Datateknik C, Examensarbete, 15 högskolepoäng

GRÄNSSNITT FÖR AOD-SIMULATOR

Johan Eriksson

Simulering & dataspelsteknik, 180 högskolepoäng

Örebro vårterminen 2016

Examinator: Martin Magnusson

INTERFACE FOR AN AOD SIMULATOR

Örebro universitet
Institutionen för naturvetenskap och teknik
701 82 Örebro

Örebro University
School of Science and Technology
SE-701 82 Örebro, Sweden

Sammanfattning

Att kunna simulera en AOD-process (Argon, Oxygen and Decarburization) är i dagsläget möjligt med hjälp av en programvara. En sådan programvara används till utbildning och forskning för utveckling av rostfritt stål.

Kobolde & Partners AB är ett företag som äger en sådan programvara. I detta projekt byggs programmet ut för att ge användare möjlighet att simulera en AOD-process med osäkerhet i mätdata. Genom detta kan forskare få ökad förståelse hur processens mätresultat skiljer sig med osäkerhet i mätdata.

Abstract

Being able to simulate an AOD process (Argon, Oxygen and Decarburization) is possible nowadays with help of software. This kind of software is used for education and research for development of stainless steel.

Kobolde & Partners AB is a company that owns this kind of software. This project will further develop this software to enable users to simulate an AOD process with uncertainty in measurement. This allows scientist to get more knowledge how a process measurement results differ with uncertainty measurements.

Förord

Jag vill tacka tekn lic Rutger Gyllenram vid Kobolde & Partners för möjligheten att få utföra detta projekt samt medarbetarna tekn dr Changji Xuan, tekn dr Joel Gustavsson samt bergsingenjör Patrik Ternstedt som alla ställt upp och hjälpt till vid behov. Jag vill även tacka min handledare civ ing Jack Pencz vid Örebro universitet för hjälp med strukturering av projektet och med rapporten.

Innehållsförteckning

1 INLEDNING 5
1.1 TIMEAOD 2 5
1.2 PROJEKTET 5
1.2.1 Den ursprungliga projektplanen 5
1.2.2 Den nya projektplanen 6
1.3 SYFTE 7
1.4 KRAV 7
1.5 BAKGRUND 7
1.5.1 Visualisering av AOD-process på ett ergonomiskt gränssnitt 7
1.5.2 Osäkerhet i mätdata 8
2 TILLVERKNINGSPROCESS OCH BEHOV AV SIMULERING 9
2.1 AOD-PROCESSEN 9
2.1.1 Avkolning 9
2.1.2 Reduktion 10
2.1.3 Avsvavling 10
2.1.4 Charge 10
2.2 SIMULERINGSBEHOV 10
3 DESIGN 12
3.1 HUR SKA AOD-SIMULATORN FUNGERA? 12
3.2 AOD-SIMULATOR 12
3.2.1 Gränssnitt 12
3.2.2 Vad som krävs i gränssnittet 12
3.3 DESIGN AV ETT GRÄNSSNITT 13
3.3.1 TimeAOD 2 struktur och design 13
3.3.2 Alternativ för design 15
4 METODER OCH VERKTYG 20
4.1 METODER 20
4.2 VERKTYG 20
4.3 ÖVRIGA RESURSER 20
4.3.1 Apache Struts 20
4.3.2 JFreeChart 20
4.3.3 Random gaussian 20
5 GENOMFÖRANDE 21
5.1 FAS 1: INFORMATIONSSÖKNING 21
5.1.1 AOD-process 21
5.1.2 Design av gränssnitt 21
5.1.3 Osäkerhet i mätdata 21
5.2 FAS 2: UPPDATERING AV STUDIE AV BEFINTLIGT PROGRAM 21
5.2.1 Uppdateringar 21
5.2.2 Struktur 21
5.3 FAS 3: IMPLEMENTERING AV GRÄNSSNITT 21
5.3.1 Befintliga gränssnittet 22
5.3.2 Databas 22
5.3.3 Formulär 22
5.3.4 Trädmeny 23
5.3.5 Simulering 23
5.3.6 Testning 23
6 RESULTAT 24
6.1 RESULTATET AV DE HÅRDA KRAVEN 24

7 DISKUSSION 29
7.1 UPPFYLLANDE AV PROJEKTETS KRAV 29
7.1.1 Hårda krav 29
7.2 PROJEKTETS UTVECKLINGSPOTENTIAL 29
7.2.1 AOD-Simulatorn 29
7.3 REFLEKTION KRING EGET LÄRANDE 30
8 REFERENSER 31

1 Inledning

Kobolde & Partners AB är ett företag med ett tiotal anställda som arbetar med att ta fram och underhålla applikationer till större företag, huvudsakligen inom stålindustrin. De anställda är utbildade inom metallurgi, kemi, fysik och datateknik. [1]

1.1 TimeAOD 2

På 90-talet utvecklades en programvara kallad TimeAOD 2, programmet bygger på en AOD-reaktionsmodell som ursprungligen utvecklades av Pontus Sjöberg [2]. TimeAOD 2 används för simulering av ståltillverkning med en så kallad AOD-process, där AOD är akronymen för ”Argon, Oxygen and Decarburization”. Med hjälp av TimeAOD 2 kan man följa en process under utvecklingen och läsa av dess processdata.

År 2003 vidareutvecklade Kobolde & Partners programvaran till ett webbaserat gränssnitt, på så sätt skall eventuella kunder få lättare tillgång till programmet samt så blir underhållet av programmet enklare. Programmet består idag av en webbserver som i sin tur är ihopkopplad med en databas, med hjälp av HTML, CSS, Javascript visas serverkoden på gränssnittet. Runt år 2006 tog de stop i utvecklingen av programmet, sedan dess har det legat orört och oanvänt.

1.2 Projektet

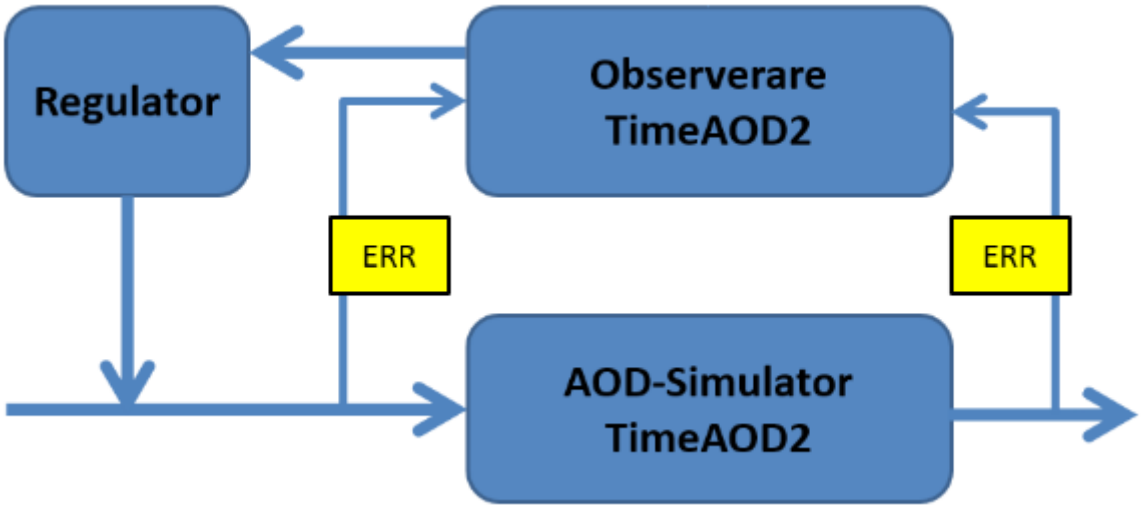
Den ursprungliga projektplanen sattes upp utan någon som helst granskningen av TimeAOD 2. När projektarbetet började, visade det sig att viktiga delar i programvaran inte var färdigutvecklade och kunde därför inte användas. Detta medföljde ändringar i projektplanen. I avsnitt 1.2.1 redovisas den ursprungliga projektplanen och den nya beskrivs i 1.2.2.

1.2.1 Den ursprungliga projektplanen

Projektet gick ut på att koppla ihop de befintliga delarna till en experimentplattform som visas i figur 1. Målet hopkopplingen var följande:

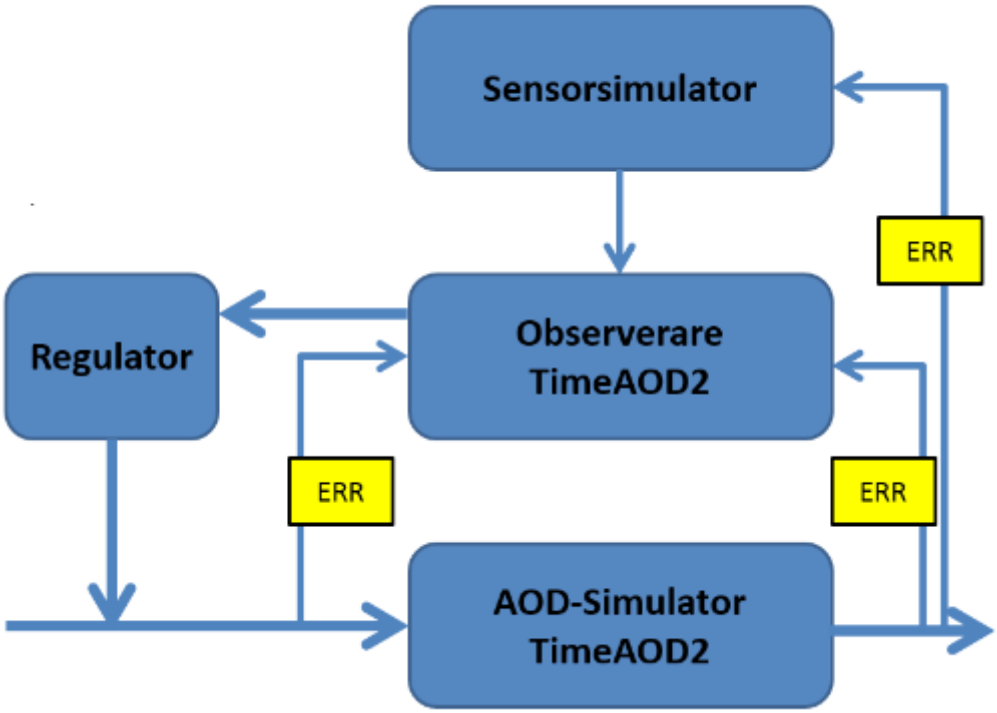
- Simulatorn ska agera process.
- Observeraren ska läsa av processdata (från simulatorn) och göra en uppskattning av viktiga processegenskaper som smältanalys, slagganalys och temperatur.
- Regulatorn ska ta hand om styrningen av själv processen, dvs vilka mängder av syre och argon som ska tillsättas för att den simulerade AOD-processen ska uppnå sitt mål.

Eftersom mätningar inte är exakta, kan en observerare i verkligheten ge felaktiga resultat. Därför var avsikten att simulera felaktiga mätresultat (processdata) i systemet genom att generera fel i de data som ”mäts” i simulatorn.



Figur 1. Systemets delar vid projektets start och hopkoppling av dessa.

Utöver simulering av felaktiga processdata skulle även en sensorsimulator kopplas in i reglersystemet. Se figur 2. En sensorsimulator behövs för att observeraren ska ge noggrannare data till regulatorn. I figur 2 visas hur det är tänkt att det skulle se ut.



Figur 2. En sensorsimulator kopplad till reglersystemet.

1.2.2 Den nya projektplanen

Under projektets början insågs det att den tilltänkta observeraren i TimeAOD 2 inte var färdigutvecklad, endast en test version fanns tillgänglig. Dess dåvarande funktionalitet var inte tillräcklig för att hålla sig till den ursprungliga projektplanen.

Genom att skapa en simulering i TimeAOD 2 och sedan ha möjligheten att ange osäkerheter i valda parametrar, skulle man kunna uppnå samma resultat som var tänkt i den ursprungliga projektplanen.

1.3 Syfte

Syftet var att bygga vidare på TimeAOD 2s webbgränsnitt, där användare ges möjlighet att simulera processer med osäkerhet i mätdata. Resultatet av simuleringen ska presenteras för att möjliggöra studier av beroendet av kvalitén på indata till AOD-processen.

1.4 Krav

Ändringar i projektplanen medförde även ändringar i kraven. Dessa krav anpassades efter nya projektplanen:

- Undersöka hur ett bra gränssnitt kan skapas för reglering av en AOD-process.
- Vidareutveckla simulatorn, där osäkerheter i mätdata ska kunna anges i utvalda parametrar. Därefter körs angivet antal beräkningar med angivna osäkerheter. Detta skall kunna köras från ett ergonomiskt välutvecklat gränssnitt som visar processdata.

1.5 Bakgrund

1.5.1 Visualisering av AOD-process på ett ergonomiskt gränssnitt

För att skapa ett så bra ergonomiskt och användarvänligt gränssnitt som möjligt kan man enkelt följa grundläggande riktlinjer som redan tagits fram i [3]. Sammanfattningsvis handlar det om följande:

1. Låta användare ha kontroll.
2. Minska användarens minnesbelastning.
3. Gör gränssnittet konsekvent.

Genom att ha dessa riktlinjer i åtanke när man skapar ett gränssnitt bör man kunna få fram ett användarvänligt gränssnitt. Ett bra gränssnitt kan vara avgörande hur användare använder ett program. Ett dåligt gränssnitt kan medföra frustration hos användare vilket kan leda till att användning av programmet uteblir. Medan ett bra gränssnitt enkelt kan öka effektiviteten hos användare vilken medför nöjdare användare. [4]

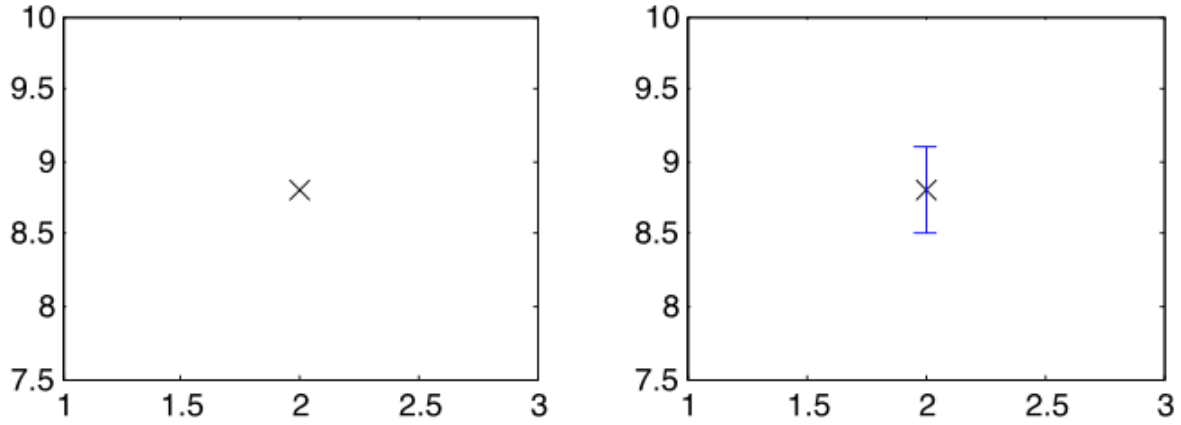
Genom visualisering av processer kan man tyda resultat som inte kan upptäckas med egna ögon vid t ex en observation av en AOD-process. Genom att bearbeta data från en process och presentera detta på ett bra sätt kan användare få ökad förståelse en process. Visualisering i graf är ett tillvägagångssätt att presentera insamlad data på – genom en graf kan användare enkelt se och tyda processdata. [5]

För att presentera en visualisering av en AOD-process med hjälp av en graf på bästa sätt är färgvalet viktigt för varje parameter. I [2] finns tydliga exempel på hur delar av en AOD-process kan presenteras via grafer. Valet av färger för ett gränssnitt har en stor betydelse då olika användare ser färger på olika sätt. Genom att använda färger som tydligt kan urskiljas mot en bakgrundsfärg underlättar betydligt för färgblinda personer. Ett gränssnitt bör vara användbart för så många användare som möjligt. Därför bör val av färger tas med som en viktig punkt vid design av ett gränssnitt. [6]

1.5.2 Osäkerhet i mätdata

Osäkerhet i mätdata är något man måste ta hänsyn till vid hantering av mätresultat. Sällan stämmer ett mätt resultat till 100 %. Instrument som används till mätning av objekt har oftast en angiven osäkerhet. För att ett instrument ska nå så precisa resultat som möjlighet bör regelbundna kontroller ske samt kalibreringar. Det är inte bara instrument som medför osäkerheter, exempelvis kan omgivningen och vem som utför mätningen spela roll.

Att ta med osäkerheter i sina beräkningar är mycket viktigt, eftersom ett slutresultat utan beräknade osäkerheter kan skilja från det egentliga resultatet. Ett resultat med osäkerheter kan på flera sätt presenteras för användare. I figur 3 visas exempel på hur osäkerheten kan redovisas med hjälp av en graf.



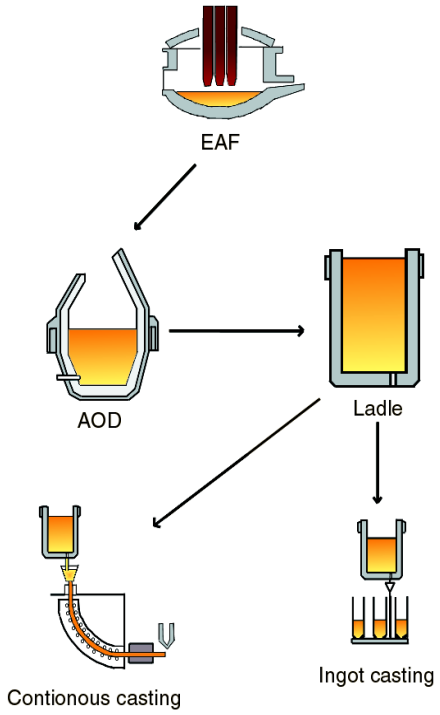
Figur 3. Exempel på visualisering av osäkerheter med hjälp av en graf.

I figur 3 kan man avläsa ett mätresultat markerat med ett kryss. Den blå markeringen visar maximum och minimum med angiven osäkerhet. Med hjälp av en sådan graf kan man se inom vilket intervall resultatet befinner sig inom. På så sätt kan ett mer korrekt resultat utläsas. Att kunna presentera osäkerheter kommer till stor nytta i detta projekt. Förhoppningsvis kommer detta leda till ökad förståelse för en AOD-process vilket i sin tur kan leda till optimeringar av processen. [7, 8, 9]

2 Tillverkningsprocess och behov av simulering

2.1 AOD-processen

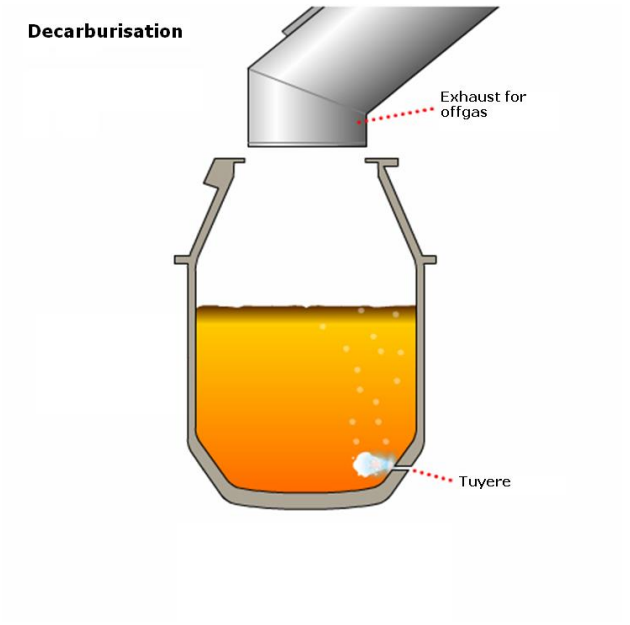
Tillverkning av rostfritt stål är en processkedja som är indelad i flera steg, enligt figur 4. Inledningsvis smälts skrot i en elektrisk ljusbågsugn (eng. Electric Arc Furnace – EAF). Därefter överförs smältan som består av järn, kol och legeringsämnen till en AOD-konverter. I detta skede har smältan en temperatur upp mot 1600 °C och vikten ligger vanligtvis runt 50 till 150 ton. Efter utförd AOD-process transporteras det färdiga stålet vidare för stränggjutning eller gjutning i formar.



Figur 4. AOD-processen för tillverkning av stål [10].

2.1.1 Avkolning

Syftet med AOD-processen är få bort kolet i stålet. Detta görs genom att tillsätta syre som i sin tur reagerar med kolet. Under reaktionen bildas koloxid som sugs ut via ett avgassystem. Denna princip visas figur 5. Det är inte bara kol som reagerar med syre, även krom reagerar med syret men denna reaktion bildar ingen gas utan all krom som reagerat med syret hamnar i slaggen. Eftersom krom är ett ämne som man vill ha kvar i stålet, blandas syret med ökande mängder argon eller kväve i olika proportioner. Genom att göra detta behålls en stor del av kromet i stålet.



Figur 5. AOD-konverter [10].

2.1.2 Reduktion

När önskad kolhalt och temperatur har nåtts från föregående steg, avkolningen, påbörjas reduktionen. Målet med detta steg är att återvinna krom som oxiderats till slaggen under förgående steg. Genom att tillsätta legeringsämnen såsom kisel och aluminium som har högre affinitet för syre än krom, så reagerar syret i kromoxiden med dessa ämnen och kromet återvänder ner i stålet. Kalk och kalciumfluorid (CaF_2) är två andra ämnen som tillsätts i detta steg. Dessa ämnen hjälper till i återvinningen av krom och hanterar slaggen på så sätt att de håller slaggen flytande och volymen relativt liten.

2.1.3 Avsvavling

Efter reduktion görs avsvavling. Målet med detta steg är att nå rätt svavelhalt i stålet. För att minska svavelhalten i stålet tillsätts kalciumoxid (CaO) som binder svavel (S). Då sker följande kemiska reaktion:

$$S_{(\text{stål})} + CaO_{(\text{slagg})} \rightarrow CaS_{(\text{slagg})} + O_{(\text{stål})}$$

där resultatet blir kalcumsulfid (CaS) och syre (O).

För att slutligen minska syrehalten tillsätts kisel eller aluminium. När denna process är klar, avlägsnas slaggen från konvertern och stålet transporteras vidare för att gjutas. Se figur 3.

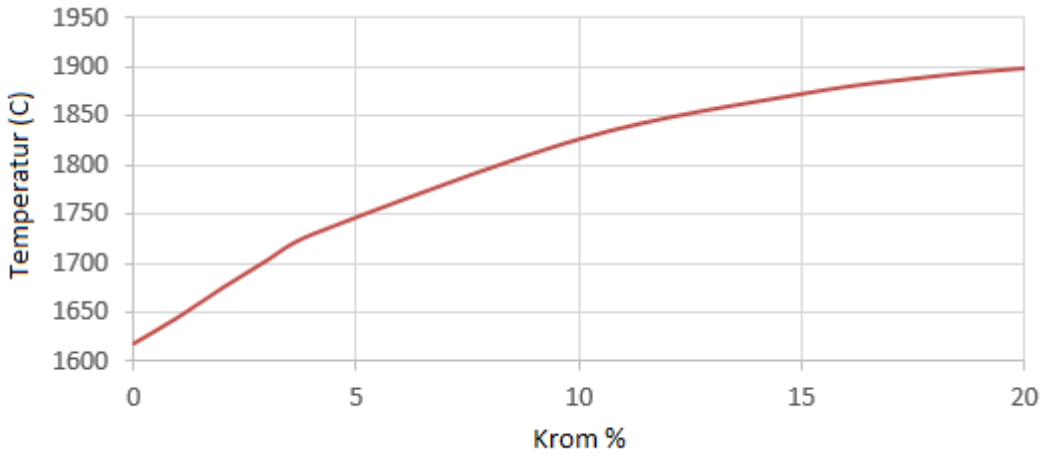
2.1.4 Charge

En flytande blandning av stål i tillverkningsprocessen kallas för en charge. Detta namn används av företag inom stålindustrin.

2.2 Simuleringsbehov

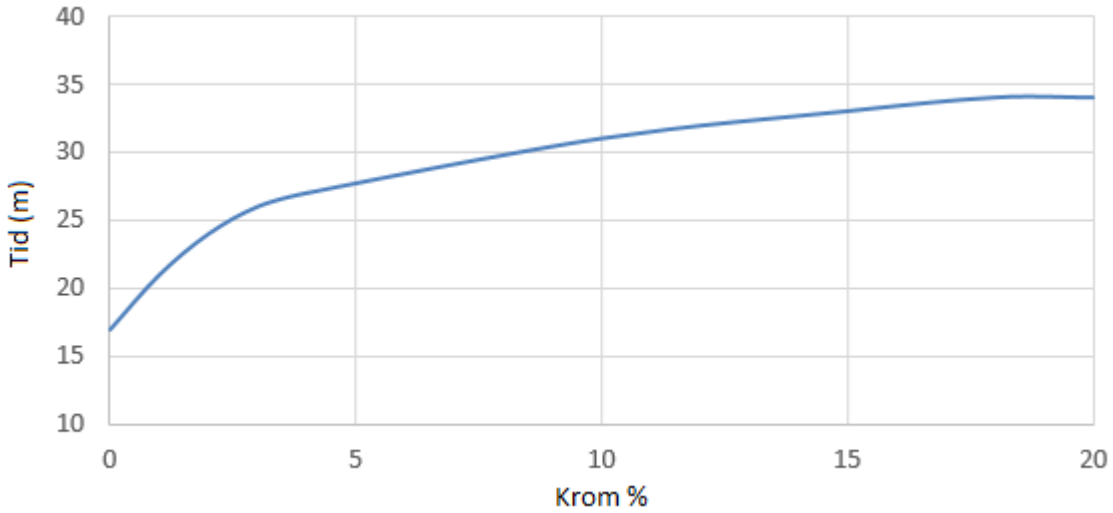
Osäkerhet i mätdata påverkar slutresultatet i ADO-processen. I en stålblandning är det i princip omöjligt att räkna ut hur många procent av ett ämne som ingår. Därför uppskattas ett ungefärligt värde för alla ämnen och därefter görs beräkningar hur mycket tillsatser som

behövs i AOD-processen. Eftersom mätdata inte är tillräckligt noggranna, påverkas resultatet i AOD-processen vilket ofta medför nödvändiga justeringar. Två experiment utfördes i TimeAOD 2 för att ta redan på hur felberäkningar påverkar en AOD-process. I figur 6 och figur 7 visas resultaten.



Figur 6. Temperaturen i AOD-processen som funktion av kromhalten.

Grafen i figur 6 visar hur kromhalten påverkar blandningens temperatur under en AOD-process. Om kromhalten ligger på cirka 0 % i en blandning kan man se att temperaturen är betydligt lägre än en blandning som innehåller cirka 20 % krom. Temperaturen är viktig att ta hänsyn till under en process. Kärlet som används för en AOD-process tål inte för hög värme. Detta kan dock delvis lösas genom att tillsätta kylstål under processen.



Figur 7. Behandlingstiden för AOD-processen som funktion av kromhalten.

Grafen i figur 7 visar hur behandlingstiden av en AOD-process påverkas av kolhalten i en blandning. Desto mindre procentandel krom som ingår i en blandning desto snabbare går processen. Detta kan enkelt ses med hjälp av grafen i figur 7.

3 Design

3.1 Hur ska AOD-simulatorn fungera?

Användare ska välja en charge som simulering ska baseras på. Simulering fungerar så att ett antal valda uträkningar görs på den valda chargen. Chargens parametrar ändras efter varje uträkning. De parametrar som ändras är de som man har möjlighet att ange osäkerheter för. Resultatet skall sedan presenteras där man tydligt kan se hur uträkningar med osäkerheter skiljer sig från en uträkning utan osäkerheter.

3.2 AOD-Simulator

3.2.1 Gränssnitt

Eftersom AOD-simulatorn är tänkt som en utbyggnad av TimeAOD 2, så kommer arbetet ta plats i programmets befintliga gränssnitt. En ny flik i gränssnittet kommer därför skapas med plats för simulatorn.

3.2.1.1 Förutsättningar för simulering

För att kunna köra en simulering krävs ett antal beräkningar av en charge. För att utföra en beräkning behövs en charge, ett parameterset, definierade avsvavlingssteg samt val av slutprodukt. Eftersom en beräkning redan kan genomföras i TimeAOD 2, krävs inga ny tillägg för att få en beräkning att fungera.

3.2.1.2 Krav på gränssnittet

De krav som fanns på gränssnittet var att kunna ange allt som behövs för att starta en beräkning, att kunna namnge en simulering och möjligheten att kunna spara en simulering. Dessutom ställde företagshandledaren kravet att osäkerhet i följande parametrar skulle anges:

- kol
- kisel
- temperatur
- stålets vikt
- slaggens vikt

3.2.2 Vad som krävs i gränssnittet

Gränssnittet byggs upp av ett formulär. Detta formulär består av inmatningsfält och knappar. Ifyllt formulär skickas till en server när en handling har valts, exempelvis handlingen att spara.

3.2.2.1 Formulär

Användaren behöver specificera ett namn samt en beskrivning av simulationen. Dessa visas i formuläret tillsammans med inmatningsfält. Eftersom användaren redan har definierat charger, parametersets, avsvavlingssteg samt slutprodukter, kommer formuläret att läsa in de lagrade definitionernas namn från databasen och presentera dem i en meny av typen drop-down. Dessa definitioner består av en mängd parametrar som inte är relevant för användaren i detta skede. Därför är det smidigast att presentera dessa med hjälp av namn.

Osäkerheten är en viktig del av simulatorn, här ska antalet uträkningar som skall göras anges samt de osäkerheterna i valda parametrar anges. För att underlätta när osäkerheten ska anges i

chargens parametrar, ska chargens parametrar presenteras i fält som inte går att förändra. Osäkerheten ska anges i ett separat inmatningsfält för varje parameter.

3.2.2.2 Spara, uppdatera och ta bort simuleringar

Det ifyllda formuläret ska sparas som en ny simulering. Detta för att man ska kunna utföra flera simuleringar på olika charger. En simulering kan komma att ändras eller justeras därför behövs en uppdateringsfunktion. En funktion för borttagning av sparade simuleringar kan komma till användning när användare inte längre vill behålla de sparade filerna.

För att funktionerna spara, uppdatera och ta bort ska fungera krävs en ny tabell i databasen och en koppling mellan den nya tabellen och servern. Servern behöver givetvis klara av att utföra dessa funktioner.

3.2.2.3 Meny för sparade simulationer

För att användaren ska ha möjlighet att se sina sparade simuleringar ska dessa presenteras i en trädmeny. I TimeAOD 2 presenteras sparade filer just genom trädmenyer. För att gränssnittet ska vara så sammanhängde som möjligt kommer denna del att efterliknade det som redan finns. Varje sparad simulering ska fungera som en länk till dess information och alla dess funktioner.

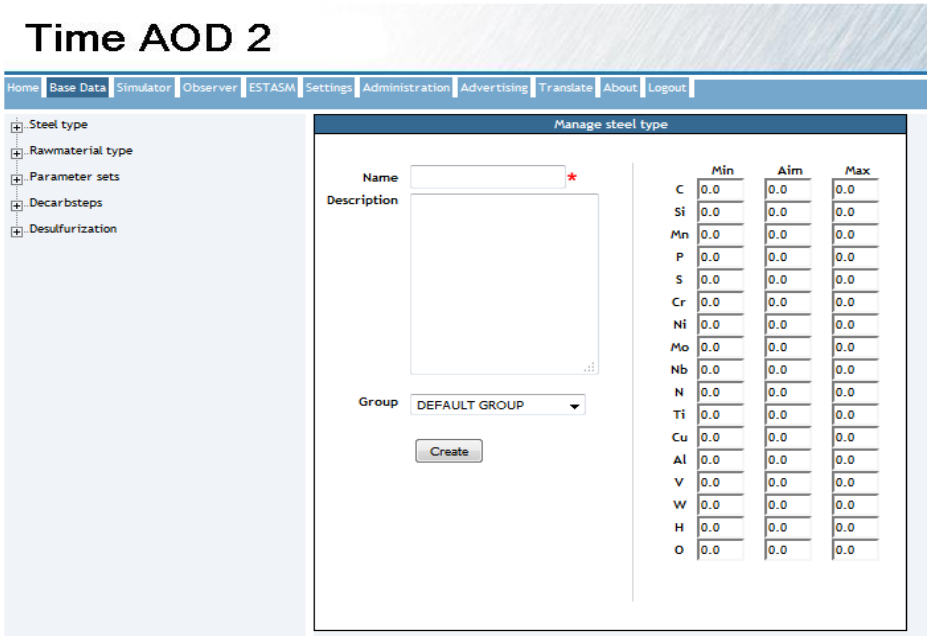
3.3 Design av ett gränssnitt

3.3.1 TimeAOD 2 struktur och design

Förståelse för TimeAOD 2s struktur och funktionalitet är viktigt för designen av simuleringsgränssnittet.

3.3.1.1 Steg 1, Definiering av basobjekt

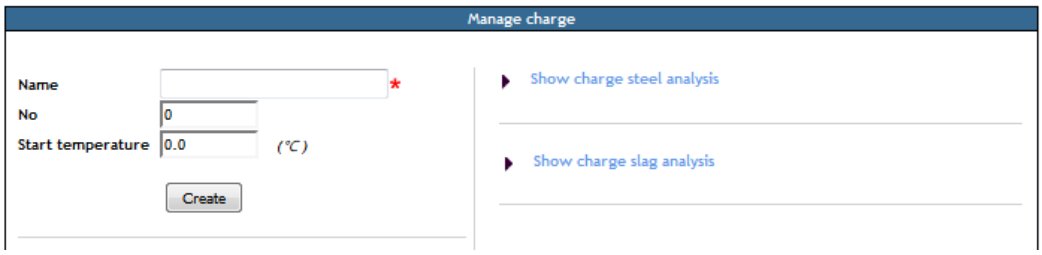
I detta steg kan användare definiera sina parametrar, avsvavlings steg och slut produkter. I figur 8 visas gränssnittet för att utföra detta.



Figur 8. Visar gränssnittet där ståltyp definieras.
(I trädmenyn till vänster har användare möjlighet att definiera andra basobjekt.)

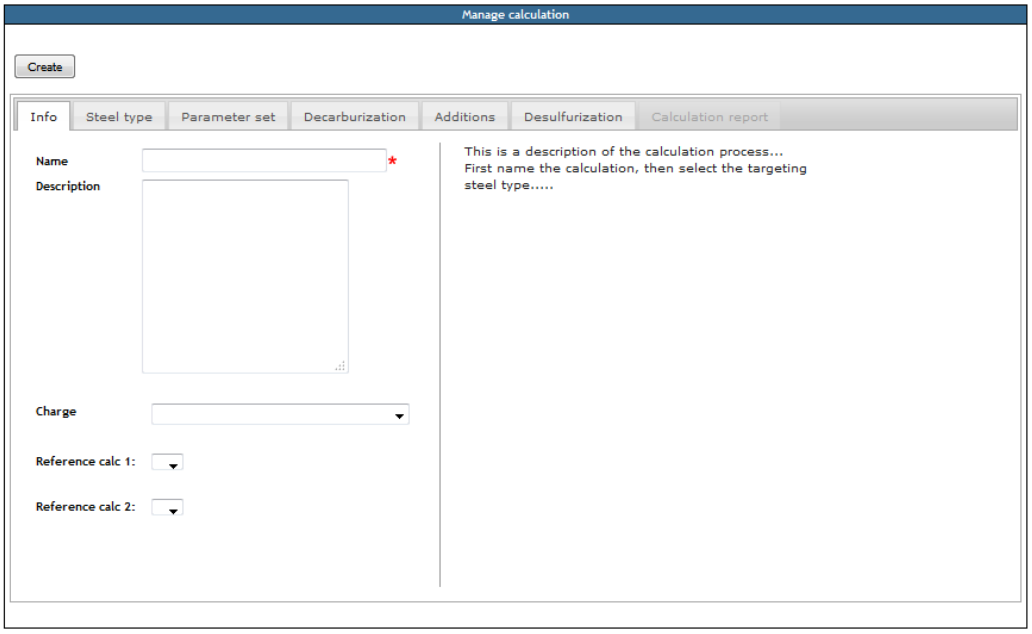
3.3.1.2 Steg 2, Definiering av charge och uträkning

I detta steg kan användare definiera sina charger med all dess data. Se figur 9.



Figur 9. Visar gränssnittet för att skapa en charge.

När allt ovanstående är definierat, kan användaren göra en beräkning på en charge. För att utföra detta gör användaren en ny beräkning och namnger denna. Sedan väljs alla definitioner en beräknings behövs för att köras. Se figur 10.



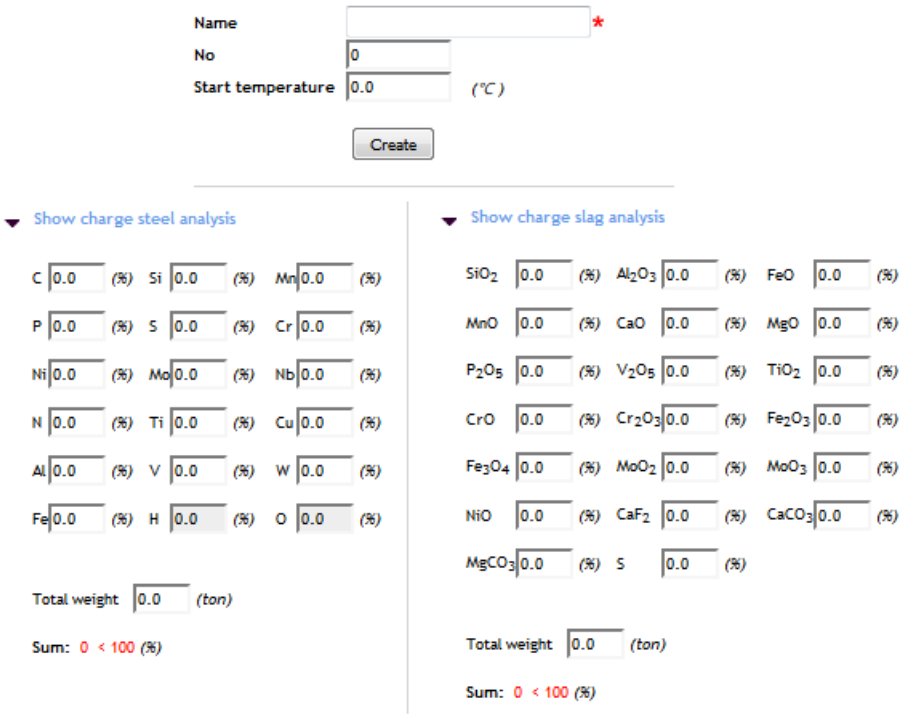
Figur 10. Visar gränssnittet för att skapa en uträkning.

3.3.2 Alternativ för design

En charge, ett parameterset, ett avsvavlingssteg och en slutprodukt kan presenteras på ett flertal sätt. I avsnitt 3.3.3.1–3.1.1.4 diskuteras hur detta kan lösas på bästa sätt.

3.3.2.1 Presentation av en charge

En charge består av en mängd parametrar. Dessa visas i figur 11.



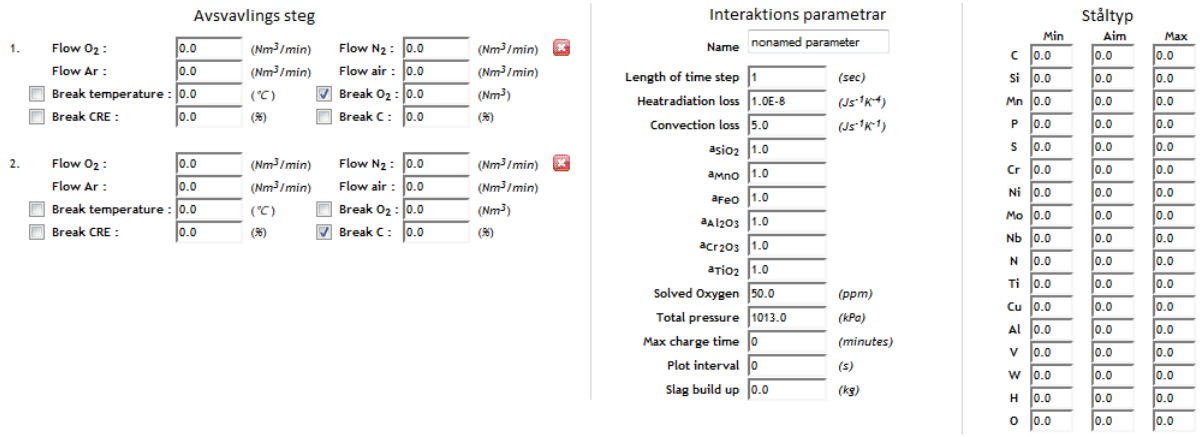
Figur 11. Visar en charges parametrar.

Alla dessa data är förstås inte relevanta för att skapa en simulering. Användare måste i simuleringen välja en charge som simuleringen ska baseras på. För att göra det enkelt för användaren samt följa hur det tidigare är gjort i TimeAOD 2, presenteras valet av charge med hjälp av menyer av typen drop-down som visar chargers namn. I figur 8 kan man se att för varje ny charge användaren skapar så måste den namnges.

För att underlätta när användare ska ange osäkerheten i chargens parametrar ska dessa presenteras på gränssnittet. Även detta var ett önskemål från projektansvarig.

3.3.2.2 Presentation av parametersets, avsvavlingssteg och produkt

Definitionerna för parameterset, avsvavlingssteg och produkt består utav en mängd värden, enligt figur 12.



Figur 12. Avsvavlingssteg, interaktionsparametrar och ståltyps parametrar.

Dessa definitioners parametrar är inte nödvändiga för användare i detta skede. De osäkerheter som ska anges är bara i chargens parametrar. Därför bör valen av dessa definitioner presenteras som charger, via namn i menyer av typen drop-down.

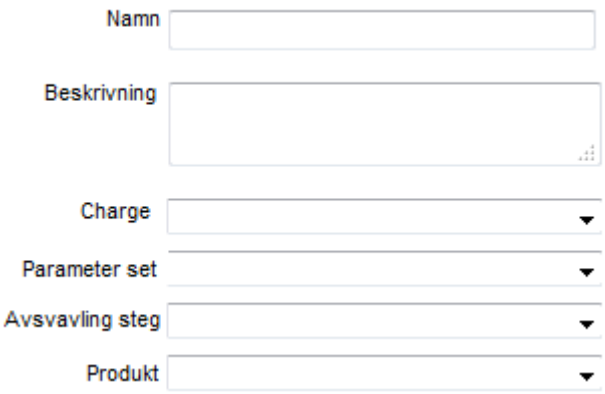
3.3.2.3 Presentation av sparade simuleringar

För att kunna följa TimeAOD 2s struktur kommer sparade simuleringar visas på ett liknande sätt som sparade definitioner. Strukturerna visas i en trädmeny på vänster sida i gränssnittet. Här kan användaren utöka eller minska de kategorier som finns att välja mellan. Med utökning av en kategori menas att alla sparade strukturer presenteras och fungerar som länkar till respektive information och handlingar. En utökad kategori ger också möjligheten att spara strukturen på nytt. Med minimering av en kategori menas att samtliga sparade strukturer inte presentas för användaren. På detta sätt kan användare enkelt få en överblick över sparade filer. Se figur 13.

Figur 13. Visar trädmenyn för basobjekt.

3.3.2.4 Design av formulär

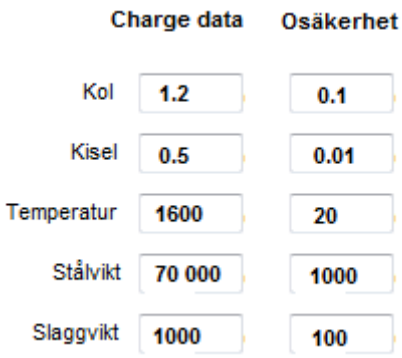
En skiss togs fram hur en del av formuläret kan tänkas se ut. Se figur 14.



Figur 14. Visar skiss över tilltänkt formulärdesign.

Här kan användaren namnge sin simulering samt beskriva den. Användaren kan välja mellan sina sparade definitioner som simuleringen ska baseras på, exempelvis presenteras alla sparade charger i en meny av typen drop-down. Detta sätt att välja bland sparade data används på flera ställen i TimeAOD 2 för att programmet ska ha lika struktur, så även skapandet av ny simulering.

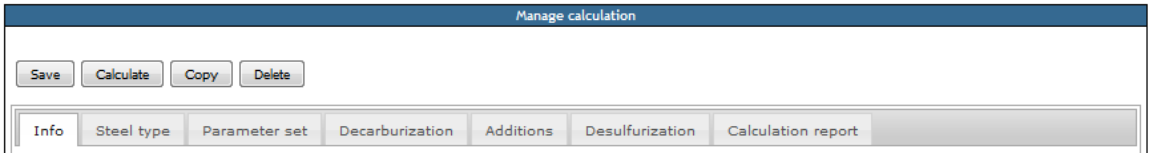
Simuleringarna baseras på osäkerhet i mätdata. Därför ska det vara enkelt att ange osäkerhet i formuläret. Programmet bör ge användaren så mycket hjälp som möjligt. Detta åstadkoms genom att visa chargens parametrar och respektive osäkerhet. Under design av programmet togs en skiss fram för ändamålet. Skissen visas i figur 15.



Figur 15. Skiss över hur osäkerhet kan presenteras.
(Värdena i figuren är exempel.)

3.3.2.5 Sparande, uppdatering och borttagning av en simulering

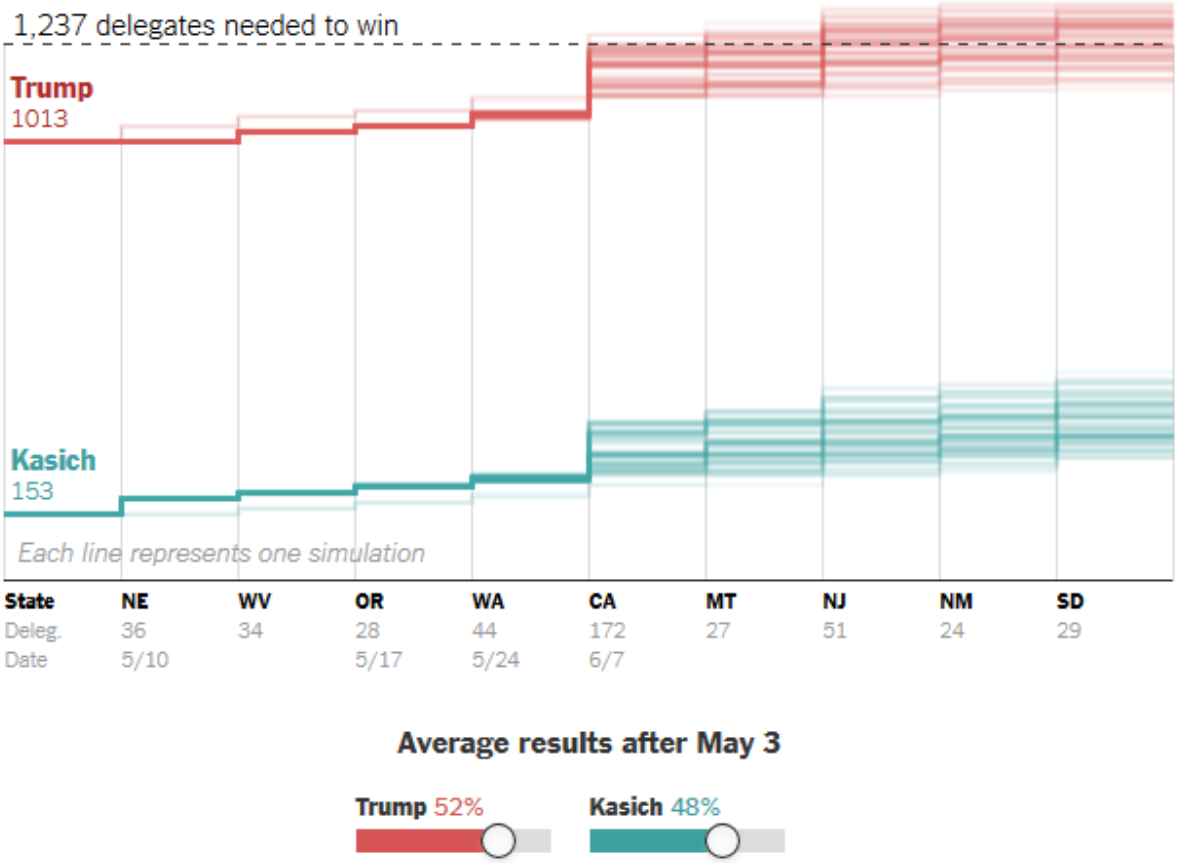
Simuleringens funktioner är viktiga och måste enkelt kunna kommas åt av användare. Dessa funktioner kommer att köras via knappar i gränssnittet. Samma funktionaliteter finns redan i TimeAOD 2 när en beräkning görs eller har gjorts. Simulatorns gränssnitt kommer därför att efterlikna placering av dessa knappar. Knapparna är positionerade högst upp till vänster när man befinner sig i en vald beräkning, i enlighet med figur 16.



Figur 16. Knapparnas position efter att användaren har valt en beräkning.

3.3.2.6 Visualisera en AOD-process

Efter egna undersökningar har ingen tidigare publicerat någon slags visualisering av en AOD-process. I TimeAOD 2 presenteras mätresultaten av en beräknad charge med grafer och tabeller av data. Visualisering av en AOD-process inklusive respektive värdes osäkerhet, är ett nytt sätt att presentera sådana processer. Dock är visualisering av andra processer med osäkerheter inte helt ovanligt. I figur 17 visas ett exempel på hur osäkerheter kan presenteras med hjälp av en graf.



Figur 17. Visualisering av osäkerhet med hjälp av grafer.
(I detta fall visas valresultat för två amerikanska republikaner efter den 3 maj, 2016.)

Figur 17 är tagen från New York Times där osäkerheten visas med hjälp av en graf. Här har man möjligheten att simulera antalet röster för varje kandidat och på så sätt se ett slutresultat. [10]

3.3.2.7 Relevant data under en simulering

Processdata för en beräkning presenteras i TimeAOD 2 genom en graf. En sådan graf visar mätvärden för endast en beräkning. Men en simulering av en charge med mätfel bör innehålla åtminstone ett tiotal beräkningar för att få säkra resultat.

Företagshandledarens önskemål för presentation av processdata var att kunna jämföra processdata med och utan osäkerheter. Detta kan åstadkommas genom att presentera processdata på två grafer. En graf visar resultatet utan osäkerheter och en annan graf visar resultat men osäkerheter.

I en beräkning med osäkerheter skiljer sig också stålets sammansättning under hela processen. Genom att presentera detta på ett bra sätt kan användare få en överblick hur sammansättningen ser ut under en process. Inspiration för visualisering av processer är inte svårt att hitta. Exemplet i kapitel 3.3.2.6 är ett sätt att visualisera osäkerhet på. Tanken för AOD-processen var att kunna presentera värden för legeringsämnen ungefär så som är gjort i exemplet. Desto större osäkerhet ju större område täcks av värdet i grafen.

4 Metoder och verktyg

4.1 Metoder

Projektet inleddes med att få tillgång till den befintliga programvaran samt en genomgång av det som skulle göra mer exakt. Projektet hade inga tydliga riktlinjer utan det var mina egna idéer som fick styra arbetet. Företagshandledaren bidrog med tips och synpunkter. Testning utav gränssnittet pågick under hela projektettiden då alla funktioner testades och utvärderades allt eftersom de implementerades. Arbetet dokumenterades i en blogg på universitets Blackboard, där man kunde följa projektet med dagliga uppdateringar.

4.2 Verktyg

- Programmeringen gjordes i Netbeans.
- Kobolde & Partners erbjöd en dator samt en arbetsplats.

4.3 Övriga resurser

4.3.1 Apache Struts

Apache Struts är ett ramverk för att skapa Java-webbapplikationer. För att uppmuntra utvecklare att använda sig utav arkitekturen Model-View-Controller (MVC) använder Struts sig av Java Servlet API:n. Med Struts har man möjlighet att skapa webbsidor med dynamisk respons. En dynamisk webbsida har möjlighet att interagera med databaser och på så sätt ha möjlighet att anpassa utseendet, exempelvis beroende på vem klienten är. [11]

4.3.2 JFreeChart

JFreeChart är ett ramverk med öppen källkod som kan användas ihop med språket Java. Ramverket erbjuder ett brett utbud utav både interaktiva och statiska diagram för webbapplikationer med möjlighet att exekveras både på klient- och serversidan. [12]

4.3.3 Random gaussian

Random gaussian är en funktion som finns i programbiblioteket Random. Funktionen genererar slumpmässiga tal med normalfördelning. Funktionen kommer till användning då ny värden behöver slumpas inom ett bestämt intervall vid beräkningar av en charge. [13]

5 Genomförande

Projektet delades in tre faser varav två kan läsas ur specifikationens tidsplan. Under fas ett utfördes informationssökning samt skissningar på design av gränssnittet. Fas två som inte kan läsas ur tidsplanen gick ut på att bli bekant med koden och strukturen i TimeAOD 2. I fas tre utfördes själva projektet, implementering av gränssnittet.

5.1 Fas 1: informationssökning

5.1.1 AOD-process

Förståelse för hur en AOD-process fungerar underlättar arbetet avsevärt. I projektet ingick en guidad tur på stålverket Outokumpu i Avesta. Turen gav inblick hur framtagning av rostfritt stål går till och en beskrivning av AOD-processen gavs i kontrollrummet, där hela processen kan följas från start till slut. Med hjälp av besöket på stålverket samt förklaringar av AOD-processen, i synnerhet intressanta data, som företagshandledare och anställda på Kobolde & Partners gav, underlättades designen av gränssnittet.

.

5.1.2 Design av gränssnitt

Ett antal artiklar om hur bra gränssnitt utvecklas söktes fram och analyserades. Eftersom TimeAOD 2 redan hade ett gränssnitt, behövdes endast en ny flik läggas till för simulatorn. Under denna flik behövde data läsas in för att sedan kunna köra igång simuleringen. Ett par skisser togs fram över hur en flik som byggs upp av formulär skulle kunna se ut. Gränssnittet för simuleringsdelen skulle visa relevanta data i grafer, och eventuellt också visa vilken beräkning varje simulation är baserad på.

5.1.3 Osäkerhet i mätdata

Visualisering av osäkerhet i mätdata var en viktig del av detta arbete. Under en AOD-process tas prover manuellt och analyseras. Sådana provresultat är ett exempel på var man kan hitta osäkerhet i mätdata. Ett annat exempel är vikten på chargen: Vågarna i sig har en felmarginal och en våg kan även vara felkalibrerad. Detta generar fel i mätdata.

5.2 Fas 2: uppdatering av studie av befintligt program

5.2.1 Uppdateringar

Eftersom ingen har arbetet med TimeAOD 2 under de senaste 10 åren, har koden hunnit bli gammal och följer inte senare HTML-versioner. På grund av detta fungerade inte en del viktiga funktioner i programmet och för att få igång dessa utfördes uppdateringar av koden.

5.2.2 Struktur

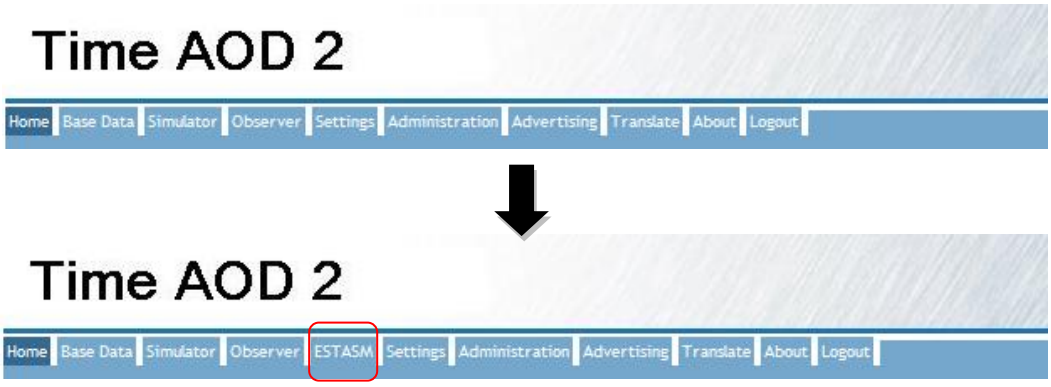
För att förstå hur ett Struts-projekt fungerade skapades nya, mindre exempelprogram som användes för tester av kommunikationerna mellan delarna [3]. Med hjälp av dessa exempelprogram och handledning som Koboldes personal gav, kunde steg tre inledas.

5.3 Fas 3: implementering av gränssnitt

Implementering av gränssnitt utfördes efter tidigare skisser och idéer som diskuterades i kapitel 3.

5.3.1 Befintliga gränssnittet

Eftersom TimeAOD 2 redan hade ett webbaserat gränssnitt, utökades det med ytterligare en flik kallad ESTASM, där den nya simulatorn visades. Se figur 18.



Figur 18. Den nya menyn med ESTASM-fliken.

5.3.2 Databas

TimeAOD 2 är kopplad till en databas och har där möjligheten att spara charger, beräkningar och liknande. För kunna spara simuleringar krävdes en ny tabell i den befintliga databasen. Den nya tabellen som skapades i databasen följde riktlinjerna som togs fram och såg ut som följande:

```
create table AOD_ESTASM_SIMULATION (
    ESTASM_SIM_ID number,
    CALC_ID number not null,
    SIM_NAME varchar2(800 byte),
    SIM_DESCRIPTION varchar2(1000 byte),
    UNCERTAINY_C number,
    UNCERTAINY_SI number,
    UNCERTAINY_T number,
    UNCERTAINY_WSTEEL number,
    UNCERTAINY_WSLAG number,
    NBR_SIMULATIONS number,
    primary key (ESTASM_SIM_ID)
);
```

Figur 19. Kod för att skapa en ny tabell.

5.3.3 Formulär

Programmets tydliga tillvägagångssätt är följande:

Logga in → Definiera basdata (parameterset, avsvavlingssteg och slutprodukt) → Gör beräkningar (skapa en charge, skapa en beräkning genom att ange basdata och charge)

Formuläret skapas till att börja med efter de skisser som tidigare tagits fram. Efter tester av beräkningar och utökad förståelse för programmet redigerades formuläret. Tidigare hade användaren valt charge, parametrar, avsvavlingssteg och produkt i skapandet av en ny simulering. Detta gjordes i steget att skapa ny beräkning. Istället gjordes formuläret om så att

endast beräkningar fanns att välja mellan vid skapandet av simuleringen. Detta följer bättre programmets tillvägagångsätt och underlättar för användare.

Koden läser in chargen som beräkningen är baserad på och presenterar chargens parametrar, enligt skissen i figur 14. Detta gör det lättare för användare att sedan ange osäkerheten i dessa parametrar.

5.3.3.1 Formulärets funktioner

Sparande, uppdatering och borttagning av en simulering utförs via knappar i formuläret. Dessa knappar är kopplade till funktioner som i sin tur är hopkopplande med databasen. På så sätt utförs de handlingar som valts.

Ytterligare en viktig funktion som körs via en knapp är simuleringen. Knappen positioneras bredvid knapparna för att spara och ta bort. Se knappen Simulate i figur 22. Funktionen öppnar ett simuleringsfönster och kör igång alla beräkningar som ska utföras enligt formuläret.

5.3.4 Trädmeny

Med hjälp av serverkoden laddas alla sparade simuleringar in till webbläsaren och presenteras i en trädmeny. Se exemplen i figur 20 och figur 21. Ett JavaScript gör det möjligt att minimera respektive utvidga trädmenyn. Detta utvecklades för eventuella framtida tillägg. På så sätt kan sparade simuleringar tydligt separeras från andra, eventuella sparningar.

5.3.5 Simulering

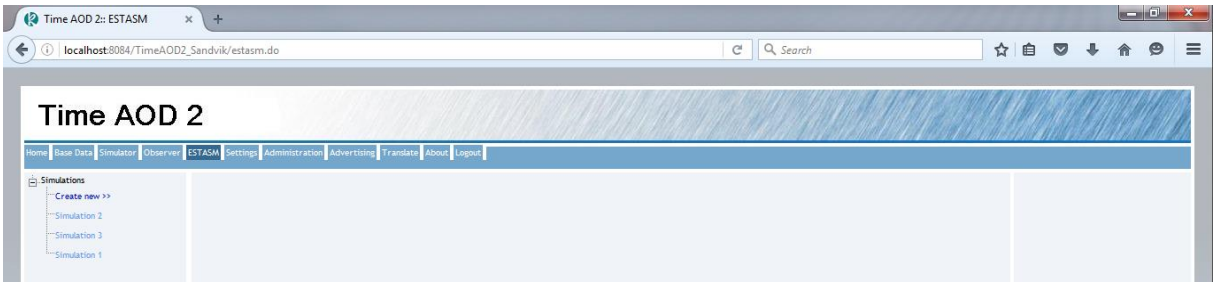
I en simulering utförs angivet antal beräkningar på en charge. Chargens parametrar ändras efter varje beräkning, där varje parameter får ett nytt värde inom respektive angivet intervall av osäkerhet. Resultatet presenteras med hjälp av grafer och med hjälp av dessa kan man se hur processen kan se ut med den bestämda osäkerheten för respektive parameter.

5.3.6 Testning

Testning skedde under hela implementeringsfasen av gränssnittet. När nya funktioner skapats på servern, utvecklades gränssnittet för att fungera ihop med dessa. Därefter utfördes tester för att bekräfta att koppling mellan gränssnitt och server fungerar som planerat. Större tester utfördes när projektet var klart. Alla funktioner som skapats i detta testades och ett antal simuleringar kördes för att upptäcka eventuella fel i beräkningarna.

6 Resultat

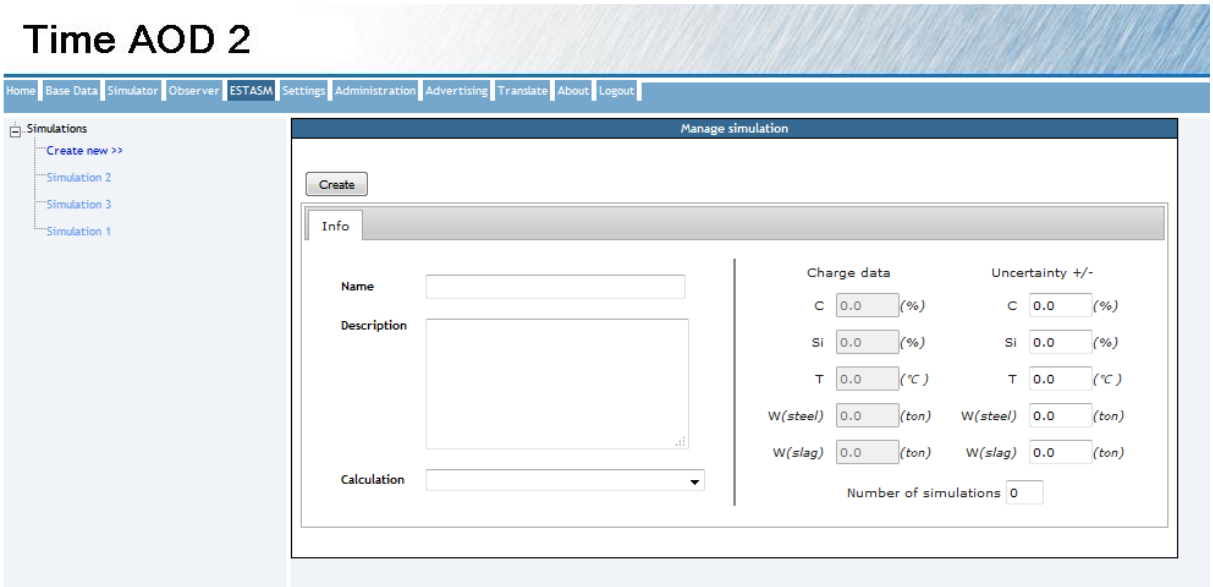
6.1 Resultatet av de hårda kraven



Figur 20. Gränssnittets menyer samt innehållet på fliken ESTASM.

ESTASM är namnet på projektet som Kobolde & Partners deltar i och AOD-simulatorn är en del av detta projekt.

På ESTASM-fliken får användaren möjlighet välja mellan tidigare sparade simuleringar eller skapa en helt ny simulering genom att klicka på ”Create new >>>”. I figur 20 har tre simuleringar redan lagts till. Presentation av sparade simuleringar visas i en trädmeny till vänster i gränssnittet enligt figur 20. På resterande flikar i TimeAOD 2 som presenterar sparade filer, listas dessa på liknande sätt för att den nya simulatordelen ska efterlikna det övriga programmet.



Figur 21. Formuläret ”Create new”.

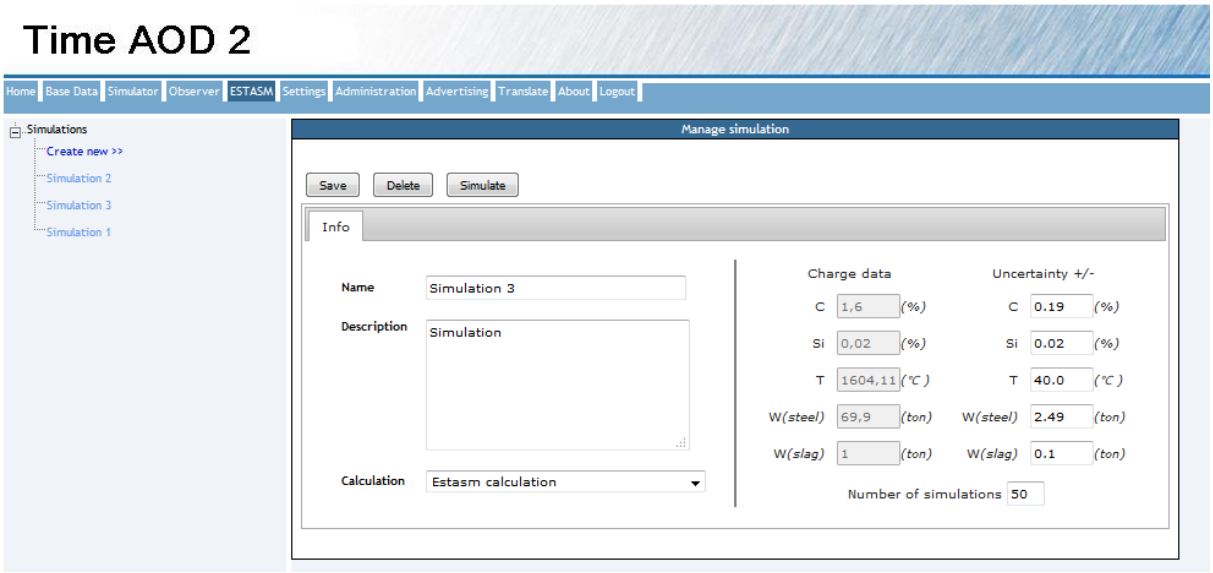
Figur 21 visar formuläret för att skapa en ny simulering. Här kan användaren döpa simuleringen, ge den en beskrivning och välja bland sparade beräkningar som simuleringen ska baseras på. Efter vald beräkning presenteras chargens data vilket underlättar när användaren ska ange osäkerheten.

Med val av antal simuleringar menas hur många beräkningar som ska utföras med den angivna osäkerheten. Fler beräkningar medför ett säkrare resultat men ökar samtidigt tiden för att utföra en simulering.

Formuläret efterliknar TimeAOD 2s övriga formulär för skapande av nya filer. Även de tre grundläggande riktlinjerna som tas upp i kapitel 1.5.1 fanns i tankarna vid skapandet.

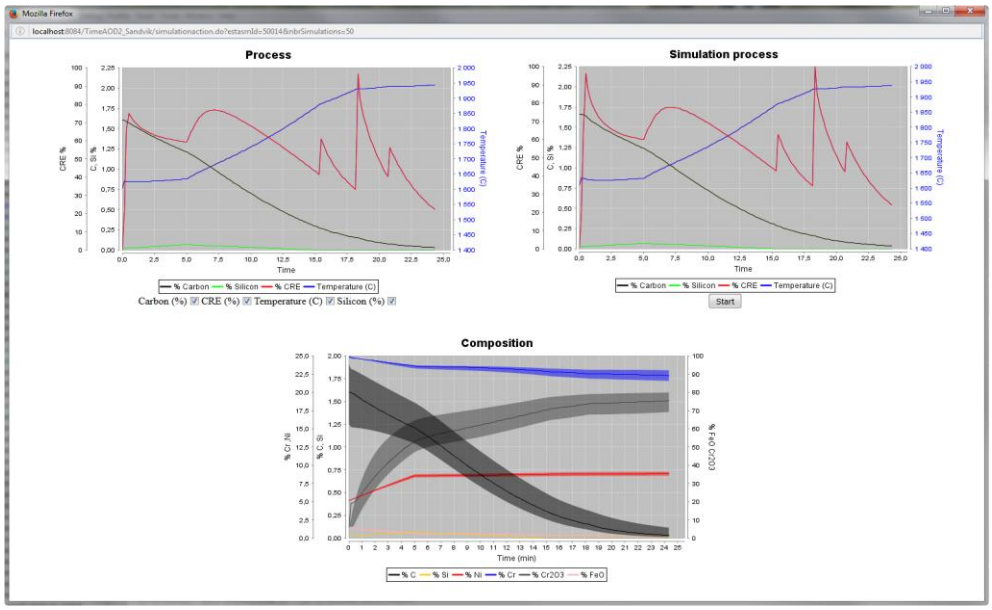
Inför ny simulering har användare full kontroll över vyn. Här får användaren all hjälp som behövs. I val av uträkningar presenteras alla sparade uträkningar med namn. Detta medför mindre minnesbelastning än tidigare skisser där charge, parameterset, avsvavlingsteg och ståltyp skulle väljas. Det enda som användaren behöver komma ihåg från den föregående fliken ”Simulator” är namnet på uträkningen.

Positionering av knappar som kan ses i figur 21 och figur 22 efterliknar TimeAOD 2s tidigare formulär. Knapparnas funktioner är välbeskrivna och ”tooltips” hjälper användaren när muspekaren förs över knapparna.



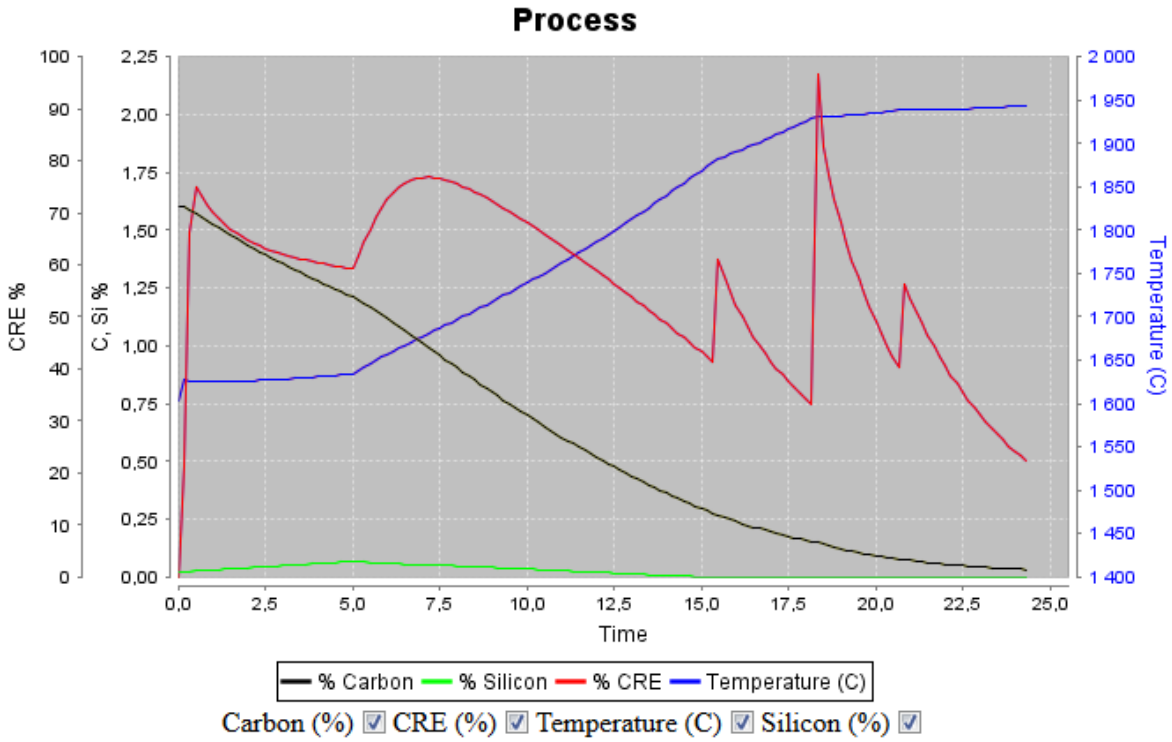
Figur 22. Vald simulering.

När användaren skapat en ny eller valt en befintlig simulering, visas vyn enligt figur 22. I figuren visas ”Simulation 3” som är skapad sedan tidigare. Användaren har här möjligheten att utföra ändringar i den sparade simuleringen eller att ta bort den helt. Med knappen Simulate startas simuleringen med tidigare sparade värdena.



Figur 23. Fönstret för resultat av simuleringen.

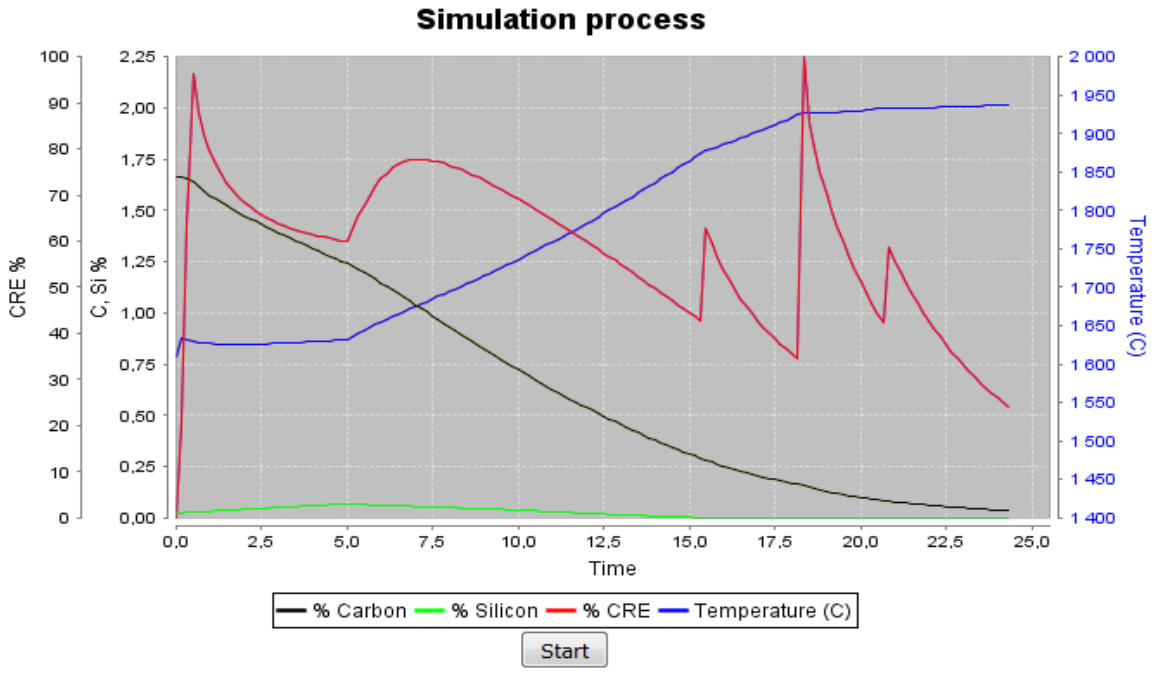
När användaren har klickat på Simulate, utförs de antal beräkningar som tidigare har angetts och mätresultaten visas med hjälp av grafer, enligt figur 23. Mätresultaten visas i ett nytt fönster för att användaren ska få en överblick över alla grafer.



Figur 24. Processgrafen från simuleringsvyn.

Processgrafen i figur 24 visar en förstorning av processgrafen som finns simuleringsfönstret. I

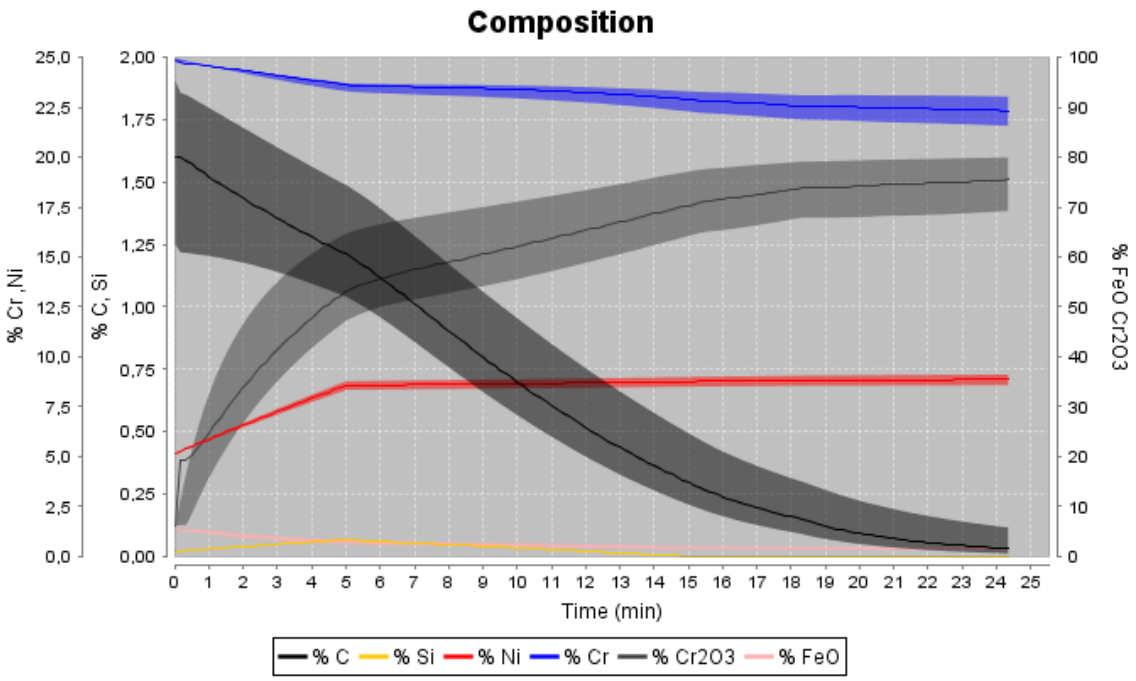
denna graf presenteras chargens processdata utan angivna osäkerheter. Genom att bocka i/av de olika parametrarna anpassas grafen efter de resultat man vill se.



Figur 25. Simuleringsprocessgrafen från simuleringsvyn.

Grafen i figur 25 visar förstorning av simuleringsprocessgrafen från simuleringsfönstret. Här presenteras processdata från chargen med osäkerheter inräknade. Endast en uträkning visas åt gången. Eftersom en uträkning tar cirka 4 sekunder, uppdateras grafen efter detta. Tidigare värden sparas inte i grafen, endast den senaste uträkning visas. Start/stop-knappen gör det möjligt för användare att pausa närsomhelst under processen. På så sätt kan grafen granskas noggrannare samt jämföras med processgrafen, där osäkerheterna inte är inräknade.

Resultaten som visas i figur 25 och figur 26 är utförda med överdrivet stora osäkerheter för att få så tydlig bild över funktionaliteten.



Figur 26. Sammanställningsgrafen från simulations vyn.

I sammanställningsgrafen, enligt figur 26, kan man följa stålets sammansättning under hela processen. Det tydligt, dragna strecket för varje parameter representerar chargens mätresultat utan osäkerhet. De skuggade områdena representerar observerarens skattade processdata. Genom denna graf kan man se inom vilka intervall stålets sammansättning ligger med den av användaren givna osäkerheten.

Sammansättningsgrafen är inspirerad från de tidigare visade exemplen på hur osäkerhet kan visualiseras. Presentation av osäkerheter med hjälp av grafer ger en tydlig överblick som i figur 25. Resultat med osäkerheter kan presenteras på ett flertal sätt, t ex skulle halten av kol och krom kunna skrivas som i tabell 1.

Ämne	Resultat %	Osäkerhet ±
Kol	0.1	0.05
Krom	23	0.5

Tabell 1. Exempel på tabell för presentation av halten av kol och krom i en stålblandning.

Tabell 1 visar endast slutresultatet. Stickprover under processen skulle generera betydligt fler Mätresultat som kan presenteras. Med hjälpa av en graf visas hela processen med dess osäkerheter. Därför är grafer är ett av de bästa sätten att visa mätvärden under processens gång.

7 Diskussion

7.1 Uppfyllande av projektets krav

7.1.1 Hårda krav

Det utvecklade gränssnittet tog plats i TimeAOD 2s befintliga gränssnitt. Vid designen av gränssnittet togs hänsyn till TimeAOD 2s befintliga design samt grundläggande riktlinjer för design av ett gränssnitt. Dessa riktlinjer tas upp i kapitel 1.5.1. Resultat av designen på formuläret för skapande/hantering av simuleringar tycker jag stämmer bra med riktlinjerna.

Användaren behöver inte minnas alla detaljer från en vy till en annan. Sådana krav har minimerats. Det som användaren behöver komma ihåg från ett tidigare steg i TimeAOD 2 presenteras med hjälp av exempelvis tidigare beräkningar som visas i menyer av typen drop-down. För hjälpa användaren ytterligare i formuläret finns så kallade ”tooltips” för varje input som kräver värden.

För att ge användaren kontroll över sin simulering och kan värden anges fritt, exempelvis osäkerheten för en parameter.

Gränssnittet för simuleringsresultat presenteras med ett fönster av typen pop-up på full skärm. Detta ger användaren god överblick över graferna och full fokus på fönstret i simuleringsvyn. Fönstret som visas i figur 23, har varken trädmeny eller knappar.

Vid design av gränssnittet granskades valet av färger framför allt i graferna för att passa så många användare som möjligt. I kapitel 1.5.1 diskuteras vikten av färger i ett gränssnitt. De valda färgerna ska nämligen vara tydliga även för synskadade personer.

I skapandet av en ny simulering har användare möjlighet att ange osäkerheter i utvalda parametrar samt hur många uträkningar simulering ska baseras på. Efter detta kan simuleringen köras igång. Under simuleringsprocessen visas stålets sammansättning med hjälp av en graf i ett fönster av typen pop-up.

7.2 Projektets utvecklingspotential

Arbetet med TimeAOD 2 kommer att fortgå efter kursen slut. Eventuellt skulle det kunna bli aktuellt att färdigställa observeraren som i nuläget endast är en testversion. I den första projektplanen beskrevs en sensorsimulator och om installation av denna blir, av kommer programmet att byggas ut för denna.

7.2.1 AOD-Simulatorn

Möjligheten att ange mätfel i flera parametrar kan komma att läggas till under efterarbetet.

För att kunna se hur behandlingstiden skiljer sig med osäkerheter behövs ytterligare en graf i fönstret för simuleringsvyn. Även denna graf kommer att läggas till under efterarbetet.

7.3 Reflektion kring eget lärande

Innan detta projekt hade jag ringa kunskap om hur webbserverprogram fungerar. Efter detta arbete har förståelsen ökat betydligt. Att få arbeta i ett stort projekt som detta har varit lärorikt och kommer definitivt att vara till hjälp i framtiden. Att ha insikt i ett webbserverprojekt, där en databas finns med, är viktig kunskap för kommande webbserverprojekt med liknande struktur.

Arbetet utfördes på Kobolde & Partners kontor i Stockholm, där projektet utfördes självständigt, men med viss hjälp av anställda på Kobolde.Hjälpen bestod av att förklara den verkliga AOD-processen (AOD) som programmet TimeAOD 2 simulerar och hur programmet skulle fungera vid projektets slut. Däremot utfördes all programmering egenhändigt.

Att utföra detta arbete på plats hos företaget har varit lärorikt, framför allt genom gott samarbete med kunnig personal i projektform.

8 Referenser

[1] Kobolde & Partners, hemsida. Stockholm: Kobolde & Partners AB, 2014.
Besöktes 2016-03-24.
URL: http://www.kobolde.com/

[2] Ternstedt, Patrik; Gyllenram, Rutger; Bengtsson, Johannes; Jönsson, Pär G., *Using an AOD simulator workbench to support process control development*. Düsseldorf: METEC InSteelCon 2011, 4th International Conference on Modelling and Simulation of Metallurgical Processes in Steelmaking (STEELSIM), 2011.
Besöktes 2016-03-24.
URL: https://www.researchgate.net/publication/278960236_Using_an_AOD_simulator_workbench_to_support_process_control_development

[3] Mandel, Theo, *The Elements of User Interface Design*. 1 uppl. Indianapolis: Wiley, 1997 – ISBN: 0471162671

[4] Galitz, O. Wilbert, *Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques*. 3 uppl. Indianapolis: Wiley Publishing 2007 - ISBN: 978-0-470-05342-3

[5] You, Nathan, *Data points: visualization that means something.* 1 uppl. Indianapolis: John Wiley & Sons 2013 – ISBN: 978-1-118-46219-5

[6] Ridgen, Christine, ”Now You See It, Now You Don’t”, *Computer,* s 104 f, Issue 7. IEEE, 2002 - ISSN: 0018-9162

[7] Brodlie, Ken; Osorio, Rodolfo A; Lopes, Adriano, “A Review of Uncertainty in Data Visualization”, *Expanding the Frontiers of Visual Analytics and Visualization,* ss 81-109. 1 uppl. London: Springer-Verlag London, 2012 – ISBN: 978-1-4471-2803-8

[8] Bell, Stephanie, *A Beginner’s Guide to Uncertainty of Measurements,* No.11, uppl 2. 2001. ISSN: 1368-6550

[9] Meyer, Veronika R, “Measurements uncertainty”, *Journal of chromatography,* Vol 1158, ss 25-24. 2007.

[10] The New York Times, hemsida.
Besöktes 2016-05-17.
URL: http://www.nytimes.com/interactive/2016/03/30/upshot/trump-clinton-delegate-calculator.html?_r=0

[11] Apache Struts, hemsida.
Besöktes 2016-05-02.
URL: https://www.struts.apache.org/birdseye.html

[12] JFreeChart, hemsida.
Besöktes 2016-05-02.
URL: http://www.jfree.org/jfreechart/

[13] Programbibliotek för Random, hemsida.
Besöktes 2016-05-02.
URL: https://www.docs.oracle.com/javase/7/docs/api/java/util/Random.html