Kravspecifikation

MQTT Broker til Linux embedded

Indhold

[1. Introduktion 3](#_Toc110416940)

[1.1 Formål 3](#_Toc110416941)

[1.2 Scope 3](#_Toc110416942)

[1.3 Definitioner, akronymer og forkortelser 3](#_Toc110416943)

[1.4 Referencer 3](#_Toc110416944)

[1.5 Overblik 3](#_Toc110416945)

[2. Funktionalitet 4](#_Toc110416946)

[Connect 4](#_Toc110416947)

[Disconnect 4](#_Toc110416948)

[Create topic (Publish) 4](#_Toc110416949)

[Subscribe 5](#_Toc110416950)

[Ping 5](#_Toc110416951)

[3. Usability 5](#_Toc110416952)

[3.1 IOT enhed 5](#_Toc110416953)

[3.2 Disconnect ved fejl 5](#_Toc110416954)

[3.3 De-facto standard/version 5](#_Toc110416955)

[4. Reliability 6](#_Toc110416956)

[4.1 Tilgængelighed 6](#_Toc110416957)

[4.2 Gendannelse 6](#_Toc110416958)

[4.3 Frekvens af fejl 6](#_Toc110416959)

[5. Performance 6](#_Toc110416960)

[5.1 Svartid 6](#_Toc110416961)

[5.2 Throughput 6](#_Toc110416962)

[5.3 Opstarts- og nedlukningstid 6](#_Toc110416963)

[6. Supportability 6](#_Toc110416964)

[6.1 Kode dokumentation 6](#_Toc110416965)

[6.2 Nødvendig dokumentation 6](#_Toc110416966)

[6.3 Kodesprog og standarder 7](#_Toc110416967)

[7. Design Constraints 7](#_Toc110416968)

[7.1 Implementeringssprog 7](#_Toc110416969)

[8. Interfaces 7](#_Toc110416970)

[8.1 Hardware Interfaces 7](#_Toc110416971)

[8.2 Kommunikation Interfaces 7](#_Toc110416972)

# Introduktion

Dette dokument indeholder alle kendte krav, både funktionelle og ikke funktionelle krav, som er stillet til en MQTT broker til Rasperry 4.

I skal bygge jeres egen MQTT broker efter de specifikationer som OASIS inc. Har stillet til rådighed i version 3.1.1 jf : http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

MQTT brokeren skal overholde alle de outlines som er defineret i specifikationen, således at brokeren f.eks skal afbryde forbindelsen til en given klient hvis username/password er forkert etc. Outlines er i specifikationen underlined med gult.

## Formål

Formålet med denne kravspecifikation er at definere de krav, der måtte være til det nye system. Det forventes at kravene løbende testes, og at der løbende skrives og vedligeholdes nødvendig dokumentation. Endvidere forventes det, at der tænkes over og implementeres en leverancemodel til at sikre at der kommer et brugbart produkt ud af det. Om der benyttes Scrum eller Prince2 er helt op til de enkelte grupper. EfterFølgende dokumentation, herunder denne FURBS skal løbende holdes opdateret og justeret efter aftale med faglærerne.

## Scope

Omfanget af dette dokument begrænser sig til udelukkende at omhandle brokerdelen. Der er således ikke defineret nogen form for klienter, hvorfor grundig test af brokeren forventes løbende.

## Definitioner, akronymer og forkortelser

I dette afsnit angives en ordforklaring /orddefinition på de gængse termer i dette document.

**MQTT: Message Queue Telemetry Transport**

**Broker: det samme som server**

**Server: Den enhed som alle publishers/subscribers forbinder til for at afsende/modtage data.**

**Publisher: En given enhed som ønsker at oprette et topic og efterfølgende levere data dertil.**

**Subscriber: En eller flere enheder som ønsker at modtage data fra et given topic.**

**Topic: Et given emne som en eller flere subscribers kan modtage data via.**

**QoS: Quality Of Service. Kan angives når en IOT enhed forbinder til en given MQTT broker.**

## Referencer

Definitionen for en MQTT broker findes her:   
http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/os/mqtt-v3.1.1-os.html

## Overblik

Der angives i dette dokument først de funktionelle krav til systemet. Disse er angivet i use-case format og findes enten som fully-dressed eller i en brief udgave. Det forventes at disse use-case udbygges undervejs og leveres tilbage ved afslutning af hvert sprint, eller senest 3 dage før afslutning af projektet, så der kan foretages en user-acceptance test.

# Funktionalitet

I denne sektion er nævnt alle de kendte funktionelle krav til systemet. Alle krav defineres i use-case format i enten en brief eller fully dressed udgave.

|  |  |
| --- | --- |
| **Use-case** | Connect |
| **Id** | 1 |
| **Version** | 1.0 |
| **Aktør(er)** | IOT enhed |
| **Trigger** | En IOT enhed ønsker at forbinde til brokeren |
| **Pre-condition** | IOT enheden kender brokerens adresse og credentials |
| **Post-condition** | Brokeren har afsendt en CONNACT og IOT enheden er forbundet |
| **Beskrivelse** | En IOT enhed anfører korrekt adresse og credentials |
| **Normalt forløb** | 1. IOT enheden anfører adresse og credentials 2. Systemet undersøger om credentials er korrekte 3. IOT enheden er forbundet |
| **Alternativt forløb** | * 1. IOT enheden connecter med Will Retain   2. IOT enhed connecter med Will QoS   3. IOT enheden connecter med Clean Session   4. IOT enheden forbinder uden username/password   5. IOT enheden anfører forkert username/password |
| **Udvidelsesmuligheder** | Secure Connection |

|  |  |
| --- | --- |
| **Use-case** | Disconnect |
| **Id** | 2 |
| **Version** | 1.0 |
| **Aktør(er)** |  |
| **Trigger** | En IOT enhed ønsker at disconnecte |
| **Pre-condition** |  |
| **Post-condition** | IOT enheden er connected |
| **Beskrivelse** | En IOT enhed skal altid have mulighed for at disconnecte på ethvert givet tidsppunkt jf. dokumentationen anført under Referencer |

|  |  |
| --- | --- |
| **Use-case** | Create topic (Publish) |
| **Id** | 3 |
| **Version** | 1.0 |
| **Aktør(er)** | IOT enhed |
| **Trigger** | En IOT enhed ønsker at oprette et topic på brokeren |
| **Pre-condition** |  |
| **Post-condition** | Topic er oprettet og en anden IOT enhed kan subscribe til det |
| **Beskrivelse** | En vigtig del af publish/subscribe paradigmet er at der kan oprettes nye topics, som kan subscribes til. Det skal være muligt for en IOT enhed at oprette et topic på en given sti. Som f.eks: alltopics/mytopic, hvor en subscriber kan nøjes med at subscribe til alltopics og også få data fra mytopic. Se dokumentationen for yderligere information. |
| **Normalt forløb** | IOT enheden ønsker at oprette et topic med QoS status 0   1. Brokeren afsender en PUBACK til IOT enheden |
| **Alternativt forløb** | 1. IOT enheden ønsker at oprette et topic med QoS status 1 2. IOT enheden ønsker at oprette et topic med QoS status 2 |
|  | 1. IOT enheden ønsker at oprette et topic med Retain = true 2. IOT enheden ønsker at oprette et topic med Dublicate = true |

|  |  |
| --- | --- |
| **Use-case** | Subscribe |
| **Id** | 4 |
| **Version** | 1.0 |
| **Aktør(er)** | IOT Enhed |
| **Trigger** | En IOT enhed ønsker at subscribe til et topic |
| **Pre-condition** | IOT enehden er forbundet til brokeren |
| **Post-condition** | IOT enheden har subscribet til et given topic i en given gruppe |
| **Beskrivelse** | En IOT enhed skal kunne subscribe til et given topic i en given gruppe, og kunne modtage data på det topic fra publisheren |

|  |  |
| --- | --- |
| **Use-case** | Ping |
| **Id** | 5 |
| **Version** | 1.0 |
| **Aktør(er)** | IOT enhed |
| **Trigger** | En IOT enhed ønsker at verificere at brokeren er alive |
| **Pre-condition** | Brokeren har modtaget en PINGREQ fra IOT enheden |
| **Post-condition** | Brokeren har afsendt en PINGRESP til IOT enheden |
| **Beskrivelse** | Når den sidste spiller forlader rummet skal rummet nedlægges |

# Usability

Systemet skal opholde specifikationen som anført under referencer.

## IOT enhed

En given IOT må ikke opleve at det ikke er muligt at connete, disconnecte, publish, subscribe og pinge brokeren. Det skal være muligt at benytte enhvert industri-standard IOT enhed til at udføre de forventede kommander med, herunder temperaturmålere, gauges etc.

## Disconnect ved fejl

Som det er anført i dokumentationen under referencer skal brokeren disconnecte en given IOT enhed ved eventuelle hændelser outlined med gult. Hændelser der ikke er outlined i gult er valgfrie med hensyn til fortsat serice af en given IOT enhed.

## De-facto standard/version

Brokeren skal opføre sig som enhver anden MQTT broker på markeret. Det er forventet at brokeren kun understøtter MQTT level 4 som anført i dokumentationen.

# Reliability

## Tilgængelighed

Systemets oppetid er 99,99%, servicevindue mandag/onsdag/fredag imellem klokken 00.00 – 03.00. Der refereres til standard Mean Time Between Failures (MTBF) på alt hardware.

## Gendannelse

Fuld backup at systemet og dets data Mean Time to Repair (MTTR) kan genskabes på minimum 3 timer.

## Frekvens af fejl

Fejlrate gennemsnit: 

* 15 - 50 fejl pr. KLOC af leveret kode, under udvikling
* 10-20 fejl pr. KLOC under interne test
* 0,5 fejl pr. KLOC i produktion

# Performance

## Svartid

Ethvert interface mellem IOT enheder, skal have en maksimumsvartid på 2 sekunder.

## Throughput

Systemet skal kunne håndtere minimum 150 samtidige transaktioner

## Opstarts- og nedlukningstid

5 minutter.

# Supportability

## Kode dokumentation

Der skal udarbejdes kodekommentar for alle metoder og klasser og der skal udarbejdes en linux manual (man) til brokeren

## Nødvendig dokumentation

Der skal udarbejdes nødvendig dokumentation for alle projektets leverancer. Således at hvis brokeren skrives objektorienteret, skal der udarbejdes klassediagrammer, statediagrammer, flowcharts SD, SSD diagrammer samt nødvendige UC-diagrammer. Skrives brokeren prodecurelt skal der ligeledes laves nødvendig dokumentation for at have hele systemet dokumenteret.

Projektets fremdift og håndtering skal ligeledes dokumenteres. Det forventes at der skrives og vedligeholdes tidsplaner, epics, backlogs, afholdes grooming alt efter valg af leverancemodel. Derudover skal der laves en risikomatrix, RACI uanset leverancemodel.

Alt dokumentation skal løbende vedligeholdes og kunne fremvises efter behov.

For sidste leverance leveres et deployment diagram.

## Kodesprog og standarder

Al kode skal skrives i maskinnært sprog. Der må derfor ikke vælges sprog som kræver en VM for at køre. Acceptable valg af sprog er følgende

1. C/C++
2. RUST

Tag en dialog med din faglærere, hvis du ønsker at bruge et andet sprog end de her anførte.

Det forventes at koden skrives efter de behørige standarder for sproget. Du kan google code conventions for det valgte sprog, hvis du er tvivl om standarden.

# Design Constraints

## Implementeringssprog

Brokeren skal skrives engelsksproget. Det er således ikke tilladt at bruge danske variabelnavne!

# Interfaces

## Hardware Interfaces

Brokeren skal kunne afvikles på et ARM device såsom Rasperry Pi 4

## Kommunikation Interfaces

MQTT protokollen