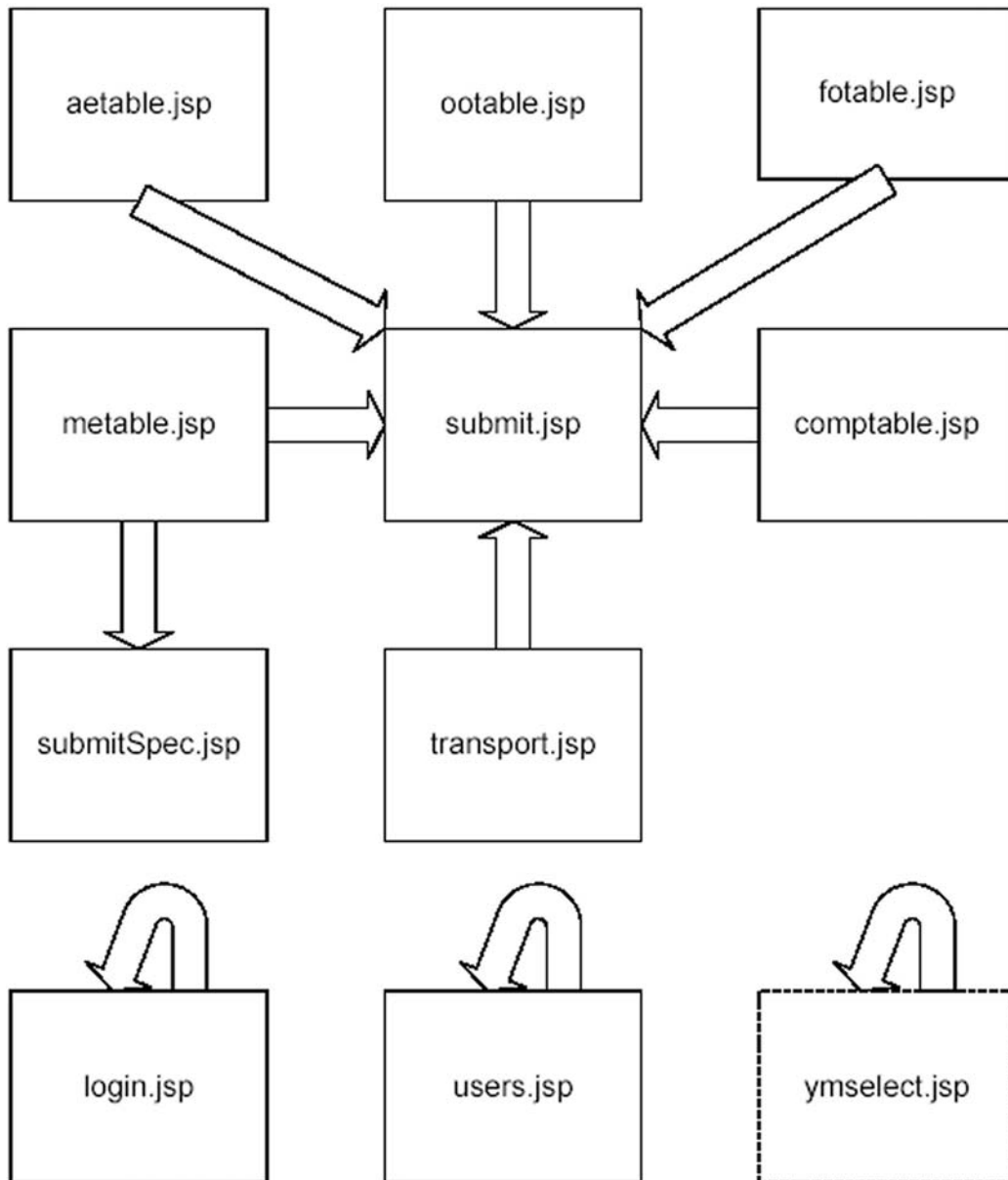
4.4 JSP-sidorna

Det finns 21 olika JSP-sidor. Nio av dem är de huvudsidor som bygger upp själva applikationen, de kan nås via länkarna i menyn som finns överst på varje sida. I figur 1 nedan är filnamnen på huvudsidorna markerade med fetstil.



Figur 1 JSP-sidornas uppbyggnad.

JSP-sidorna är uppbyggda på tre sätt. Det första sättet innebär att `menu.jsp` är inkluderad längst upp på sidan. Det andra sättet används för tabellerna man matar in data i. Det sista sättet är en variant av det andra sättet där `ymselect.jsp` får ett argument så att inte länkarna till årets tolv månader visas. I figur 1 motsvarar de tre största rektanglarna de tre olika uppbyggnadssätt som finns. Namnen på de JSP-sidor som kan nås från huvudmenyn, den som visas av `menu.jsp`, är markerade med fetstil. För att kunna välja år och månad skriver `ymselect.jsp`, där det behövs, ut en meny under `menu.jsp`.



Figur 2 Vart skickas data från formulären?

För att ta emot data från klienten används en mängd olika formulär på de olika sidorna. Figur 2 visar vilka sidor det finns formulär på och var de sedan skickar data från formulären. De sidor som tar emot data skickar tillbaka användaren till den sida data kom ifrån utom i vissa fall. Undantagen är login.jsp som skickar användaren vidare till me.jsp och submit.jsp som när den får data från transport.jsp skickar användaren vidare till users.jsp. Det formulär som visas av ymselect.jsp, en lista där man kan välja år, behandlas direkt med en rad JavaScript som skickar användaren vidare till samma sida men med det valda året som argument. Två av sidorna postar data till sig själva, login.jsp och users.jsp.

4.5 Klasser

Förutom JSP-sidorna består applikationen också av sju olika klasser. Tre av klasserna, JDBCUserDatabase, JDBCOperationDatabase och JDBCOperationSpecial sköter kopplingen till databasen. JDBC är en förkortning för Java Database Connectivity och namnet på det API de tre klasserna använder för att ansluta till databasen. Drivrutinen som används är Microsoft SQL Server 2000 Driver for JDBC, Service Pack 1, Version 2.2.0029.

Tabell 2 Klasserna i applikationen.

Klass	Beskrivning
JDBCUserDatabase	Hanterar den tabell i databasen som heter Users (se tabell 3 för beskrivning). Den används av JSP-sidorna login.jsp och users.jsp och innehåller funktioner för att lägga till och ta bort användare samt för att hämta användardata.
JDBCOperationDatabase	Hanterar de tabeller i databasen som lagrar datummärkt statistik, alltså ME, AE, TanksOO, TanskFO och Compressors. De JSP-sidor som heter något med *table.jsp anropar den här klassen för att hämta data. Formulären på de här sidorna använder submit.jsp för att skicka data till klassen. Standardklassen ArrayList används som containerklass för att skicka och ta emot data.
JDBCOperationSpecial	Hanterar tabellen Melol i databasen. Klassen används bara av submitSpec.jsp, som skickar data, och metable.jsp som hämtar data.
User	Används för att paketera användardata som skickas till och från JDBCUserDatabase av login.jsp och users.jsp. Har fält för användarnamn, lösenord, namn, e-postadress, begynnelse datum och administratörsstatus.
Toolkit	Innehåller statiska funktioner för att hantera inställningsfilen, jutlandica.properties, samt diverse andra funktioner. Funktionen för att skapa en anslutning till databasen, som används av JDBC-klasserna, finns här. Funktionerna har inget logiskt samband utan är samlade här därför att de används av flera olika JSP-sidor och klasser men inte passar in i någon annan klass.
DefaultRB	Implementerar den abstrakta standardklassen ResourceBundle. Används i en funktion som finns i Toolkit för att skapa en ny inställningsfil, jutlandica.properties, med förvalda inställningar. DefaultRB är en förkortning av Default Resource Bundle.
CreatePropertiesFile	En körbar klass som skapar en ny inställningsfil med hjälp av klassen Toolkit.

Beslutet att använda standardklassen ArrayList som containerklass för data till och från JDBCOperationDatabase var tänkt att möjliggöra en mer likartad uppbyggnad av JSP-sidorna så att de fem huvudsidorna skulle kunna dela på kod. I efterhand kan konstateras att inga större vinster gjordes i förhållande till den ökade komplexiteten.

För att göra koden mindre komplex hade det varit bättre att använda klassen `ResultSet`, åtminstone för att hämta data från databasen. När man hämtar ut data ur ett `ResultSet` kan man nämligen använda en kolumns rubrik för att referera till det data man vill ha ut ur tabellen, det går inte med en `ArrayList`. En viss vinst har dock gjorts eftersom de fem huvudsidorna kan dela på kod för att skicka data till databasen, det görs av JSP-sidan `submit.jsp`, vilket inte hade varit möjligt på samma sätt om inte `ArrayList` hade använts.

4.6 Inställningsfil

Applikationen konfigureras via en fil som heter `jutlandica.properties`. Filen hanteras av klassen `DefaultRB` som är en implementation av standardklassen `ResourceBundle`. `DefaultRB` läser värdena för nyckelorden i inställningsfilen. Ett exempel är nyckelordet `DB_NAME` som returnerar namnet på databasen, standardvärdet ”jutlandica” returneras om inget annat angetts i inställningsfilen. När inställningsfilen saknas returneras bara standardvärden.

4.7 JavaScript

JavaScript används för att se till att användarna fyller i alla inmatningsfält som krävs och kunna ge snabb återkoppling om något fattas. Funktionerna för detta finns i filen `checkform.js`. Eftersom Internetanslutningen till Stena Jutlandica går via satellit har anslutningen väldigt hög latens. Det gör att återkopplingen på en handling i gränssnittet är väldigt fördröjd. Det kan göra att en användare klickar flera gånger för att skicka ett formulär vilket resulterar i att samma data läggs till i databasen flera gånger. För att undvika detta låser ett JavaScript formuläret så att bara det första klicket resulterar i att data skickas. JavaScript används också för att sätta fokus i det första inmatningsfältet på `login.jsp`, för att kontrollera formuläret på `users.jsp` och för att begära en ny sida när ett nytt årtal väljs.

4.8 Formatmall (Cascading Style Sheets)

För att enkelt kunna ändra formateringen av gemensamma komponenter på sidorna, till exempel bakgrundsfärgen i kolumnerna för inmatning, används en formatmall, `format.css`.

4.9 Databas

Det finns inga uttryckliga relationer mellan tabellerna i databasen däremot skapas en relation mellan tabellen `Users` och de andra tabellerna av att det alltid är en existerande användares användarnamn som fylls i tillsammans med varje rad i de övriga tabellerna. Undantaget är tabellen `MELol` där användarnamn saknas. I ett tidigt skede av utvecklingsarbetet existerade en relation mellan `Users` och de andra tabellerna men den orsakade fler begränsningar än fördelar. Enligt användarna är det inte så intressant eller viktigt att alltid kunna spåra vem som gjort vilken inmatning. Syftet med inloggningsfunktionen är snarare att begränsa utomstående möjligheter att mata in data och att under en kortare tidsperiod bakåt hålla reda på vem som gjort vilken inmatning. Alla tabeller utom `Users` och `MELol` har tre gemensamma kolumner, `Edate`, `valid` och `Login`. Den första kolumnen, `Edate` (Entry date), lagrar ett datum där år, månad och dag används. Den andra kolumnen, `valid` är till för att hålla reda på

ändringar, den lagrar sant eller falskt. Varje gång en ny rad läggs till i en tabell sätts valid till sant och på alla andra rader med samma datum (Edate) i den tabellen sätts valid till false. På det här sättet sparas alla ändringar. Kolumnen Login lagrar ett användarnamn och fungerar som en slags signatur och skapar den lösa relation som nämns ovan. I tabellen Users finns också en kolumn, Administrator, som lagrar sant eller falskt. Den talar om ifall användaren skall ha tillgång till sidan users.jsp när denne loggar in och därmed kunna lägga till och ta bort andra användare. I Bilaga 1 finns ett UML-diagram över tabellerna i databasen.

Tabell 3 Tabellerna i databasen.

Tabell	Beskrivning
Users	Information om vilka användare som får använda applikationen. Deras login och lösenord osv.
ME	Huvudmaskiner (Main Engines), gångtider, smörjoljeförbrukning och gångtider för propelleraxlar.
AE	Hjälpmaskiner (Auxiliary Engines), gångtider, smörjoljeförbrukning och vattenförbrukning för separatorer.
TanksOO	Sludge & Urea (Tanks Other Oil), nivåer för diverse tankar, mängd rent utsläppt vatten, mängd ilandpumpat förorenat vatten och mängd bunkrad Urea.
TanksFO	Brännoljetankar (Tanks Fuel Oil), nivåer i brännoljetankar, brännoljemätare, mängd bunkrad brännolja och gångtid för värmepanna.
Compressors	Luftkompressorer, gångtider för startluftkompressorer och gångtider för arbetsluftkompressorer.
MElol	Smörjoljenivåer för huvudmaskinerna (Main Engines Lub Oil Levels).

4.10 Övrigt

Tabellen med rubriker i, ovanför en iframe, och tabellen i en iframe ligger skilda åt på olika sidor. För att få bredderna på kolumnerna att stämma överens dem emellan har en shim använts. Det är en genomskinlig bild, en bildpunkt stor, den har filnamnet shim.gif. I varje tabell finns en extra rad där varje cell innehåller en shim. Raderna syns som två tunna extra streck under kolumnernas rubriker. Bredden på shim.gif i img-taggen är satt till den önskade bredden för varje kolumn. Genom att ange samma bredd på shim.gif för motsvarande kolumner i och utanför iframe fås kolumnbredderna att stämma överens.

5 Vidareutveckling

Under arbetets gång har många idéer till förbättringar och vidareutvecklingar av applikationen kommit fram. De önskemål om förbättringar av applikationen som kommit fram gäller främst funktioner som finns tillgängliga i excelbladet men saknas i applikationen. När man fyller i sina värden beräknas de efter hand i excelbladet vilket gör att man lättare upptäcker fel. I webbapplikationen beräknas inga värden förrän alla inmatade värden skickats till databasen. För att korrigera värdena krävs det då att användaren gör en ändring. För att minska antalet ändringar kan dessa beräkningar visas direkt vid inmatningen. För att skapa snabb respons till användaren skulle det med fördel kunna implementeras med JavaScript. Nackdelen med JavaScript är att det är beroende av vilken webbläsare som används det kan man dock bortse ifrån i det här fallet eftersom det är väl känt vilken webbläsare som används. En annan utveckling är att göra det möjligt att ladda hem en tabell som ett excelblad och därigenom få tillgång till alla de funktioner som finns i Excel för att kunna analysera driftsdata närmare.

Applikationen kan konfigureras med en deadline från midnatt och framåt. Efter deadline, som ställs in i hela timmar, är applikationen stängd för inmatning. Standard är att deadline är satt till klockan 2400 dygnet efter, således har deadline då ingen funktion. För att bättre kunna anpassa applikationen för andra fartyg där man till exempel pejlår klockan 1200 skulle man kunna göra det möjligt att konfigurera också starttiden.

Alla Stena Lines fartyg skickar in en rapport över bland annat bränsleförbrukningen varje månad. Rapporterna sammanställs på inköpsavdelningen och ligger till grund för hur stora inköp av bland annat bränsleolja som behöver göras. Det här arbetet skulle förenklas mycket om alla fartyg använde något som liknade den här applikationen. Då skulle alla data finnas i en databas och sammanställningen med alla fartyg skulle kunna genereras automatiskt.

6 Resultat och reflektioner

Applikationen som har utvecklats kommer sannolikt att börja användas ombord på Stena Jutlandica. Det krävs dock ytterligare en period av parallellt användande med excelbladet för att avgöra om det finns några större brister i applikationen. Det som fattas när vid en jämförelse med specifikationen (se bilaga 2) är främsta hanteringen av noteringar för enskilda rutor. I gengäld har en del tillägg gjorts, såsom lagring av gångtider för propelleraxlarna, smörjoljenivåer i huvudmaskinerna per den första i varje månad och en sida för att förenkla övergången från excelbladet till applikationen. Under utvecklingens gång visade sig specifikationen vara bra att ha. Främst för att behålla fokus på de ursprungliga målen med applikationen under samarbetet med användarna. Men också för att kunna släppa nya idéer och funktioner så att applikationen blev klar i tid.

7 Slutsats

Valet att bygga applikationen som en webbapplikation föll väl ut. Gränssnittet fungerar bra och den höga latensen har inte ställt till med några allvarliga problem. Utvecklingen tog ungefär så lång tid som planerat, mycket tack vare att utvecklingsspråket, d v s Java/JSP, inte var något hinder i sig. Den testningen som hittills gjorts ombord på fartyget pekar mot att applikationen fungerar minst lika bra som excelbladen men med de fördelar en webbapplikation för med sig. En av de viktigaste erfarenheterna man kan dra av arbetet är hur viktigt det är att prata med användarna. Bara att vara närvarande när de arbetar och observera arbetsmiljön har gett kunskap om hur applikationen bäst skall utformas. Även när man tror att alla aspekter av applikationens användning och funktion är utredda och diskuterade kommer det fram nya saker. Det är viktigt att ge kontakten med användarna tid, hur förberedd man än är verkar det som om det kommer fram nya saker så fort man är närvarande. En annan erfarenhet är vikten av att tydligt motivera designbeslut i källkodens arkitektur och sedan hålla fast vid besluten fullt ut. Håller man inte fast vid besluten eller tar dem på för svaga grunder kan källkoden bli svårläst och dåligt strukturerad.

Referenslista

Examensarbete: *Datoriserad Maskinkladd*, Jenny Östh & Andreas Sjunnesson, Chalmers Lindholmen Sjöbefälsskolan, Göteborg 2001.

Termer och förkortningar

ME	Main Engine (Huvudmaskin, HM)
AE	Auxiliary Engine (Hjälpmaskin, HJM)
FO	Fuel Oil (Bränsleolja)
LO	Lubricant Oil (Smörjolja)
PS	Port Side (Babord)
SB	Starboard (Styrbord)
TK	Tank
DWT	Dirty Water Tank
DOT	Dirty Oil Tank
DO	Diesel Oil
SETT	Settlingstank (Sedimenteringstank)
SERV	Servicetank (Dagtank)
BWHT	Flockningstank
Pejla	Läsa av värden, t ex vattenåtgång eller gångtid
Urea	Gödsel som används för avgasrening
Sludge	Smutsig olja
Bunkra	Ta ombord förnödenheter t ex. bränsle.

Bilaga 1: UML-diagram databas

Bilaga 1. UML-diagram databas

Users	
	Login Password Administrator Name Email Bdate

ME	
	Edate valid Login Runtime1 Runtime2 Runtime3 Runtime4 PSgear SBgear LO1 LO2 LO3 LO4 Filled1 Filled2 Filled3 Filled4

AE	
	Edate valid Login Runtime1 Runtime2 Runtime3 Runtime4 Filled1 Filled2 Filled3 Filled4 FOseparator LOseparator

TanksOO	
	Edate valid Login TK24 TK26 TK22 TK23 TK14 TK25 overboard15ppm SludgeToBoat UREAtanks UREAbunker

TanksFO	
	Edate valid Login TK32PS TK33SB TK36SETT TK37SERV MESBfom MESBfom AEfom Boilerfom FObunker BoilerR

Compressors	
	Edate valid Login Start1R Start2R Start3R Work1Total Work1Load Work2Total Work2Load

MElol	
	Edate ME1 ME2 ME3 ME4

Specifikation: Webbapplikation för hantering av driftsdata från fartygsmaskineri

Bilaga 2. Specifikation

Inledning

Den här specifikationen beskriver vilka mål som skall uppnås av applikationen. Här beskrivs vilka data som ska lagras i databasen. Jag tar också upp gränssnittet, fast bara på en mycket grundläggande nivå eftersom det i första hand ska utvecklas under tiden tillsammans med användarna. Jag ger också en beskrivning av applikationens kontext och det sammanhang den ska passa in i. Sist följer en ordlista över förkortningar och termer.

Mål

Applikationen ska utgöra ett stöd för personalen när de ska analysera statistiken. En baktanke med det excelblad som nu används är att skapa rutiner för att varje dygn titta på statistiken, dessa rutiner ska bevaras och underhållas. Applikationen ska vara lätt tillgänglig för alla som behöver informationen i den. Alla ändringar ska bokföras och gamla felaktiga inmatningarna ska sparas.

Databas

I databasen ska följande lagras:

Gångtider	Huvudmaskin 1-4
	Hjälpmaskin 1-4
	Växel SB
	Växel PS
	Panna
	Startluftkompressor 1-3
	Arbetsluftkompressor 1
	Arbetsluftkompressor 2
Smörjoljeförbrukning	Huvudmaskin 1-4
Bränsleoljeförbrukning	Huvudmaskiner SB
	Huvudmaskiner PS
	Hjälpmaskiner
	Panna
Mängd påfylld smörjolja	Huvudmaskin 1-4
	Hjälpmaskin 1-4
Vattenförbrukning	Separator för bränsleolja (FO)
	Separator för smörjolja (LO)
Nivåer	TK 24 FO Leak
	TK 26 FO Sludge
	TK 22 DWT
	TK 23 DOT
	TK 14 BWHT
	TK 25 LO Sludge
	TK 32 BB FO
	TK 33 SB FO
	TK 36 SETT
	TK 37 SERV
	Ureatankar
Bunkrat	Urea
	Bränsle (FO)
Lossat	15 ppm överbord
	15 ppm till sludgebåt

Bilaga 2. Specifikation

Användare	Login
	Lösenord
	Namn
	Administratörsrättighet
	Start datum
	E-postadress

Driftdata delas upp i fem tabeller som motsvarar det som visas i de fem huvudkategorierna i applikationen. Dessa kategorier är ME, AE, Sludge & Urea, FO samt Compressors. Tillsammans med driftsdata ska datum lagras och vem som gjort inmatningen den dagen. Varje ändring av driftsdata ska lagras. Bara vilka användare som finns ska kunna uppdateras utan att ändringar loggas. Användarna ska ha rättigheter på två olika nivåer. Skillnaden mellan de båda nivåerna är att de på den högsta, som är systemadministratörer, ska kunna lägga till och ta bort användare.

Gränssnitt

Hela applikationen skall skrivas på engelska. Alla facktermer och förkortningar som skrivs ut ska vara på engelska. Statistiken samlas in en gång om dygnet, klockan 24, och ska presenteras månadsvis. Data för ej inmatade dagar ska representeras av tomma rader. Utformningen och upplägget av gränssnittet bör efterlikna det excelblad som nu används så långt möjligt, detta för att underlätta en övergång. Varje sida bör få plats att skrivas ut på ett liggande blad. Det skall gå att generera årsrapporter. Inloggning ska krävas för att kunna föra in eller ändra data. Det bör gå att koppla korta noteringar till enskilda celler, se Excels system för kommentarer. Det bör gå att skrolla tabellen med data under rubrikerna till kolumnerna.

Kontext

Applikationen ska kunna användas från klienter med Windows NT Workstation med Explorer 5.50.4807.2300. Klienterna har bildskärmar med 1024*768 och 16 miljoner färger. De datorer som finns i Stena Jutlandicas maskinrum uppfyller dessa krav.

Förkortningar och termer

ME = Main Engine (Huvudmaskin)
AE = Auxiliary Engine (Hjälpmaskin, HJM)
FO = Fuel Oil (Bränsleolja)
LO = Lubricant Oil (Smörjolja)
PS = Port Side (Babord)
SB = Starboard (Styrbord)
TK = Tank
DWT = Dirty Water Tank
DOT = Dirty Oil Tank
DO = Diesel Oil
SETT = Settlingstank (Sedimenteringstank)
SERV= Servicetank (Dagtank)
BWHT = Flockningstank
Ventax (även VT) = Kompressor för arbetsluft
Pejla = läsa av värden och notera.
Urea = Gödsel för avgasrening
Sludge = Smutsig olja
Bunkra = Tanka ombord

Användarhandledning för Stena Jutlandica: Operation Data

Del i examensarbete av Hugo Egerström

Innehållsförteckning

1	Allmän information	27
2	Navigering.....	27
2.1	Huvudmeny	27
2.2	Månadsmeny.....	28
2.3	Tabeller.....	28
2.4	Årsammanställning	28
3	Rapportering och ändringar	29
3.1	Att rapportera driftsdata för ett dygn	29
3.2	Att göra en ändring	29
3.3	Att rapportera månadens smörjoljenivåer för huvudmotorerna.	30
4	Hantera användare	30
4.1	Registrera/Lägga till en ny användare	31
4.2	Ta bort en användare.....	31
4.3	Ändra en användare	31
5	Utskrifter	31
6	Vanliga frågor	32

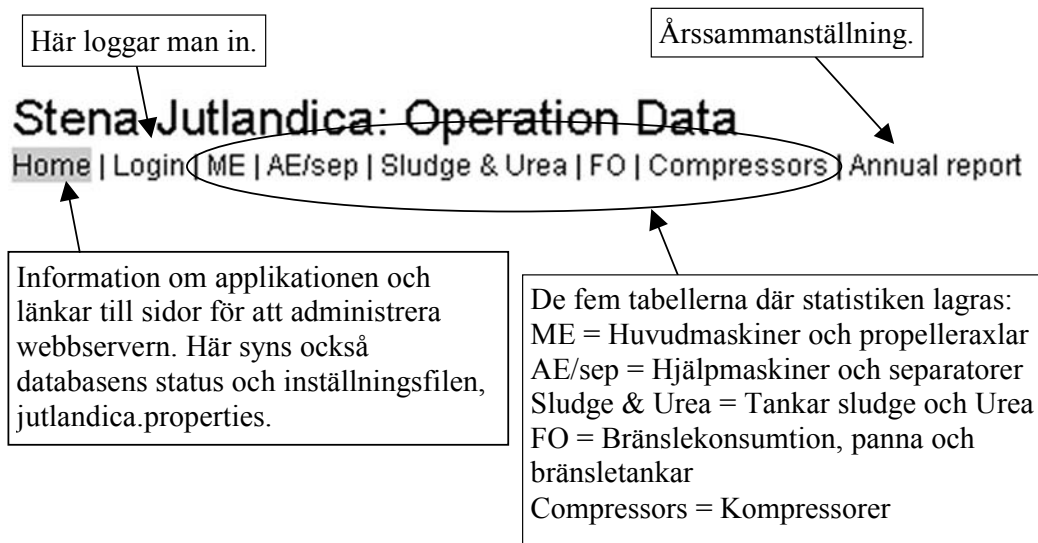
1 Allmän information

”Stena Jutlandica: Operation Data” är en webbapplikation och används således genom en webbläsare. För att kunna använda ”Stena Jutlandica: Operation Data” måste du ha tillgång till en dator med Internetuppkoppling och webbläsaren Internet Explorer version 5.5 eller senare. För att kunna logga in och rapportera måste man vara registrerad av en administratör och av denne ha fått ett användarnamn och ett lösenord.

2 Navigering

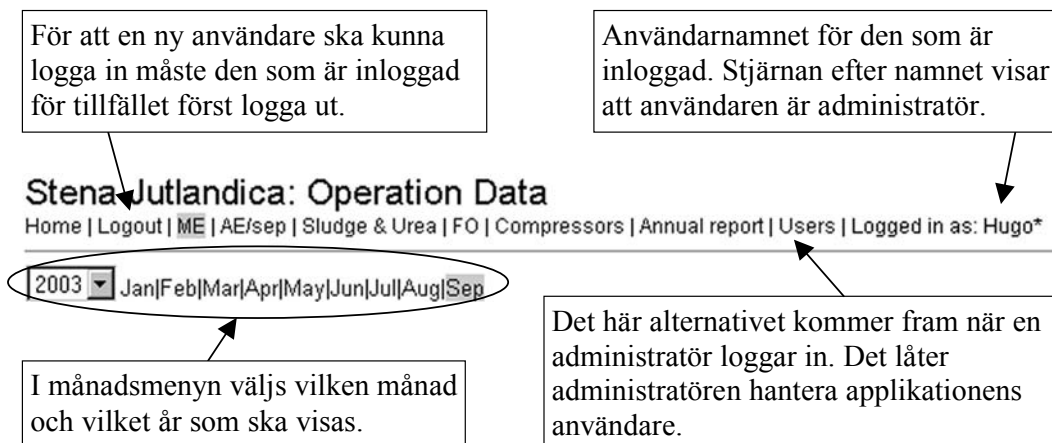
2.1 Huvudmeny

Den valda sidan i menyn är markerad med grå bakgrund. Övriga alternativ i menyn får grå bakgrunder när muspekaren passerar över dem. I figur 1 nedan förklaras de olika alternativen i menyn.



Figur 1 Huvudmeny

När du loggar in förändras huvudmenyns utseende. I figur 2 nedan förklaras vad som hänt när en användare med användarnamnet Hugo loggat in. Användaren Hugo är administratör.



Figur 2 Huvudmenyn och månadsmenyn

2.2 Månadsmeny

När någon av de fem tabellerna är vald visas månadsmenyn som syns i figur 2. Efter inloggning visas alltid tabellen ME.

2.3 Tabeller

I kolumnen längst till vänster i alla tabeller räknas månadsens alla dagar upp. De röda siffrorna markerar söndagar. En stjärna vid siffrorna innebär att de inmatade värdena för det datumet är ändrade, mer om det under rubriken ”Rapportering och ändringar”. I kolumnen längst till höger i alla tabeller hittar man användarnamnet som tillhör den som matat in en viss rads värden. De kolumner i tabellerna med grå bakgrund innehåller värden som matats in av en användare. De andra kolumnerna, med vit bakgrund, innehåller värden som beräknats utifrån värdena i de grå kolumnerna.

Stena Jutlandica: Operation Data

Home | Login | ME | AE/sep | Sludge & Urea | FO | Compressors | Annual report

2003 ▾ Jan|Feb|Mar|Apr|May|Jun|Jul

Date	Runtime kl 2400				Hours/day				
	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4	Total
18	40390	39390	23596	24268	20		13	20	53
19	40410	39390	23609	24288	20		13	20	53
20	40430	39390	23621	24309	20		12	21	53
21*	40450	39390	23631	24329	20		10	20	50
22	40470	39398	23640	24346	20	8	9	17	54
23	40490	39418	23650	24349	20	20	10	3	53
24	40510	39438	23660	24351	20	20	10	2	52
25	40530	39451	23669	24360	20	13	9	9	51
26	40550	39471	23677	24365	20	20	8	5	53
27	40571	39492	23685	24367	21	21			52
28	40590	39511	23694	24371	19	19			51
29									
30	40631	39544	23715	24387	41	33	21		95
31	40651	39564	23727	24388	41	33	21		95
Total runtime ME/month					585				
Total LO file /month									
FO consumption									
Level 1-J									

En stjärna betyder att raden är ändrad.

Söndagar visas med röda siffror.

Dagens datum är markerat med grå bakgrund.

Figur 3 Detalj av tabell ME

2.4 Årsammanställning

Den här sidan är uppdelad i två sidor, Page 1 och Page 2. För att växla mellan de två finns det två länkar upp i det övre vänstra hörnet av tabellen. På den första sidan finns en översikt över huvudmaskiner och hjälpmaskiner med värden från tabellerna ME och AE/sep. Den andra sidan visar en översikt med värden från övriga tabeller. Kolumnen längst till höger innehåller summan av varje rads värden.

3 Rapportering och ändringar

För att kunna rapportera, eller med andra ord mata in, driftsdata i applikationen måste man logga in. För att kunna rapportera en ändring måste man också logga in.

3.1 Att rapportera driftsdata för ett dygn

När man loggat in dyker först tabellen ME upp. På raden ovanför raden med dagens datum finns nu ett antal tomma vita rutor och i kolumnen längst till höger i tabellen finns en knapp märkt "Add", se figur 4. Fyll i värdena i deras respektive rutor och klicka på knappen "Add" eller tryck enter på tangentbordet. Efter en stund kommer sidan fram igen, den här gången med de värden som just matats in ifyllda. Där knappen "Add" tidigare fanns syns nu ett användarnamn. Upprepa samma procedur för att rapportera värdena i de andra tabellerna. När alla värden rapporterats, klicka på "Logout" i huvudmenyn, så att ingen annan av misstag rapporterar felaktiga värden i ditt namn. Varje rad och dygn är öppet för rapportering i ett visst antal timmar efter det att dygnet har passerat, det vill säga efter klockan 12 på natten. Som längst är det öppet i 24 timmar tills nästa dygn öppnas, en kortare tid kan ställas in, se Installations och driftsanvisning för instruktioner. Rapporteras ingen driftsdata för ett dygn under den tid det är öppet lämnas dygnet tomt och kan sedan inte rapporteras i efterhand.

Stena Jutlandica: Operation Data

Home | Logout | ME | AE/sep | Sludge & Urea | FO | Compressors | Annual report | Users | Logged in as: Hugo*

Date	Runtime kl 2400				Hours/day				Gear		Hours/day				LO meter kl 2400				LO consumption				Filled LO				Login
	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4	Total	PS	SB	PS	SB	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4		
1																											
2																										Add	
3																											
4																											
5																											

En vit ruta där värden kan matas in.

Klicka på "Add" för att rapportera driftsdata.

Figur 4 Rapportering av driftsdata

3.2 Att göra en ändring

En ändring kan bara göras av en rad som redan innehåller driftsdata. Tomma rader som inte har rapporterats i tid kan alltså inte rapporteras genom att göra en ändring. För att göra en ändring av en rad så klickar man på datumet i kolumnen längst till vänster. Då visas den raden med värdena förfyllda i vita rutor och en knapp, "Add", längst till höger. Ovanför den raden finns samma rad med fasta värden. För att göra ändringen är det nu bara att ändra det värde som är fel och klicka på "Add". En rad som redan ändrats en gång kan ändras igen, hur många gånger som helst. När man inte är inloggad och klickar som för att göra en ändring så visas alla ändringar som gjorts. De ändrade radernas värden syns då överstrukna ovanför de rätta raderna. Samma sak sker när man är inloggad, enda skillnaden då är att raden man klickat på går att ändra.

Bilaga 3. Användarhandledning

Stena Jutlandica: Operation Data

Home | Logout | ME | AE/sep | Sludge & Urea | FO | Compressors | Annual report | Users | Logged in as: Hugo*

2003		Jan				Feb				Mar				Apr				May				Jun				Jul				Aug				Sep			
Date	Runtime kl 2400				Hours/day				Gear				Hours/day				LO meter kl 2400				LO consumption				Filled LO				Login								
	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4	Total	PS	SB	PS	SB	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4	ME1	ME2	ME3		ME4							
20	40430	39390	23621	24309	20		12	21	53	44002	44001	20	21	71010	82510	44960	50250	60		30	40										mm						
21*	40450	39390	23631	24329	20	-	49	20	50	44022	44021	20	20	71070	82510	44990	50300	60	-	30	50	-	-	-	-	-	-	-	-	-	mm						
22	40470	39398	23640	24346	20	8	9	17	54	44042	44041	20	20	71130	82530	45010	50340	60	20	20	40									500	mm						
23	40490	39418	23650	24349	20	20	10	3	53	44062	44060	20	19	71190	82580	45040	50340	60	50	30											kb						
24	40510	39438	23660	24351	20	20	10	2	52	44081	44080	19	20	71240	82630	45070	50350	50	50	30	10										kb						
25	40530	39451	23669	24360	20	13	9	9	51	44102	44100	21	20	71300	82660	45090	50370	60	30	20	20										kb						
26	40550	39471	23677	24365	20	20	8	5	53	44121	44119	19	19	71360	82710	45120	50380	60	50	30	10										kb						
27	40571	39492	23685	24367	21	21	8	2	52	44142	44141	21	22	71420	82760	45140	50390	60	50	20	10										kb						
28	40590	39511	23694	24371	19	19	9	4	51	44161	44159	19	18	71470	82810	45160	50400	50	50	20	10											kb					
29																																					
30	40631	39544	23715	24387	41	33	21	16	111	44201	44200	40	41	71590	82880	45210	50430	120	70	50	30											mm					
31	40651	39564	23727	24388	20	20	12	1	53	44221	44219	20	19	71650	82930	45240	50430	60	50	30												mm					
Total runtime ME/month					585	455	332	297	1669			619	619					1690	1080	870	640	0	0	0	500												
Total LO filled/month					4780																																
LO consumption/run hour					2.86																																
LO level 1-Jul					ME1	ME2	ME3	ME4	ME1-4			PS	SB						ME1	ME2	ME3	ME4	ME1	ME2	ME3	ME4											
					9.8	9.4	10.8	10.2										2.89	2.37	2.62	3.84																

Här lagras och rapporteras smörjoljenivåerna för huvudmaskinerna.

En överstruken rad som visar de ändrade värdena för raden under.

Den här raden ska just ändras. Raden ovanför visar också de förfyllda värdena.

Figur 5 Att göra en ändring

3.3 Att rapportera månadens smörjoljenivåer för huvudmotorerna.

Längst ner i tabellen ME lagras smörjoljenivåerna för de fyra huvudmaskinerna per den första i varje månad. Se figur 5. Smörjoljenivåerna rapporteras genom att fylla i de vita rutorna och klicka på "Add" till höger om dem. För att ändra värdena gör man på samma sätt, alltså rapporterar värdena igen. Det här är en enklare form av rapportering, inget användarnamn sparas i samband med rapporteringen och ändringar registreras inte. Det finns inte heller någon tidsbegränsning för hur länge rapporteringen av smörjoljenivåer är öppen, värdena går alltid att ändra.

4 Hantera användare

För att hantera användare måste man kunna logga in och dessutom vara administratör så att alternativet "Users" syns i huvudmenyn, se figur 2. Under den första rubriken på sidan Users, "Existing users:" finns en lista. Den visar vilka registrerade användare som finns i applikationen. Under nästa rubrik, "Add new user:", finns ett formulär för att registrera och lägga till en ny användare. Se figur 6. Längst ner finns en länk som heter Transport, mer om den under Övrigt i Installations och -driftsanvisningen.

Det finns två fält för lösenordet för att undvika skrivfel.

Bocka i här för att göra den nya användaren till administratör.

Add new user:

Login	Password	Verify	Admin Name	E-mail	Begin
<input type="text"/>	<input type="text"/>	<input type="text"/>	* <input type="checkbox"/>	<input type="text"/>	2003-09-04
<input type="button" value="Add"/>					

Figur 6 Lägga till ny användare

4.1 Registrera/Lägga till en ny användare

För att lägga till en ny användare fyller man i uppgifterna i de vita fälten under rubriken ”Add new user:” och klickar sedan på ”Add”. Se figur 6. Användaren kommer sedan upp i listan ovanför och kan logga in i applikationen. Dagens datum hamnar i kolumnen ”Begin”.

Existing users:

Login	Name	E-mail	Begin	
Hugo*			2003-04-25	
kb*	Kalle Bohman	kallekarin@hotmail.com	2003-05-05	Delete
goa			2003-05-28	Delete
pt	Patrik Tornblad		2003-05-28	Delete
mm	Mattias M		2003-06-03	Delete
rn	robert		2003-06-11	Delete
LB	Lennart Borg		2003-06-12	Delete
AF*			2003-07-08	Delete

Listan sorteras efter datumen i kolumnen ”Begin”.

Klicka på Delete för att ta bort användaren på respektive rad.

Figur 7 Listan över registrerade användare

4.2 Ta bort en användare

För att ta bort en användare klickar man på ”Delete” i kolumnen längst till höger på den rad användaren står på. Den driftsdata en användare rapporterat påverkas inte på något sätt av att användaren tas bort ur applikationen.

4.3 Ändra en användare

Det går inte att ändra några uppgifter om en användare. För att komma runt det måste man först ta bort användaren och sedan registrera den om igen med de ändrade uppgifterna.

5 Utskrifter

För att skriva ut en tabell används utskriftsfunktionen i webbläsaren. Den första sidan som skrivs ut visar alltid huvudmenyn i applikationen, för att undvika att alltid få med den ska man välja att endast skriva ut sida nummer två. För att en hel tabell ska få plats på en sida bör man välja att skriva ut på en liggande sida. I dialogrutan som kommer upp när man klickar på ”Skriv ut” ska alternativet att skriva ut i formatet ”Landskap” klickas i för att sidan ska skrivas ut liggande.

6 Vanliga frågor

- Fråga: När jag har loggat ut kan man fortfarande använda webbläsarens ”back”-knapp och gå tillbaka och ändra det jag rapporterat. Är inte det fel?
- Svar: Det går inte att ändra några värden efter att man loggat ut, även om det ser ut så. När man försöker kommer man till en helt vit sida och inget ändras i databasen.
- Fråga: Jag har glömt bort mitt lösenord, kan jag be en av mina kollegor som är administratör kolla upp det åt mig?
- Svar: Nej, administratören kan inte se vilket lösenord du har. För att du ska kunna logga in igen måste han ta bort dig som användare och sedan lägga till dig som användare igen.
- Fråga: När jag skriver ut en tabell är bakgrunden vit i alla kolumner trots att den är grå i vissa kolumner på skärmen. Hur kan jag få med den grå bakgrunden i kolumnerna när jag skriver ut?
- Svar: Det här svaret gäller Internet Explorer. Innan du skriver ut, gå till Verktyg i menyn och välj Internet Alternativ. Du får då upp en dialogruta med ett antal flikar längst upp. Klicka på fliken Avancerat längst till höger. I listan du får upp kan du gå ner till rubriken Utskrift och bocka i alternativet Skriv ut bakgrundsfärger och bilder. Klicka på Ok och skriv ut som vanligt. Kom ihåg att den här inställningen nu gäller alla utskrifter från Internet Explorer tills du bockat ur rutan igen.

Installations- och driftanvisning för Stena Jutlandica: Operation Data

Innehållsförteckning

1	Systemkrav.....	35
2	Installation.....	35
2.1	Installation av databashanterare och webbserver:	35
2.2	Installation av applikationen Stena Jutlandica: Operation Data.....	35
3	Konfiguration	35
4	Övrigt.....	36

1 Systemkrav

Den server Stena Jutlandica: Operation Data ska installeras på måste klara av att köra SQL Server 2000. Klienterna behöver webbläsaren Internet Explorer lägst version 5.5 på en skärm med lägst upplösningen 1024x768 bildpunkter för att applikationen ska se ut och fungera som avsetts.

2 Installation

2.1 Installation av databashanterare och webbserver:

1. Börja med att installera SQL Server 2000.
2. Installera sedan Java 2 SDK, SE v1.4.1_02
3. Lägg till en miljövariabeln JAVA_HOME och låt den peka på rootkatalogen till installationen av Java 2SDK ovan.
4. Installera Tomcat 4.1.24 och se till att den hittar Java 2 SDK. Kom ihåg vilket användarnamn och lösenord du sätter till administratören.
5. Från början är Tomcat inställt för att servera webbsidor via port 8080. Ändra det till standard, som är port 80, genom att söka i filen "Tomcat 4.1\conf\server.xml" efter 8080 och sedan ändra det till 80. Spara filen och starta om Tomcat.

2.2 Installation av applikationen Stena Jutlandica: Operation Data

1. Starta Enterprise Manager och importera databasen från filen jutlandica.mdb till en databas med samma namn i SQL Server.
2. Lägg till användare *root* med lösenordet *drikka33cola!*. Ge användaren *root* skriv- och läsrättigheter till de nya tabellerna i databasen.
3. Se till att Tomcat är igång. Gå till adressen <http://localhost/manager/html> med en webbläsare, logga in med det användarnamn och lösenord du uppgav under installationen av Tomcat.
4. Under rubriken "Upload a WAR file to install" finns en knapp, "Browse...", klicka på den och välj filen jutlandica.war. Klicka sedan på "Install".
5. Kontrollera att applikationen fungerar genom att gå till adressen <http://localhost/jutlandica> med en webbläsare. Kopplingen till databasen fungerar om sidan innehåller en grön text som lyder "The database is up and running."

3 Konfiguration

Stena Jutlandica: Operation Data konfigureras genom filen jutlandica.properties som ligger i "Tomcat 4.1\webapps\jutlandica\WEB-INF\classes". Vid installation ser den ut såhär:

```
#Properties for Stena Jutlandica: Operation Data - webb application
#Sun May 11 20:35:39 CEST 2003
DEADLINE_HOURS=24
DB_START_YEAR=2003
DB_NAME=jutlandica
DB_LOGIN=root
DB_DRIVER=com.microsoft.jdbc.sqlserver.SQLServerDriver
DB_PASSWORD=drikka33cola!
DB_URL=jdbc\:microsoft\:sqlserver\://localhost\:1433;databaseName=jutlandica;SelectMethod=cursor
```

Bilaga 4. Installations- och driftanvisning


De första två raderna kan man bortse ifrån, de talar bara om vad filen är till för och när den är skapad. Efterföljande rader börjar med ett attribut och sedan attributets värde, på engelska ”property”. Attributen är de variabler som styr hur applikationen ska uppföra sig. Här följer en tabell över attributen och vad de är till för.

Attribut	Beskrivning
DEADLINE_HOURS	Talar om hur många timmar efter midnatt det ska vara möjligt att föra in föregående dygns värden. Det från början inställda värdet, 24, betyder alltså att det är möjligt ända tills nästa dygn tar vid.
DB_STAR_YEAR	Den här variabeln talar om för applikationen vilket som är det första året i databasen. Låt det här vara innevarande år om applikationen just börjat användas.
DB_NAME	Namnet på databasen i SQL Server. Behövs för att ansluta till databasen.
DB_LOGIN	Användarnamnet som används för att ansluta till databasen.
DB_DRIVER	Namnet på den JDBC-drivrutin som används för att ansluta till databasen.
DB_PASSWORD	Det lösenord som används för att ansluta till databasen tillsammans med användarnamnet i DB_LOGIN.
DB_URL	Den sträng som skickas till drivrutinen för att ansluta till databasen. När man kör databasen på en annan server än webbservern måste man ändra där det står localhost. Mer information om detta finns i dokumentationen om JDBC.

Vid installation finns en användare registrerad i applikationen som standard, användarnamnet är *hugo* och lösenordet är *hej*.

4 Övrigt

När man börjar använda applikationen kommer de värden som är beroende av transport från tidigare månad att ge för höga summor. Det beror på att de då utgår ifrån startvärden på noll. För att ändra dessa startvärden kan man använda ett inbyggt verktyg som finns under menyvalet Users i applikationen. Logga in som administratör och klicka på länken längst ner på sidan Users. Det kommer då upp fem olika formulär, ett för varje tabell. Fyll här i lämpliga värden så kommer applikationen att summera rimligare gångtider, förbrukningar och så vidare. Om man fyller i fel i något formulär kan det enkelt korrigeras genom att fylla i igen och klicka på Add.

 LINKÖPINGS UNIVERSITET	Avdelning, Institution Division, Department Institutionen för Datavetenskap 581 83 LINKÖPING	Datum Date 2003-09-17
--	--	------------------------------------

Språk Language X Svenska/Swedish Engelska/English	Rapporttyp Report category Licentiatavhandling X Examensarbete C-uppsats D-uppsats Övrig rapport —	ISBN ISRN LITH-IDA-EX-ING--03/014--SE Serietitel och serienummer ISSN Title of series, numbering _____
URL för elektronisk version http://www.ep.liu.se/exjobb/ida/2003/dd-c/014/		

Titel Title	Webbapplikation för hantering av driftsdata från fartygsmaskineri. Webb application for handling of operation data from ship machinery.
Författare Author	Hugo Egerström

Sammanfattning Abstract	<p>“Stena Jutlandica: Operation Data” is the name of a webb application with the purpose of managing runtime data from the machinery of the ship Stena Jutlandica. The application will be replacing an excel spreadsheet which has been in use for several years. One of the reasons to collect runtime data is that faults can be discovered more quickly. Examples of runtime data generated from the machinery is runhours for different engines, levels in different tanks and water consumption. Runtime data is stored in a databas on a server in Stena Lines IT-department. On board Stena Jutlandica the webb application is accessed thru a webb browser via a sattelite which keeps the ship on line. This report describes how the development has proceeded and how the resulting application works.</p>
-----------------------------------	--

Nyckelord Keyword	Webbapplikation, driftsdata, fartyg, jsp, java
-----------------------------	--