

# Visjonsdokument

Unity integrasjon med SensMax TAC-B

IDATT2900-022

Erik Borgeteien Hansen,

Oda Alida Fønstelien Hjelljord

## Revisjonslogg

Dato	Versjon	Beskrivelse	Forfatter
01.02.2023	0.1	Begynne å fylle ut kapittel 1 og 2	Erik, Alida
02.02.2023	0.2	Kapittel 3 og 4, startet arbeid på 5 og 6	Alida, Erik
15.02.2023	0.3	Forsatte arbeid på 5 og 6, Gjennomgang, la til teoriseksjon	Alida
16.02.2023	0.4	Fullførte underliggende teknologier kapittel	Alida
15.05.2023	1.0	Stave- og grammatikkontroll	Alida

# Innhold

<b>1</b>	<b>Innledning</b>	<b>1</b>
<b>2</b>	<b>Underliggende teknologier</b>	<b>2</b>
2.1	Unity . . . . .	2
2.2	MQTT . . . . .	2
<b>3</b>	<b>Sammendrag problem og produkt</b>	<b>3</b>
3.1	Problemsammendrag . . . . .	3
3.2	Produktsammendrag . . . . .	3
<b>4</b>	<b>Overordnet beskrivelse av interessenter og brukere</b>	<b>4</b>
4.1	Oppsummering interessenter . . . . .	4
4.2	Oppsummering brukere . . . . .	4
4.3	Brukermiljøet . . . . .	5
4.4	Sammendrag av brukernes behov . . . . .	6
4.5	Alternativer til vårt produkt . . . . .	6
<b>5</b>	<b>Produktoversikt</b>	<b>7</b>
5.1	Produktets rolle i brukermiljøet . . . . .	7
5.2	Forutsetninger og avhengigheter . . . . .	7
<b>6</b>	<b>Produktets funksjonelle egenskaper</b>	<b>8</b>
<b>7</b>	<b>Ikke-funksjonelle egenskaper og andre krav</b>	<b>9</b>

# 1 Innledning

I dette dokumentet vil det fremlegges visjon for en “SensMax Unity pakke” som gruppen skal lage i samarbeid med Able Magic AS (Ablemagic). Gruppen skal utvikle en pakke til Unity som tillater kommunikasjon mellom Unity og SensMax TAC-B 3D-WP mennesketellende radarsensor (sensoren), dette vil forenkle bruk av sensoren i spill og interaktive utstillinger som Ablemagic lager for sine kunder.

Siden pakken skal distribueres som åpen kildekode vil vårt produkt ikke bare bidra i Ablemagic sin utvikling, men også i alle prosjekter som jobber med denne sensoren. Det som blir utviklet i dette prosjektet er mellomvare mellom spillmotoren Unity og webgrensesnittet til SensMax TAC-B 3D-WP.

## 2 Underliggende teknologier

### 2.1 Unity

Unity er spillmotor som gir utviklere verktøy for å skaper interaktive 2D og 3D spill. Unity ble utviklet av Unity Technologies for å tilgjengeliggjøre spillutvikling for flere utviklere, spesielt uavhengige utviklere [1]. En spillmotor er et programvarerammeverk fortrinnsvis laget for å utvikle spill. En spillmotor vil inneholde relevante biblioteker og støtteprogrammer for å bidra til spillutvikling [2].

### 2.2 MQTT

Sensoren som brukes i dette prosjektet benytter MQTT-protokollen som kommunikasjonsmetode. MQTT er en publisere/abonnere-protokoll krever minimalt med plass og båndbredde for å koble IoT-enheter. Denne typen arkitektur kobler klientene fra hverandre for å muliggjøre skalerbare løsninger uten avhengighet mellom dataprodusenter og datakonsumenter [3].

MQTT sitt publiser/abonner-mønster (pub/sub) er et alternativ til den tradisjonelle klient/server-arkitekturen. I motsetning til klient/server-arkitekturen er det i MQTT en server som håndterer kommunikasjonen mellom flere publiserende og abonnerende klienter [4].

### 3 Sammendrag problem og produkt

#### 3.1 Problemsammendrag

Problem med	mangel på enkel måte å få kommunikasjon mellom Unity og sensoren
berører	utviklere i Unity som vil benytte sensoren
som resultatet av dette	må utviklere eventuelt utvikle kommunikasjon mellom sensoren og Unity hver gang de skal benytte sensoren
en vellykket løsning vil	gjøre det enkelt å inkorporere sensoren i Unity-prosjekter.

Tabell 1: Problemsammendrag

#### 3.2 Produktsammendrag

For	Ablemagic
som	har behov for et kodebibliotek som utvider funksjonaliteten til spillmotoren Unity, også kalt en Unity-pakke.
Produktet	er mellomvare mellom Unity og sensorens webgrensesnitt
som	åpner for kommunikasjon mellom sensoren og Unity, slik at denne sensorens input kan brukes i Unity.
I motsetning til	dagens system der det ikke finnes en enkel pakkeløsning
er vårt produkt	en pakke som gjør utviklingsprosessen raskere og enklere dersom du vil benytte sensoren i Unity.

Tabell 2: Produktsammendrag

## 4 Overordnet beskrivelse av interessenter og brukere

### 4.1 Oppsummering interessenter

Navn	Utdypende beskrivelse	Rolle under utviklingen
Erik Borgeteien Hansen	Student NTNU	Utvikler, kontakt med kunden, dokumentasjon
Oda Alida Fønstelien Hjelljord	Student NTNU	Utvikler, kontakt med kunden, dokumentasjon
Able Magic AS	Ablemagic er kunden som utlyste denne oppgaven. De er et Trondheim basert spillfirma som fokuserer på å produsere interaktive fortellinger og installasjoner for sine kunder [5]. Ablemagic bruker sensorer i sine installasjoner og spill	Bidrar med arbeidsplass og verksted. Stiller med brukere til testing av produktet.
NTNU	Norges teknisk-naturvitenskapelige universitet. Utlyste oppgaven som bachelor oppgave og er arbeidsgiveren til studentene	Bidrar med veileder som bistår med kommentarer på innleveringer og rådgivning i prosjektperioden

Tabell 3: Interessenter

### 4.2 Oppsummering brukere

Navn	Utdypende beskrivelse	Rolle under utviklingen	Representert av
Spillutviklere	Spillutviklere som jobber i Unity og ønsker å benytte SensMax TAC-B 3D-WP sensoren i sitt arbeid	Ingen	Ablemagic

Tabell 4: Brukere

### 4.3 Brukermiljøet

Denne mellomvaren må være enkel å bruke sammen med Unity. Siden programvaren skal brukes etter gruppen slutter vedlikehold må dokumentasjonen være spesielt god. God dokumentasjon av koden og klar brukerveiledning vil være avgjørende for om programmet blir brukt og vedlikeholdt. Siden den skal distribueres som åpen kildekode er det mulig for folk utenfor gruppen å vedlikeholde programvaren etter avsluttet prosjektperiode. Dette vil bare skje dersom koden har et godt grunnlag og er brukervennlig.



## 4.4 Sammendrag av brukernes behov

Behov	Prioritet	Påvirker	Foreslått løsning
Motta posisjonsdata fra sensoren	Høy	Data-innhenting	Mellomvare som håndterer dette
Pakken er lett å bruke i prosjekter	Høy	Bruker-vennlighet	En Unity pakke som skal enkelt kunne kobles til sensorens webgrensesnitt
Godt dokumenterte metoder som er enkle å forstå hva gjør	Høy	Bruker-vennlighet	En Unity pakke som er godt dokumentert så bruker enkelt forstår hvilke metoder som gjør hva
Posisjonsdata kan omregnes fra relativ posisjon i forhold til sensoren til kartesiske koordinater i rommet	Høy	Data-håndtering	Mellomvare som håndterer dette
Koblingen til serveren er stabil under bruk	Høy	Data-innhenting	MQTT-server vil kjøre som del av brukerens Unity-prosjekt og dermed være like stabil som prosjektet
Motta bevegelsesdata fra sensoren	Medium	Data-innhenting	Mellomvare som håndterer dette
Lett å navigere dokumentasjon på nett	Medium	Bruker-vennlighet	GitHub Wikien vil tilgjengeliggjøre dokumentasjonen
Oppsett av posisjonen til sensoren og andre data for omregninger er enkelt, men ikke nødvendig dersom bare rådata er ønsket	Lav	Bruker-vennlighet	Dette er en valgfri del av oppsettet til pakken

Tabell 5: Brukernes behov

## 4.5 Alternativer til vårt produkt

Per 03.02.2023 finnes det ingen gode alternativer til vårt produkt på markedet. Det er mulig å lage sin egen løsning i Unity for en utvikler, og utviklere vil ha kunnskapen til å gjøre dette. Løsningen utviklere lager selv vil være mer rettet mot deres behov, men vil ta tid og energi å lage.

## 5 Produktoversikt

### 5.1 Produktets rolle i brukermiljøet

I brukermiljøet i dag er det vanskelig å ta i bruk sensoren, da det krevet at utvikleren som vil bruke sensoren må ha kjennskap til en rekke forskjellige teknologier og standarder, sensorens API sender nemlig data i JSON-format over MQTT-protokollen. Produktet vårt har dermed som hensikt å senke terskelen til å bruke denne sensoren til interaktive prosjekter.

I tillegg til å senke terskelen for folk flest som ikke har kunnskapen til å implementere sin egen løsning, vil vi også senke terskelen til de utviklerne som har kapasiteten til å utvikle sin egen implementasjon. For å oppnå dette målet er det viktig at mellomvaren vi utvikler gjør det vesentlig enklere å ta i bruk sensoren, uten tap av funksjonalitet.

### 5.2 Forutsetninger og avhengigheter

Siden prosjektet er en utvidelsespakke til en Unitys kodebase er vi avhengig av Unity sin stabilitet. Da Unity er en godt etablert spillmotor kan vi anta at sjansen for ekstreme omveltninger er lav. Dette vil forhåpentligvis redusere behovet for langsiktig vedlikehold. Videre vil koden være avhengig av flere andre kodebiblioteker, for eksempel MQTT-protokoll biblioteker, matematiske biblioteker, eller andre biblioteker. Det vil derfor bli viktig å tenke gjennom valg av disse bibliotekene for å minimere vedlikeholdsbehovene til kodebasen.

Gruppen er også avhengig av at sensoren og annet maskinvare er funksjonelt og tilgjengelig gjennom prosessen sånn at utviklingen kan følge forventet plan.

## 6 Produktets funksjonelle egenskaper

Pakken har relativt enkle funksjonelle krav da fokuset i prosjektet i stor grad er på de ikke funksjonelle kravene som dokumentasjon og brukervennlighet. Under er en liste av de funksjonelle kravene som forutsatt av gruppen ved prosjektstart.

- MQTT-server som kommuniserer mellom Unity og sensoren
- Pakken kan bli installert i alle typer Unity-prosjekt
- Funksjonalitet for å konvertere JSON-data fra sensoren til datastrukturen vi velger å bruke
- Funksjonalitet for å konvertere koordinat-data fra sensoren til koordinat-data som kan brukes i Unity, for eksempel til å posisjonere spill-objekter som spillerens avatar
- Funksjonalitet for å lese bevegelsesdata fra sensoren
- Server starter når Unity-prosjektet starter
- Mulighet for å kommunisere med flere sensorer samtidig
- Funksjonalitet for å registrere sensorens plassering i spillmiljøet med id
- Funksjonalitet for å bestemme hva pakken skal lese av informasjon fra en gitt sensor (posisjonsdata, bevegelsesdata, begge)
- Funksjonalitet for å bare få informasjon om det er en person i en gitt sone fra sensor
- Funksjonalitet for overvåkning av hvor mange mennesker som er i et bygg/rom ved hjelp av en eller flere sensorer
- Støtte for begge størrelsene pakke sensoren kan sende og automatisk deteksjon dette

## 7 Ikke-funksjonelle egenskaper og andre krav

- All funksjonalitet som brukere kan ta i bruk må være godt dokumentert. Dette er for at brukere enkelt kan forstå hvilken funksjonalitet pakken har, og for at de skal ta denne i bruk på en god og effektiv måte med lite friksjon.
- API-et vi utvikler må være godt testet slik at det oppfører seg som forventet fra brukerens side. Vi vil sikte på å høy testdekningsgrad der vi ser at dette er hensiktsmessig.
- Produktet skal være responsivt, og gi god tilbakemelding til bruker når ting laster eller av andre grunner stopper opp.
- Produktet skal gi god beskjed dersom feil eller mangler oppstår, og loggføre dette slik at det kan sendes til utvikler slik at dette kan rettes opp i.
- Produktet skal ha et oversiktlig brukergrensesnitt som er klart, tydelig og lett å forstå.
- Brukergrensesnittet skal følge WCAG 2 der det er mulig og/eller hensiktsmessig.
- Produktet skal være lett å videreutvikle, slik at andre utviklere kan legge til funksjoner eller fikse feil dersom de ønsker det.
- Produktet skal være kompatibelt med så mange Unity-versjoner som mulig, men "Long Term Support"-versjoner (LTS) skal være prioritert.
- Produktet skal virke som forventet på Windows, MacOS og Linux.
- Produktet skal være trygt, og ikke skape flere sikkerhetshull enn de sensoren eventuelt har fra før.
- Produktet skal være modulært og tilby bruker muligheten til å legge til ikke essensiell funksjonalitet kun ved behov.

## Referanser

- [1] Lindsay Schardon. What is Unity? – A Guide for One of the Top Game Engines, January 2023. URL <https://gamedevacademy.org/what-is-unity/>. Section: 2D.
- [2] Rafael Valencia-García, Katty Lagos-Ortiz, Gema Alcaraz-Mármol, Javier del Cioppo, and Nestor Vera-Lucio. *Technologies and Innovation: Second International Conference, CITI 2016, Guayaquil, Ecuador, November 23-25, 2016, Proceedings*. Springer International Publishing, October 2016. ISBN 978-3-319-48023-7. Google-Books-ID: 1vcMvgAACAAJ.
- [3] The HiveMQ Team. Getting Started with MQTT, April 2020. URL <https://www.hivemq.com/blog/how-to-get-started-with-mqtt/>.
- [4] The HiveMQ Team. Publish & Subscribe - MQTT Essentials: Part 2, January 2015. URL <https://www.hivemq.com/blog/mqtt-essentials-part2-publish-subscribe/>.
- [5] Able Magic AS. Ablemagic - om oss, 2022. URL <https://www.ablemagic.no/about>.