SpareBank 1

Vedlegg A

Kravspesifikasjon

Dette dokumentet beskriver krav til applikasjonen som skal designes i prosjektet *Nettverksbasert applikasjonsovervåking*. Det beskrives her både krav til selve applikasjonen og hva som forventes av de forskjellige lagene i denne, men også krav til hvordan systemet skal designes og dokumenteres.

1. PRESENTASJON

PROSJEKTTITTEL Nettverksbasert applikasjonsovervåking

APPLIKASJONSTITTEL PySniff

OPPGAVE Utvikle et rammeverk for overvåking av nettverkstrafikk mellom applikasjoner i et nettverk.

1.1 MEDLEMMER I PROSJEKTET

Anders Struksnæs
Lars Haugan
Ole Henrik Paulsen
Tim Sæterøy

1.2 OPPDRAGSGIVER

SpareBank 1 Gruppen AS
KMIT & Innkjøp
Hammersborggata 2
0191 Oslo
origo@sparebank1.no

1.3 KONTAKTPERSON

SpareBank 1 Gruppen AS
Martin Jensen
Martin.Jensen@sparebank1.no
40218026
Avdelingsleder KMIT & Innkjøp, Drift Alliansen

1.4 VEILEDER

Torunn Gjester
Seksjon for Informasjonsteknologi
Høgskolen i Oslo og Akershus

1.5 BAKGRUNN

Origo og Drift Alliansen er avdelinger for IT brukerstøtte og drift i SpareBank 1 Gruppen. Avdelingene leverer tjenester til SpareBank 1 Gruppen AS med datterselskap og bankene i SpareBank 1 Alliansen. Som en del av oppgavene til Origo og Drift Alliansen jobbes det med monitorering og overvåking av IT-tjenestene innen SpareBank 1 Gruppen AS.
Hensikten med prosjektet er utviklingen av et nytt overvåkingsverktøy som kan benyttes for overvåking av applikasjoner som benyttes i SpareBank 1 Gruppen AS. Applikasjonen vil være et supplement til dagens driftsovervåking og innføre begrepet trending i overvåkingen. Data skal trendes i form av å finne en grense på hva som er normalt og hva som ikke er normalt i bestemt tidsperiode.

2. FORORD

Denne kravspesifikasjonen er beregnet for utviklere, oppdragsgiver og andre som er med i prosessen med utvikling av applikasjonen. Kravspesifikasjonen definerer de krav som skal være til prosjektet og applikasjonen, og er ment som et kontinuerlig verifiseringsdokument for å sikre god utvikling i prosjektet.

Innholdsfortegnelse

3. SYSTEMKRAV

3.1 FUNKSJONSKRAV

3.1.1 KRAV TIL SENSOR

3.1.1.1 FUNKSJONELLE KRAV

- Sensor skal være uavhengig av resten av applikasjonen.
- Hvis det skal brukes flere sensorer, skal disse være uavhengig av hverandre.
- Sensoren skal sniffe TCP, men det skal være mulig å legge til andre protokoller fra transportlaget i OSI-modellen
- Sensoren skal ikke forstyrre eller «stjele» kapasiteten eller ressursene til andre applikasjoner på serveren den står på.
- Sensoren skal bare videresende pakker som er definert i filteret.

3.1.1.2 IKKE FUNKSJONELLE KRAV

- Sensor skal ha en oppetid på minimum 96,0% første året etter prosjektet er avsluttet.
- Sensor skal ved bruk på applikasjonsserver ikke benytte mer enn en prosessorkjerne.

3.1.2 KRAV TIL DATABASE

3.1.2.1 FUNKSJONELLE KRAV

- Databaselag 1 skal inneholde all data sensoren sender til databasen.
- Databaselag 1 skal være lagret i RAM.
- Databaselag 2 skal kun inneholde aggregert data som hentes i fra databaselag 1.
- Det skal være et logisk skille mellom databaselag 1 og databaselag 2 i form av egen databaseprosesser kjørende.
- Det er kun Webservice og Core som skal ha leserettigheter til databaselagene.
- Det er kun Sensoren og Core som skal ha skriverettigheter til databaselag 1.
- Det er kun Core som skal ha skriverettigheter til databaselag 2.

3.1.2.2 IKKE FUNKSJONELLE KRAV

- Databaseserveren skal ha en oppetid på minst 90% over ett år i fra prosjektslutt.
- Data i databaselag 1 skal ikke være lagret lengere enn maksimalt 14 døgn.
- Data i databaselag 1 skal minst være lagret i databaselag 1 i ett døgn.
- Data i databaselag 2 skal ikke være lagret i lengere tid enn 27 måneder.
- Data i databaselag 2 skal minst være lagret i databaselag 2 i 25 måneder.
- Gjennomsnittet på spørre-kall skal være under 2 sekunder ved mindre en 4 klienter tilkoblet frontend.
- Spørring på real-time data skal maksimalt ha en forsinkelse på 5 sekunder i fra spørringen ankommer databasen til databasen svarer.

3.1.3 KRAV TIL CORE

3.1.3.1 FUNKSJONELLE KRAV

- Legge til rette for aggregering av applikasjonsdata.
- Et tidsintervall skal styre når aggregeringen skal utføres.
- Core skal slette eventuell gammel data i databaselag 1, slik at databasen ikke blir full.
- Aggregeringen skal bruke et felles pluginbibliotek, og legge til rette for utvidelser i form av flere plugins.
- Core skal kjøre i bakgrunnen og ikke påvirke andre prosesser.
- Skal ikke være tilknyttet noen brukerøkt styrt av en bruker.
- Funksjonalitet i Core skal være konfigurerbar.
- Core skal logge handlinger slik at det er mulig å vite når og hva som utføres. Feilsituasjoner skal også logges, brukt for feilsøking.

3.1.4 KRAV TIL WEBSERVICE

3.1.4.1 FUNKSJONELLE KRAV

- Webservice skal gjøre kall mot databaselag 1 og databaselag 2.
- Feilsituasjoner og statusinformasjon skal logges til fil

    - Et kall mot webservice skal resultere i en JSON-streng
- Webservice skal benytte pluginer for ulike systemer/protokoller
    - Hver enkelt plugin skal kun være beregnet på en spesifikk protokoll/system
    - Disse skal ligge i en egen mappe, lib/plugins
- Variabler som IP, port, osv skal ligge i konfigurasjonsfiler.

3.1.4.2 IKKE FUNKSJONELLE KRAV

- Kall mot database skal ikke ta over to sekunder

3.1.5 KRAV TIL FRONTEND

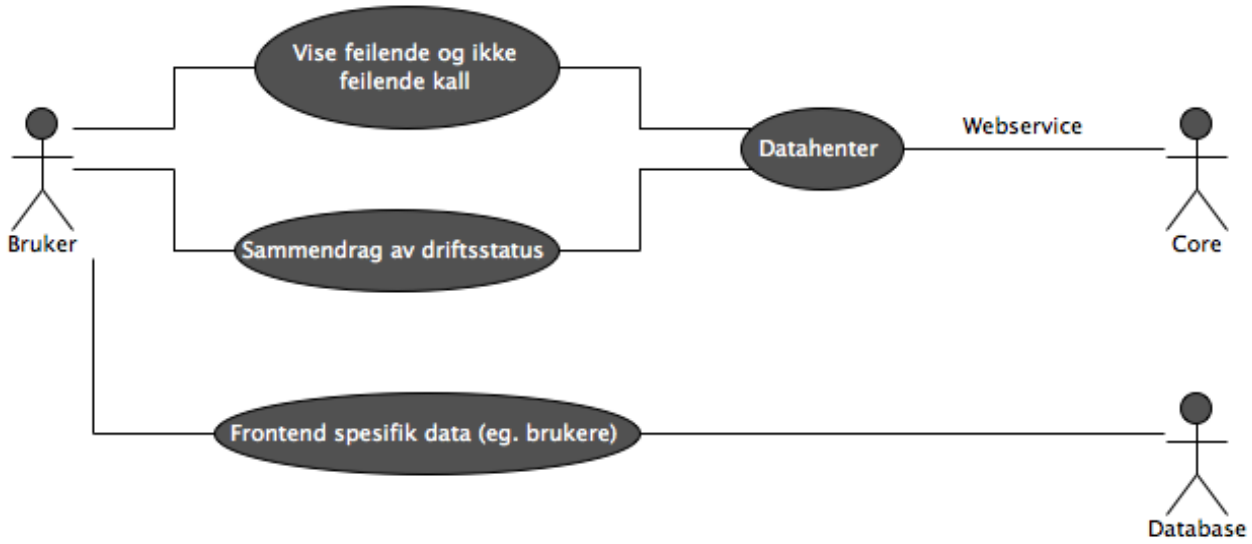
3.1.5.1 FUNKSJONELLE KRAV

- Støtte nye nettlesere, som siste versjon av Chrome, Firefox og Internet Exolorer.
- Presentere data fra webservicen som grafer.
- Benytte data fra webservice

3.1.5.2 IKKE FUNKSJONELLE KRAV

- Systemet skal kunne benyttes på en overvåkingsskjerm for å gjøre det mulig å respondere på driftshendelser.
- Skal benyttes javascript for å gjøre visning og bruk av frontend så responsiv som mulig.
- Innlasting av nettsidene skal ta mindre enn 5 sekunder.
- Innlasting av nettsidene skal helst lastes på mindre enn 2 sekunder.
- Data kan sendes asynkront for å gjøre siden mer responsiv.
- Data skal oppdateres minst hvert minutt.

3.1.5.3 MODELL



3.2 TEKNISKE KRAV

- Systemet skal kunne kjøre på 64-bits versjon av Linux-distribusjonene som benyttes i produksjonsmiljøet til SpareBank 1 (CentOS, RedHat).
- Systemet skal utvikles og kjøres på Python versjon 2.7.3

3.3 KRAV TIL DATALAGRING

3.3.1 FUNKSJONELLE KRAV

- Gruppen skal bruke Dropbox, Google Docs og Git
- Alle på prosjektgruppen skal ha tilgang til overnevnte lagringsplasser

3.3.2 IKKE FUNKSJONELLE KRAV

- Det skal gjennomføres backup av Dropbox, Google Docs minst 1 gang per uke. (For Git se punkt eget punkt)
- Dokumenter og innhold i overnevnte tjenester skal ikke slettes før minst ett halvt år etter prosjektslutt.

4. KRAV TIL DESIGN

4.1.1 FUNKSJONELLE KRAV

- Nettsiden skal benytte skrifttype (font) som er beregnet for lesing på datamaskin.
- Skrifttype brukt på nettsiden skal være åpen og tilgjengelig for bruk på Windows, Mac OS X og Linux.

4.1.2 IKKE FUNKSJONELLE KRAV

- Det bør være lett å få oversikt
- Det skal være et felles grensesnitt på alle sidene.
- Det skal være gode kontraster i fargebruken.
- Nettsiden skal være kompatibel med alle moderne nettlesere som Chrome, Firefox, Opera og Internet Explorer med siste versjon.

5. KRAV TIL KODE

5.1 KODESTANDARD

- Koden skal benytte UTF-8 tekst-encoding.
- Koden skal være objektorientert.
- Feilmeldinger skal alltid håndteres med logging av feilsituasjoner til fil.
- Tekst skal aldri skrives stil stdout eller stderr, men til logg.

5.2 VARIABLER OG FUNKSJONER

5.2.1 FUNKSJONELT

- Flere ord i variabelnavn skal settes sammen med bruk av underlinje (_).
- Flere ord i funksjoner skal settes sammen med bruk av underlinje (_).

5.2.2 IKKE FUNKSJONELLE KRAV

- Variabelnavn skal være logisk i sammenhengen.

5.3 KOMMENTERING

- Funksjoner skal alltid kommenteres med beskrivende tittel og returverdi.
- Variabler trenger ikke å kommenteres.
- Klasser skal kommenteres med hva de skal brukes.

5.4 SPRÅK

Gjennomgående språk i koden skal være engelsk, dette for å lette arbeidet med å sette seg inn i koden for andre utviklere på et senere tidspunkt, samt at norske bokstaver (æ, ø og å) har vist seg å skape trøbbel i tidligere prosjekter.

6. KRAV TIL DOKUMENTASJON

6.1 OBLIGATORISK DOKUMENTASJON

Følgende obligatoriske dokumenter skal utformes i løpet av prosjektets periode:

6.1.1 STYRINGSDOKUMENTER

- Prosjektskisse
- Forprosjektrapport
- Arbeids- og fremdriftsplan
- Kravspesifikasjon

6.1.2 SLUTTDOKUMENTASJON

- Kravspesifikasjon
- Prosessdokumentasjon
- Produktdokumentasjon
- Installasjonsdokumentasjon
- Brukerdokumentasjon

6.2 GENERELLE KRAV TIL DOKUMENTASJON

- Dokumentasjon skal følge standard dokumentmal fra SpareBank 1.
- Dokumentene skal være skrevet for visning på papir.

6.3 VERSJONSKONTROLL

6.3.1 FUNKSJONELLE KRAV

- Alle på prosjektgruppen skal kunne skrive, klone og hente til/fra git-repositoriet.
- Ingen skal jobbe direkte på Master branch.
- Master branch skal bli brukt til produksjonsetting
- Alle som skriver til git-repositoriet skal kommentere dette på norsk.

6.3.2 IKKE FUNKSJONELLE KRAV

- Git-repositoriet skal være privat, og kun prosjektmedlemmer skal ha tilgang.
- Det skal tas backup av git-repositoriet minst én gang per dag.
- Git skal være tilgjengelig minst 99% av prosjektperioden.

7. UTVIDELSER

- Administrasjonsmuligheter fra frontend.
- Illustrering av trender og data som grafer på frontend.
- Tilgangskontroll for visning av frontend.
- Støtte for flere protokoller

8. FREMMEDORD OG DEFINISJONER

Aggregering	- Kombinere eller slå sammen data. F.eks minke data over lengre tidsperioder.
Trending	- Det normalet i en gitt tidsperiode.
Sensor	- En applikasjon som leser av nettverkstrafikk eller eksempelvis tekstfiler.