

# Systemtesting for SheepTracker 1.0

## GRUPPE 12

INGRID WATNEDAL MYRANN

EIVIND MORCH

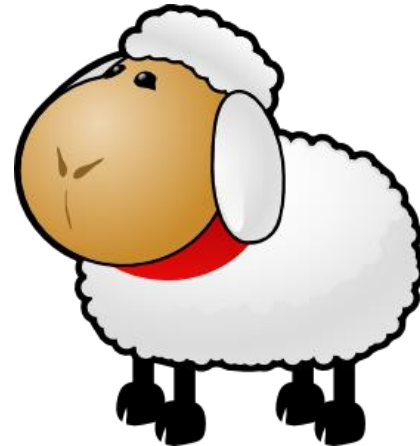
KJERSTI FAGERHOLT

MAGNUS NERMARK

PAUL PHILIP MITCHELL

NTNU

IT 1901 – Prosjektarbeid del 1



## Innholdsfortegnelse

Test 01 – Login med bruker .....	2
Test 02 – Registrering av bruker .....	3
Test 03 – Oppdatering av bruker .....	5
Test 04 – Endring av passord .....	7
Test 05 – Registrering av ny sau .....	8
Test 06 – Sau finnes på kart og koordinater er tilgjengelig .....	9
Test 07 – Valgt sau viser korrekt informasjon med god responstid .....	11
Test 08 – Sletting av sau .....	12
Test 09 – Filtrering og søk .....	14
Test 10 – Posisjonslogg .....	16
Test 11 – Endring av sauevekt .....	17
Test 12 – Visning av alle sauer .....	18
Test 13 – Sletting av logg .....	19
Test 14 – Alarmmelding til brukeren .....	20
Test 15 – Maks kapasitet på antall sauer .....	21
Test 16 – Maks kapasitet på antall brukere .....	22
Test 17 – Dimensjonering: forespørsler og rapportering .....	24
Test 18 – Utilgjengelighet, opptid & operativitet .....	25

## Test 01 – Login med bruker

---

### Beskrivelse

Testen skal sikre at brukere kan logge seg inn på systemet, gitt at brukeren allerede er registrert.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker åpner opp systemet, fyller inn tekstboksene for henholdsvis «e-mail» og «passord» og deretter trykker på «Login»-knappen. Dette gjøres flere ganger med forskjellige typer input.

Hva slags output er godkjent?

Dersom brukeren logger inn med korrekt e-mail og passord, samt at denne brukeren finnes i databasen, vil testen regnes som godkjent. I tillegg, dersom brukeren ikke finnes i systemet og vi får feilmelding, godkjennes det.

Hvilke situasjoner skal skape brudd?

1. Feltet for e-mail er ikke fylt ut
2. Feltet for passord er ikke fylt ut
3. E-mailen finnes ikke i databasen
4. Passordet stemmer ikke med e-mailen
5. Ingen internettkobling
6. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent fem minutter.

Hvorfor er denne testen viktig?



Denne testen sikrer at brukeren faktisk kan logge seg inn i systemet. Dersom brukeren kan logge seg inn, vet vi at vår løsning for passord-kryptering fungerer. Dette er svært viktig, slik at brukerenes passord ikke kan hentes ut av eventuelle administratorer som håndterer databasen. I tillegg vil suksessrik login sikre at formatet på mailen til den innloggede bruker er riktig, slik at mail kan sendes ut ved en eventuell alarmsituasjon.

### **Godkjent.**

## Test 02 – Registrering av bruker

---

### Beskrivelse

Testen skal forsikre oss om at brukere kan registrere seg selv på systemet og at duplikater av samme bruker ikke er tillatt.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker åpner opp systemet, navigerer seg fram til «registrering»-tabben, fyller inn tekstboksene på følgende måte; først ingen, deretter noen, så alle. I tillegg vil vi teste om brukeren kan fylle inn en e-mail på invalid format, om passordene stemmer overens, om e-mailen allerede er brukt og om brukeren har fylt inn koordinater ved å trykke på «hente posisjon»-knappen. Deretter trykker brukeren på «registrer»-knappen. Dette gjøres iterativt med forskjellige typer inputs, der inputs er hva som fylles inn i tekstboksene.

Hva slags output er godkjent?

Dersom brukeren fyller inn korrekt informasjon med hensyn til formater o.l. i alle tekstbokser, og får oppkobling mot databasen vil testen godkjennes.

Hvilke situasjoner skal skape brudd?

1. Ikke alle felter er fylt ut (med unntak av «buddy»-feltet).
2. E-mailen er på feil format (ola@nordmann.no).
3. E-mailen finnes allerede i databasen.
4. Passordene i «passord»-feltene stemmer ikke overens.
5. Feltene for «latitude» og «longitude» er ikke fylt ut (v.h.a. «hente posisjon»-knappen).
6. Brukeren er ikke tilkoblet internett.
7. Kobling mot databasen feiler.

Estimert tid

Testen tar omtrent ti minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer i hovedsak at den informasjonen som sendes til databasen er korrekt. Dette er viktig slik at administrator har oversikt over hvem som bruker systemet. I tillegg er informasjon som korrekt e-mail, korrekte koordinater o.l. kritisk i forhold til funksjonaliteten til programmet. Man ønsker heller ikke duplikater av brukere med samme e-mail adresse da dette skaper problemer i database-systemet.

**Godkjent.**



## Test 03 – Oppdatering av bruker

---

### Beskrivelse

Testen skal sikre at brukere har mulighet til å oppdatere informasjonen om seg selv, gitt at mail-adressen er hans egen og/eller ikke finnes blant andre brukere av systemet og at passordet er korrekt. I tillegg sjekker vi her om loggføring foregår etter kriteriene.

### Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker åpner opp systemet, fyller inn tekstboksene for innlogging og dermed blir viderekoblet til «hoved»-vinduet. Her skal brukeren endre på feltene for henholdsvis «Fornavn», «Etternavn» og «E-mail». Deretter skal brukeren klikke på «Oppdater profil»-knappen. Samme prosess gjøres igjen ved å sette en venn som reserve, og for å oppdatere posisjonskoordinater. Til slutt navigerer brukeren seg til «Logg»-fanen for å se at endringer har blitt registrert.

### Hva slags output er godkjent?

Dersom brukeren fyller inn en gyldig e-mail, da at mailen er 1. uforandret eller 2. ikke finnes i databasen fra før, samt at passordet for den innloggete brukeren er korrekt, vil testen godkjennes. Deltesten for loggføring godkjennes dersom en ny rad i «Logg»-fanen har blitt opprettet for hver gang en gyldig oppdatering skjer.

### Hvilke situasjoner skal skape brudd?

1. Feltet for e-mail er ikke fylt ut
2. Feltet for passord er ikke fylt ut
3. E-mailen finnes allerede i databasen hos en annen bruker
4. Passordet stemmer ikke med e-mailen
5. E-mailen er ikke på riktig format ([ola@nordmann.no](mailto:ola@nordmann.no))
6. Loggføring skjer ikke ved trykk på «Oppdater profil»
7. Ingen internettkobling
8. Kobling mot databasen feiler

### Estimert tid



Testen tar omtrent ti minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å oppdater sin egen kontaktinformasjon. Dersom brukeren har endret sin e-post adresse er det selvfølgelig kritisk for programmets funksjonalitet at brukeren har mulighet til å si ifra om dette, slik at alarmer kan sendes ut. I tillegg må brukeren ha mulighet til å kunne endre posisjonskoordinater dersom han/hun endrer gårdsadresse. Loggføring må skje slik at brukeren har oversikt over hva som har blitt gjort til enhver tid.

**Godkjent.**



## Test 04 – Endring av passord

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å endre passord.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker åpner opp systemet, fyller inn login-informasjon og logger seg inn til hovedvinduet. Her skal brukeren trykke på «Endre passord»-knappen som åpner opp et JOptionPane. Brukeren fyller her inn sitt gjeldende passord, samt det nye passordet (to ganger, for verifisering). Vi tester også med feil gjeldende passord som input, samt å prøve å endre til gjeldende passord.

Hva slags output er godkjent?

Vi har tre cases her: 1. brukeren skriver inn sitt gjeldende passord som nytt passord, dette skal gi brukeren en melding om at det ikke er hensiktsmessig.

2. Brukeren skriver inn feil gjeldende passord, da skal ikke brukeren få skifte passord.

3. Brukeren skriver inn gjeldende passord og nytt passord, da skal passordet blir endret.

Hvilke situasjoner skal skape brudd?

1. Ingen av feltene er fylt ut.
2. Feltet for «nytt passord» og «nytt passord 2» stemmer ikke overens
3. Feltene for nytt passord er likt som gjeldende passord
4. Gjeldende passord stemmer ikke
5. Ingen internettkobling
6. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent fem minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å endre sitt gjeldende passord. Denne funksjonaliteten er viktig, siden et viktig sikkerhetsprinsipp som alle bør følge er å skifte passordene sine ofte. Det kan i tillegg hende at brukeren har registrert seg med et «test-passord» bare for å se at alt fungerer, og ønsker å endre sitt passord når brukeren er registrert.



**Godkjent.**

## Test 05 – Registrering av ny sau

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å legge til sauer i systemet.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Legg til en sau»-fanen. Her skal brukeren skrive inn henholdsvis navn på sau, fødselsår til sau og vekt til sau i de tre tekstboksene, og deretter klikke på «Lagre»-knappen. Deretter navigerer brukeren seg til «Søk etter sauer»-fanen for å sjekke om sauen har blitt lagt til i listen over alle sauer, samt at all informasjon er fylt ut korrekt. Vi sjekker også loggen for å se at alt blir loggført.

Hva slags output er godkjent?

Testen erkjennes godkjent dersom sauen blir lagt til i listen over sauer, samt at loggføringen skjer.

Hvilke situasjoner skal skape brudd?

1. Alle eller noen av feltene er ikke fylt ut.
2. Feltet for «Fødselsår» ikke er på riktig format (YYYY)
3. Feltet for «Vekt» inneholder noe annet enn hel- eller flyttall
4. Ingen internettkobling
5. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent ti minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å legge til nyer sauer i systemet. Dette er såklart helt sentralt i applikasjonen, da det hele baserer seg på sauer. Sjekken for riktig format på fødselsår og vekt er viktig slik at ugyldige verdier ikke blir sendt til databasen. Loggføring må





skje i tråd med kriteriene slik at brukeren har god oversikt over hvilke sauer som har blitt lagt til til enhver tid.

### **Godkjent.**

## Test 06 – Sau finnes på kart og koordinater er tilgjengelig

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å finne sine sauer på kartet og at koordinater blir vist korrekt.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Søk etter sauer»-fanen. Her skal brukeren klikke på en hvilken som helst sau som allerede ligger i tabellen, og dermed få ut koordinatene til denne sauen. Brukeren skal også kunne trykke på knappen «Vis på kart» for å se hvor denne sauen faktisk befinner seg. Koordinatene sjekkes opp mot Google Maps for å se at posisjonen til sauen faktisk stemmer.

Hva slags output er godkjent?

Testen erkjennes godkjent dersom sauen vises tydelig på kartet og koordinatene er klart vist. Hvis koordinatene som er vist ikke stemmer overens med posisjonen på kartet, vil testen ikke erkjennes godkjent.

Hvilke situasjoner skal skape brudd?

1. Sauelisten er ikke populert.
2. Valgt sau mangler bredde- og lengdegradskoordinater.
3. Kartet vises ikke.
4. Koordinatene stemmer ikke overens med posisjonen vist på kartet.
5. Ingen internettilkobling
6. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent fem minutter.



Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren alltid får opp riktig informasjon om valgt sau. Kartfunksjonen er svært sentral, og det er derfor viktig at koordinatene til sauene stemmer perfekt overens med posisjonen til sauene på det grafiske kartet.

**Godkjent.**



## Test 07 – Valgt sau viser korrekt informasjon med god responstid

---

### Beskrivelse

Testen skal sikre at brukere får opp fullstendig korrekt informasjon om hver sau som er valgt og at formatet til informasjonen er riktig (e.g. dersom helsetilstand < 0.15, så skal teksten markeres rød) og at dette skjer på tilstrekkelig tid (som vil si alt under 0.5 sekunder)

### Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Søk etter sauer»-fanen. Her skal brukeren klikke på flere sauer som allerede ligger i tabellen, og få ut informasjon som ID, navn, vekt, alder, helsetilstand og bredde- og lengdegrad. Denne informasjonen sjekkes opp mot riktig sau i databasen via mysqladmin. I tillegg skal teksten for helsetilstand endres til riktig farge basert på verdi. Tiden det tar fra å velge en sau til output blir fremvist måles for hver sau. Dette testes med System.TimeToMillis() som viser i konsollen hvor lang tid det tok fra en sau ble valgt, til output ble vist.

### Hva slags output er godkjent?

Testen erkjennes godkjent hvis absolutt all informasjon om hver eneste sau er korrekt fremvist og at helsetilstandsteksten er satt til riktig farge basert på verdi. Tiden det tar fra å velge en sau til output blir vist må være < 0.5 sekunder for alle valgt sauer.

### Hvilke situasjoner skal skape brudd?

1. Sauelisen er ikke populert.
2. Valgt sau mangler bredde- og lengdegradskoordinater.
3. Valgt sau mangler enten ID, navn, vekt, alder eller helsetilstand.
4. Responstiden for en hvilken som helst sau er > 0.5 sekunder.
5. Ingen internettilkobling
6. Kobling mot databasen feiler

### Estimert tid

Testen tar omtrent fem minutter.



Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren alltid får opp fullstendig korrekt informasjon om alle sauene som er i systemet. Responstiden her er viktig for brukervennligheten; ingen vil vente lenge på å få ut informasjonen om en sau. Det er også vikte å sjekke at helsetilstandsteksten blir markert rød til riktig tid, så det er svært tydelig.

**Godkjent.**

## Test 08 – Sletting av sau

---

Beskrivelse

Testen skal sikre at brukere har muligheten til å slette sauer.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker åpner opp systemet, fyller inn login-informasjon og logger seg inn til hovedvinduet. Deretter navigerer brukeren seg fram til «Søk etter sauer»-fanen, velger en sau og trykker på «Slett sau». Dette bringer opp et JoptionPane med valgene «Slett sau» og «Avbryt». Vi tester begge alternativene for flere sauer. I tillegg sjekker vi om loggen blir oppdatert ved sletting.

Hva slags output er godkjent?

Dersom sauen forsvinner fra listen i «real-time» (d.v.s. at vi ikke trenger å oppdatere hele tabellen for å få ut alle sauene på nytt), samt at sauen blir borte fra databasen, vil testen godkjennes. I tillegg må det skrives til loggen av valgt sau ble slettet for at testen skal bli erkjent godkjent.

Hvilke situasjoner skal skape brudd?

1. En annen sau blir slettet.
2. Ingen sau blir slettet.
3. Flere sauer enn forventet blir slettet.
4. Responstiden er for dårlig.
5. Ingen internettkobling



## 6. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent ti minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å slette korrekt sau. Det er viktig at det ikke er noe kluss med dette, som f.eks. at feil sau blir slettet. I tillegg er loggføring av sletting av sau viktig.

**Godkjent.**



## Test 09 – Filtrering og søk

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å filtrere, rearrangere og søke etter sauer.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Søk etter sauer»-fanen. Her skal brukeren kunne skrive i «Søk etter sauer»-feltet, og vi ignorerer casekontroll. Sauene som inneholder teksten som brukeren skrev skal da vises, og alle de andre skal skjules – i tillegg skal dette skje umiddelbart, uten noen ventetid. Sauene som vises må da fremdeles inneholde korrekt informasjon dersom man klikker på en av de. Søkefunksjonen fungerer kun på navn – d.v.s. at man ikke kan søke på ID, vekt o.l. I tillegg skal man kunne arrangere sauene i ascending eller descending rangering dersom man dobbelklikker på «ID», «Navn» eller «Vekt».

Hva slags output er godkjent?

I vår simulering har vi følgende sauer: (1) Peter, (2) Per, (3) Per Øyvind, (4) Håvar og (5) Lise. Dersom vi skriver «pet» i søkefeltet, skal kun (1) vises. Skriver vi «pe» skal (1), (2) og (3) vises. Skriver vi «hå» skal (4) vises, og for «Li» skal (5) vises. Hvis alt dette skjer, samt at det er ingen ventetid, vil testen erkjennes godkjent. I tillegg må rearrangering (ved å dobbelklikke på kolonnenavnene) fungere for at testen skal være godkjent.

Hvilke situasjoner skal skape brudd?

1. Søkeordet finnes ikke i noen av sauenes navn.
2. Arrangeringen blir feil ved sortering på ID.
3. Arrangeringen blir feil ved sortering på navn.
4. Arrangeringen blir feil ved sortering på vekt.
5. Alle sauer vises uansett
6. Ingen sauer vises uansett
7. Feil sauer vises
8. Feil informasjon om valgt sau blir vist etter sortering / rearrangering.
9. Ingen internettkobling
10. Kobling mot databasen feiler



Estimert tid

Testen tar omtrent femten minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å søke etter sauer og arrangere sauene i stigende eller synkende rekkefølge for ID, navn og vekt. Dette er en svært viktig funksjon dersom brukeren har et stort antall sauer i systemet sitt, der det ville tatt lang tid å navigere seg fram til ønsket sau.

**Godkjent.**



## Test 10 – Posisjonslogg

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å se på tidligere posisjoner til valgt sau.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Søk etter sauer»-fanen. Her skal brukeren velge en sau og panelet for posisjonslogg skal vises. Posisjonsloggen i seg selv skal kunne inneholde timestamp for når endringen skjedde, og dersom man klikker på en av radene, skal labelene «Latitude» og «Longitude» fylles ut med korrekte lengde- og breddegrader. I tillegg skal brukeren kunne trykke på «Vis på kart» etter å ha markert en rad i posisjonsloggen, og da vil man kunne se posisjonen til sauene på kartet på dette tidspunktet. Dette gjennomføres for flere av sauene og flere av tidspunktene.

Hva slags output er godkjent?

Dersom sauens posisjonsfelter blir oppdatert når en velger en rad i posisjonsloggen, samt at korrekt posisjon settes på kartet, vil testen erkjennes godkjent.

Hvilke situasjoner skal skape brudd?

1. Feil koordinater blir satt i koordinatfeltene.
2. Ingen koordinater blir satt i koordinatfeltene.
3. Ingen tidligere koordinater blir satt i posisjonsloggen (gitt at det faktisk finnes tidligere)
4. Nye koordinater blir lagt til dersom sauene endrer posisjon.
5. Ingen internettkobling
6. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent femten minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å faktisk gå tilbake i tid og se hvor enhver sau var til enhver tid. Dette er viktig for å for eksempel kunne fastslå hva slags mønster sauene beveger seg i, eller muligens hvorfor helsetilstand blir endret når sauene er i et gitt område.

**Godkjent.**





## Test 11 – Endring av sauevekt

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å endre vekten til en gitt sau.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Søk etter sauer»-fanen. Her skal brukeren velge en sau og informasjon om valgt sau skal vises. Brukeren kan så klikke på «Endre»-knappen ved siden av «Vekt», og dermed blir en JOptionPane vist. Brukeren skriver så inn et hel- eller flyttall, og klikker «Lagre». Vekten sjekkes deretter i databasen for å være sikker på at det faktisk ble endret.

Hva slags output er godkjent?

Dersom sauens vekt blir oppdatert i databasen og informasjons-panelet til sauene, blir testen erkjent godkjent. I tillegg skal loggen bli oppdatert for at testen skal være godkjent.

Hvilke situasjoner skal skape brudd?

1. Ikke hel- eller flyttall skrevet inn av brukeren.
2. Vekten er den samme som før
3. Ingen internettkobling
4. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent fem minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å endre vekten til en gitt sau. Dersom vekten er en viktig opplysning for bonden, noe den sannsynligvis er, er det viktig at bonden kan oppdatere vekten til enhver sau.

**Godkjent.**



## Test 12 – Visning av alle sauer

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å vise all sauer en har i systemet på kartet.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og klikker på «Vis sauer»-knappen øverst til høyre. Alle sauer en bruker har registrert skal da vises på ett og samme kart. Dersom brukeren ikke har noen sauer registrert, skal ingen sauer vises. Sauenes posisjoner vil bli sjekket opp i mot koordinatene som ligger i databasen.

Hva slags output er godkjent?

Dersom brukeren får åpnet kartet ved å trykke på «Vis sauer»-knappen, samt at alle sauene har korrekte posisjoner og at responstiden fra en klikker på knappen til alle sauer er vist er < 1.5 sekund, vil testen erkjennes godkjent.

Hvilke situasjoner skal skape brudd?

1. Ingen sauer vises dersom det finnes registrerte sauer hos brukeren.
2. Sauer vises dersom det ikke finnes registrerte sauer hos brukeren.
3. Feil sauer vises.
4. Sauene har ikke korrekte koordinater eller er utenfor visningsvinduet.
5. Responstiden er > 1.5 sekund.
6. Sauenes posisjon kan endres inne i kartvinduet ved å dra.
7. Ingen internettkobling
8. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent fem minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å vise alle sauene en har registrert, på ett og samme kart. Dette er en viktig og brukervennlig funksjon som tillatter brukeren å få en noenlunde oversikt over alle sauene sine på ett sted.

**Godkjent.**



## Test 13 – Sletting av logg

---

### Beskrivelse

Testen skal sikre at brukere har muligheten til å slette loggen sin.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en bruker logger seg inn på systemet og navigerer seg fram til «Logg»-fanen. Her blir brukeren vist all aktivitet som har blitt gjort siden brukeren ble opprettet, gitt at brukeren ikke har slettet noe fra før. Brukeren skal så kunne trykke på knappen «Slett logg», og hele historikken vil da bli slettet.

Hva slags output er godkjent?

Dersom alle rader i loggen blir slettet etter å ha trykket på «Slett logg»-knappen, samt at responstiden for dette er svært lav (tilnærmet umiddelbar), vil testen godkjennes.

Hvilke situasjoner skal skape brudd?

1. Ingen data i loggtabellen.
2. Responstiden er ikke tilnærmet umiddelbar.
3. Ingen internettkobling
4. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent fem minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren har mulighet til å slette alt fra loggpanelet når en selv vil. Applikasjonen har ingen rammer for hvor mye loggdata som kan vises til enhver tid, fordi vi synes det er mer hensiktsmessig for brukeren å kunne se på all tidligere aktivitet. Dersom brukeren mot formodning ikke ønsker å ha tidligere loggføringer vist, er det viktig at han/hun har mulighet til å slette loggen.

**Godkjent.**



## Test 14 – Alarmmelding til brukeren

---

### Beskrivelse

Testen skal sikre at brukere får tilsendt alarmmelding dersom en sau sin helsetilstand er kritisk eller er blitt angrepet.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at en administrator logger seg inn på systemet og huker av for «Aktivere simulering». Dette vil oppdatere helsetilstand og boolsk angrepsvariabel to ganger i minuttet. Vi lar simuleringen fortsette til vi ser at helsetilstanden til en sau er blitt satt til kritisk eller hvis sauene har blitt angrepet.

Hva slags output er godkjent?

Dersom en sau sin helsetilstand er satt til kritisk eller har blitt angrepet etter simuleringen, og brukeren for opp en JwarningPane med hvilken sau som har blitt angrepet eller har kritisk helsetilstand, og mail har blitt sendt til både brukeren og eventuell venn som er satt som reserve, vil testen erkjennes godkjent.

Hvilke situasjoner skal skape brudd?

1. Brukeren får ingen alarmmelding i programmet.
2. Brukeren får ingen e-post tilsendt.
3. Reservebuddy får ingen e-post tilsendt.
4. Status endres ikke i «Valgt sau» panelet.
5. Ingen internettkobling
6. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent ti minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at brukeren får opp en alarmmelding både i programmet og i innboksen til mailadressen sin og reservevennen sin dersom en sau har kritisk helsetilstand eller har blitt angrepet. Dette er en svært viktig del av funksjonaliteten til programmet, da brukere må bli varslet umiddelbart dersom en kritisk endring i helsetilstand eller angrep til enhver sau skjer.



Hvis dette tar for lang tid, mister man mye av hensikten av programmet generelt.

**Godkjent.**

## Test 15 – Maks kapasitet på antall sauer

---

### Beskrivelse

Testen skal sikre at systemet har i det minste en kapasitet på 10 000 sauer, noe som er et av kravspesifikasjonene.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at databasen først tømmes for alle sauer, og deretter blir fylt opp med 10 000 nye sauer v.h.a. en for-løkke. Deretter kjøres test 05 til og med test 14 på nytt.

Hva slags output er godkjent?

Testen erkjennes godkjent dersom brukeren kan logge inn med en responstid < 5 sekunder, visning av alle sauer skjer innen 10 sekunder, samt resten av testene 05 til og med 14 gjennomføres innenfor rimelighetens grenser når det gjelder responstid.

Hvilke situasjoner skal skape brudd?

1. Ingen sauer lastes inn fra databasen.
2. Ikke alle sauer lastes inn fra databasen.
3. Sauene blir lastet inn fra databasen, men det tar for lang tid.
4. Test 05-14 gjennomføres ikke.
5. Test 05-14 gjennomføres, men tar altfor lang tid.
6. Ingen internettkobling
7. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent to timer.

Hvorfor er denne testen viktig?

Denne testen sikrer at systemet kan håndtere 10 000 sauer uten at det går altfor mye ut på brukervennligheten til programmet. Systemet skal kunne håndtere 10 000 sauer, og hvis responstiden til diverse funksjoner i programmet blir altfor høy, vil programmet i seg selv være svært lite brukervennlig. Vi er avhengig her av at responstiden holdes lav, selv om antall sauer er høyt.

**Godkjent.**

## Test 16 – Maks kapasitet på antall brukere

---

Beskrivelse

Testen skal sikre at systemet har i det minste en kapasitet på 200 brukere samtidig, noe som er et av kravspesifikasjonene.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at databasen først tømmes for alle brukere og sauer, og deretter blir fylt opp med 200 nye brukere v.h.a. en for-løkke. Hver av disse brukerne vil i tillegg ha 50 sauer, slik at totalt antall sauer blir 10 000. Deretter kjøres test 05 til og med test 14 på nytt.

Hva slags output er godkjent?

Testen erkjennes godkjent dersom brukeren kan logge inn med en responstid < 5 sekunder, visning av alle sauer skjer innen 10 sekunder, samt resten av testene 01 til og med 14 gjennomføres innenfor rimelighetens grenser når det gjelder responstid.

Hvilke situasjoner skal skape brudd?

1. Programmet starter ikke.
2. Det tar for lang tid å starte programmet.
3. Ingen sauer lastes inn fra databasen.
4. Ikke alle sauer lastes inn fra databasen.
5. Sauene blir lastet inn fra databasen, men det tar for lang tid.
6. Test 01-14 gjennomføres ikke.



7. Test 01-14 gjennomføres, men tar altfor lang tid.
8. Ingen internettkobling
9. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent to timer og femten minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at systemet kan håndtere 200 brukere samtidig med til sammen 10 000 sauer uten at det går altfor mye ut på brukervennligheten til programmet. Systemet skal kunne håndtere 10 000 sauer, og hvis responstiden til diverse funksjoner i programmet blir altfor høy, vil programmet i seg selv være svært lite brukervennlig. Vi er avhengig her av at responstiden holdes lav, selv om antall sauer er høyt.

**Godkjent.**



## Test 17 – Dimensjonering: forespørsler og rapportering

---

### Beskrivelse

Testen skal sikre sauene kan rapportere sin lokasjon minst tre ganger i døgnet og at bøndene kan forespørre lokaliseringinformasjon minst to ganger i døgnet.

Hvordan skal testen gjennomføres?

Testen gjennomføres ved at programmet vil være oppe et helt døgn, og simuleringsfrekvensen (som tidligere var en gang per 30. sekund) vil nå endres til én gang hver 8. time. Det vil si at hver 8. time vil noen sauer bli oppdatert. Deretter vil en bruker forespørre lokasjonsinformasjon én gang om morgningen og én gang om kvelden, med til sammen 12 timer mellom hver.

Hva slags output er godkjent?

Testen erkjennes godkjent dersom sauene rapporterer om sin lokasjonsinformasjon minst tre ganger i løpet av de 24 timene, og bonden hadde mulighet til å sende forespørsler om lokasjonsinformasjon både om morgningen og om kvelden.

Hvilke situasjoner skal skape brudd?

1. Sending av GPSData slutter å fungere.
2. Mottagelse av GPSData slutter å fungere.
3. Ingen internettkobling
4. Kobling mot databasen feiler

Estimert tid

Testen tar omtrent 24 timer.

Hvorfor er denne testen viktig?

Denne testen sikrer at systemet faktisk vil sende ut informasjon om sauene minst tre ganger i døgnet og at brukeren har mulighet til å sende forespørsel minst to ganger i døgnet. Merk at det kun er en simulator som her håndterer forespørsler og rapporteringer, ikke en reell server.

**Godkjent.**





## Test 18 – Utilgjengelighet, oppetid & operativitet

---

### Beskrivelse

For utilgjengelighet i denne testen, må vi øke antall timer som skal være tillatt fra 3 til 5 timer. Dette er fordi den faktoren av utilgjengelighet som er høyest sannsynlig er at NTNU sine sentrale IT-systemer er nede for vedlikehold, og ut i fra vår erfaring vil dette være maksimalt 5 timer. Det er viktig å merke seg at dersom NTNU sine servere ikke hadde blitt brukt i vår løsning, så ville utilgjengeligheten vært utviklernes ansvar gjennom hele programmets livsløp. Testene sikrer at programmet er tilgjengelig, kjørende og operativt så mye som overhodet mulig.

### Hvordan skal testen gjennomføres?

Testen gjennomføres simpelthen ved å se på logger for oppetid fra MySQL-serveren vi bruker. Vi sjekker loggene fra da vi først startet å bruke mySqlAdmin og helt fram til enden av utviklingsfasen. Loggene vil da vise når systemet har vært nede. For tilgjengelighet må vi da bare sjekke at tiden som systemet har vært nede per gang er  $< 5$  timer (som er vedlikeholdstiden NTNU forholder seg til). For oppetid sjekker vi da hvor mange timer systemet har vært nede i forhold til hvor mange timer systemet har vært oppe. Operativitet ble sjekket i test 17.

### Hva slags output er godkjent?

Testen erkjennes godkjent dersom alle nedetidene til systemet holder seg under 5 timer og at oppetiden til programmet holder seg over 95 %. I tillegg ble operativitet godkjent under test 17.

### Hvilke situasjoner skal skape brudd?

1. Overbelastning av database (høyest usannsynlig)
2. Back-end oppdateringer fungerer ikke.
3. Ingen internettkobling
4. Kobling mot databasen feiler

### Estimert tid

Testen tar omtrent 24 timer. Dette er fordi operativitet må sjekkes. Test for oppetid og tilgjengelighet har vært kjørende siden starten av utviklingen, så å sjekke output vil bare ta noen få minutter.

Hvorfor er denne testen viktig?

Denne testen sikrer at systemet har en oppetid og tilgjengelighet som er så god som overhode mulig, samtidig som at programmet er operativt hele døgnet. Back-end oppdateringer (som f.eks. simuleringer) må fungere som normalt i løpet av hele testperioden.

**Godkjent – i løpet av 2000 timer oppetid, har NTNU sine IT-systemer vært nede til sammen 15 timer. Dette tilsvarer 99.25% oppetid. Se <http://www.itea.ntnu.no/drift/nyheter/> for mer informasjon om vedlikehold på NTNU sine IT-systemer.**

