Prosessdokumentasjon

Dette er prosessdokumentasjon

Hva som er gjort og valg:

* Desktop:

Et krav fra oppgaven var å bare levere en JFrame, så vi brukte flere internalframes.   
Den ene JFramen blir nå brukt som et desktop men en egen menylinje helt oppe på siden. I denne menyen er det mulig å sette fokus/åpne, adde kontrollvinduer, avslutte programmet og lese noe informasjon om programmet.

* Inndeling av vinduer:

Vi valgte å dele inn programmet i 3 hoved deler(internalframes). En for kontrollvindu, som skal kontrollere heiskort og stå utplassert i bakkene. En for informasjonsvindu, som skal være ”nettsiden” og selgervindu, som både skal inneholde muligheten til å selge heiskort, og den statistikken som er nødvendig. Vi valgte å legge disse to siste sammen, for vi ville gjøre det så realistisk som mulig.

* Selgervindu GUI:

Når vi lagde selgervindu prøvde vi å passe på at vinduet ikke innholdet for mye så det ble uoversiktlig, og var enkelt nok satt sammen til at en ”ansatt” eller en person med litt forkunnskaper skulle kunne bruke programmet raskt og lett. Siden skisenteret kun tilbyr 3 forskjellige produkter(korttyper) valgte vi å vise alle rett i vindu. Hadde det vært behov for flere vinduer ville det vært mulig og bytte ut panelet med registrering med et panel som inneholder knapper for flere produkter.

* Logg inn:

Siden selgevindu er ment for ansatte og eiere hos skisenteret mente vi at sikkerheten ved programmet kan være viktig. Vi lagde derfor 2 brukere, men ferdig passord og brukernavn. Passordet er lagd i en array som blir sammenligner inputen. Hvis admin logger inn settes en boolean til true, og et bilde vises på skjermen.

Når du logger ut vil også programmet skrive til fil.

* Start/steng skift:

Vi valgte å legge inn en åpna/lukke skift som åpner eller stenger skisenteret og styrer om alle transaksjoner kan gå gjennom.

* Reg keycard/lag ny:
* Reg korttyper:

Å registrere en korttype krever at et keycard finnes i lista på skjermen. Hvis den er tom vil du få feilmelding. Registrering av korttypen har en default-innstilling, med 1 enkel voksenbillett. Dette kan endres på i vindu. Når du oppretter en korttype blir et keycard og korttypen send med til en handleliste som holder styr på hva kunde kjøper men lagrer ikke opplysningene i registeret siden kjøpet kan avbrytes. Handleliste vil bli mer forklart lengre ned. Totalsum, handleliste og keycardlista blir også oppdatert. Ved registrering av dagskort er det også en test som finner ut når kortet blir registrert. Hvis klokken er etter stengetid vil kortet bli aktivert neste dag, hvis kortet blir kjøpt midt på dagen får du en rabatt. Rabatten kan stilles på.

* Rabatter:

Vi har to hoved-rabatt grupper, skole og høytid. Hvis begge rabattene er valgt, vil den høyeste rabatten velges. I systemets pris-endring-metode finne det også muligheter for skisenteret til å gi tilbud på forskjellige kort.

* Handleliste:
* Handleliste framvisning:

Siden det var behov for flere typer informasjon i den synlige handlelista bestemte vi oss for å bruke en JTable. Problemet vi fort møtte var at en vanlig JTable ikke støtter å legge til nye rader etter den er opprettet. Vi brukte derfor DefaultTableModel() som støtter denne funksjonen.

* Legge i registrene:
* Kundebehandling:
* Endre priser:

Vi syns det ville være lurt å gi admin ved skisenteret muligheten til å endre priser og antall klipp selv, og gjøre det lett. Vi la også med en gjenopprett-metode. Når prisene endres lagres alle prisene i Pris.java, sånn at når programmet starter henter den inn sine verdier for prisene der. Når du endrer prisene blir de nye verdiene sendt med til infovindu så prislistene der holder seg oppdatert.

* Vise/Lagre statistikk

Ved betaling blir først all ny data lagt til i registrene, også adminstat som er lagd for alle salgstall. Dette registeret blir lagret og lest fra ved oppstart.

* Dagsoppgjør:

Når en ansatt lukker skiftet og logger av ved stengetid lagres informasjonen om hva som er solgt osv ved den dagen. Hvis ved en feil en ansatt skulle logge av og på flere ganger i løpet av en dag vil kun den første lagringen skje, mens alt som blir registrert etter bli tatt med til neste dags oppgjør. For at dagene skal lagres, ligger alle dagene i et linkedlist i adminstat. Ved programstart blir det som ligger i denne listen lagt til i vindu.

* Diagrammer:

Det er lagd en klasse som oppretter et panel med et diagram. Klassen får inn 3 verdier og setter størrelse på søylene etter disse. Denne klassen blir brukt flere ganger med forskjellige verdier.

* Script:
* Kontrollvindu
* Infovindu:

Infovindu er som sagt flere ganger lagd for en nettside. Den har ingen metoder men inneholder flere paneler men informasjon, bilder og en JTable med priser.

Ansvar:

Aktivekort:

Martin lagde første utkast av klassen, med listen, enkle liste metoder, og oppdater og sjekk metodene.

Sondre tok så over og forbedret deler av koden, lagde "avanserte" stringer, utvidet sjekker tok også ansvar for at klassen utvidet seg med programmet

Kontrollvindu:

Lagd av Martin orginalt med da sjekk metoden og panelet.

Sondre stod for timer og begge to fikset opp i flere "feil" som kom etter at prorammet ble utbedret.

Kunde:

Martin og Sondre (valgt ut variabler osv)

Kundebehandling:

KundeBehandlingPanel metodene ble skrevet av Sondre og store deler av panelet, av Martin.

Pris:

Lagd av Martin men satt i egen klasse og overført av Sondre

PrisPanel:

Panel og metoder er lagd av Martin og sondre har flyttet og gjort alle endringer som følger med det

SelgerMetoder:

Metodene er lagd av både Martin og Sondre. Martin har lagd registrering av korttypene, sletting, logg inn,dagsoppgjør og små metoder.

Sondre har lagd metodene for å lage keycard og fjerne dem, betal.

Begge to har også gjort endringer og forbedringer rundt om i alle metodene når det har vært nødvendig

SelgerVindu:

Martin har lagd panelene.

Sondre har knyttet klassene sammen og noen metodre.

Begge to har også gjort endringer og forbedringer rundt om i metodene.

SkiSenter:

Denne klassen ble skrevet både Martin og Sondre, men hovedsakelig Martin, som oppretta DesktopPane, InternalFrame og MenuBar.