

Tampereen ammattikorkeakoulu
Tietotekniikan koulutusohjelma
Ohjelmistotekniikka
Matti Tuuri

Opinnäytetyö

Sensoriverkkosovellus Maemo5- ja Symbian-alustoille

Työn ohjaaja
Työn tilaaja
Tampere 12/2010

Lehtori Tony Torp
MeshWorks Wireless Oy, ohjaajana
ohjelmistosuunnittelija Henri Vuorela

Tampereen ammattikorkeakoulu
Tietotekniikan koulutusohjelma, Ohjelmistotekniikka

Tekijä: Matti Tuuri

Työn nimi: Sensoriverkkosovellus Maemo5- ja Symbian-alustoille

Sivumäärä: 24

Valmistumisaika: 3.12.2010

Työn ohjaaja: Lehtori Tony Torp

Työn tilaaja: MeshWorks Wireless Oy

TIIVISTELMÄ

Tässä työssä toteutettiin ZigBee-verkkotekniikan avulla kerättyyn tietoon pohjautuva sensoriverkkosovellus. Työn tarkoituksena oli määritellä ja toteuttaa MeshWorks Wireless Oy:lle aputyökalu langattomien sensoriverkkojen asennusta varten.

Työssä käytettiin apuna Tampereen ammattikorkeakoulussa ja muissa tiloissa sijaitsevia sensoriverkkoja. Työn tavoitteena oli helpottaa langattomien sensoriverkkojen asennusvaihetta matkapuhelimella toimivan aputyökalun avulla. Sovelluksen avulla langattomien sensoriverkkojen asentaja pystyy helposti tarkistamaan matkapuhelimen avulla laitteiden yhdistymisen verkkoon, jolloin erillistä tietokonetta ei tarvitse kantaa mukana asennusvaiheessa.

Työn tuloksena on saatu aikaan helppokäyttöinen ja toimiva käyttöliittymä aputyökaluksi sensoriverkkojen asentajalle. Työtä voidaan kehittää edelleen useilla eri osa-alueilla ja siitä on hyötyä muiden sensoriverkkoihin pohjautuvien sovellusten kehittämisessä.

Writer: Matti Tuuri
Thesis: Sensor network program for Maemo5-and Symbian-
platforms
Pages: 24
Graduation time: 12/10
Thesis Supervisor: Lecturer Tony Topr
Co-operating company: MeshWorks Wireless Ltd

ABSTRACT

The purpose of this thesis was to develop a mobile application which uses data that has been collected with ZigBee -sensor network technique. The idea was to create a mobile tool for MeshWorks Wireless Ltd to improve the installation process of wireless sensor networks.

In this work there were used wireless sensor networks which were located at TAMK University of Applied Sciences and other environments. The purpose was to ease the installation process of wireless sensor networks with a tool that works on mobile phones. With the help of this software the installer can easily verify that devices are connected to the network. Therefore the installer does not have to carry a laptop with him during the process.

As a result of this thesis there were created a user-friendly and useful user interface for the installer of sensor networks. In the future this software can be developed in several different sectors. The software that was created is also very useful for developing other similar wireless sensor network applications.

SISÄLLYSLUETTELO

TIIVISTELMÄ	2
ABSTRACT	3
SISÄLLYSLUETTELO.....	4
LYHENTEET JA TERMISTÖ	5
1. JOHDANTO	1
2. SENSORIVERKKO	2
2.1 ZIGBEE	3
2.2 MESHWORKS WIRELESS OY	3
2.2.1 SeeMoto	3
2.2.2 FeelPlace	4
2.3 LAITTEET	4
2.3.1 TGG – Transceiver Gateway.....	5
2.3.2 TS - Temperature Sensor	5
2.3.3 THS – Temperature and Humidity Sensor	6
2.3.4 PS – Pulse Sensor.....	6
3. KÄYTTÖLIITTYMÄN SUUNNITTELU.....	8
3.1 SOVELLUKSEN KÄYTTÖTARKOITUS.....	8
3.2 KÄYTETYT TEKNIIKAT	9
3.2.1 Qt	9
3.2.2 JSON	9
3.2.3 JavaScript	10
3.3 KÄYTETYT TYÖKALUT.....	10
3.3.1 Nokia Qt SDK.....	10
3.3.2 Qt Creator	10
3.3.2 Mad Developer.....	11
4. MOBIILISOVELLUKSEN TOTEUTUS	12
4.1 TIEDON HAKU JA KYSELYN RAKENNE	12
4.1.1 Device Filter.....	13
4.1.2 Attribute Filter.....	14
4.1.3 History Filter.....	14
4.2. KÄYTTÖLIITTYMÄ	14
4.2.1 Kirjautuminen.....	15
4.2.2 Asset-näkymä.....	15
4.2.3 Laitenäkymä.....	16
4.2.4 Mittaustietonäkymä	17
4.3 ASENNUS	18
4.3.1 Nokia N900	18
4.3.2 Nokia 5800 XpressMusic	19
5. MOBIILISOVELLUKSEN TESTAUS.....	21
5.1 TESTIN TOTEUTUS	21
5.2 TESTAUS	22
5.3 LOPPUTULOS	22
6. YHTEENVETO	23
LÄHTEET	24

LYHENTEET JA TERMISTÖ

JavaScript	Web-ympäristössä käytettävä komentosarjakieli.
ZigBee	Lyhyen kantaman tietoliikenneverkko.
ZigBee-allianssi	Yhteisö, joka vastaa Zigbee-standardin kehittämisestä.
JSON	(JavaScript Object Notation) on yksinkertainen kuvauskieli.
XML	eXtensible Markup Language on merkintäkieli, jota käytetään tiedonvälitykseen järjestelmien välillä.
Qt	Alustariippumaton ohjelmien kehitysympäristö.
WSN	(Wireless Sensor Network) Langattomien sensorien verkko.
SeeMoto	Liikkuvaan kalustonhallintaan kehitetty kokonaisratkaisu.
FeelPlace	Palvelu energiatehokkuuden parantamiseksi kiinteistöissä.
Qt-Creator	Avoimeen lähdekoodiin perustuva työkalu Qt-kehitykseen.
GPS	(Global Positioning System) on yhdysvaltain armeijan kehittämä satelliittipaikannusjärjestelmä.
GSM	Global System for Mobile Communications on maailmanlaajuinen matkapuhelinjärjestelmä.
WiFi	WLAN (Wireless Local Area Network) on tietokoneiden langattomissa verkoissa yleisesti käytetty tekniikka.
Bluetooth	Avoin standardi laitteiden langattomaan kommunikointiin lähietäisyydellä.
Maemo5	Linux-pohjainen kehitysalusta kämmenmikroille.
Symbian	Älypuhelimille tarkoitettu käyttöjärjestelmä.
HTTP-POST	Tiedonsiirtotapa, jolla tiedot lähetetään selaimesta palvelimelle.
SIS	Symbian-tiedostojen asennuspaketti.
MAC-osoite	(Media Acces Control) yksilöi verkossa olevan laitteen.

1. JOHDANTO

Langattomat sensoriverkot ovat uutta tekniikkaa, joka mahdollistaa monipuolisen tiedonkeruun ja edullisten sekä vähän virtaa kuluttavien järjestelmien toteuttamisen. Zig-Been 2.4 GHz:n taajuutta käyttävät verkot mahdollistavat langattoman tiedonsiirron lyhyillä kantama-alueilla. ZigBee-verkot voivat sisältää suuria määriä laitteita, jotka lähettävät esimerkiksi mittaustietoa tietyin aikavälein.

MeshWorks Wireless Oy on vuonna 2007 perustettu Tampereella sijaitseva langattomaan tiedonsiirtotekniikkaan erikoistunut yritys. Yritys tuottaa monille eri toimialoille langattomia ratkaisuja, jotka mahdollistavat reaaliaikaisen näkymän liiketoiminnan kannalta tärkeään mittaustietoon. MeshWorks Wireless Oy:n tuotemerkkeihin kuuluu mm. SeeMoto ja FeelPlace. Vuodesta 2010 MeshWorks Wireless Oy on kuulunut osaksi E-Bros Oy:tä, jonka osaamisalueina ovat logistiikka- ja varastojärjestelmien hallintasovellukset.

Tämän työn tarkoituksena on perehtyä langattomiin sensoriverkkoihin ja suunnitella sekä toteuttaa SeeMoto-palvelun rajapintaa hyödyntävä mobiilisovellus, jota on tarkoitus hyödyntää apputyökaluna langattomien sensoriverkkojen asennuksessa. Työssä on käytetty Qt-sovelluskehitysympäristöä ja JSON (JavaScript Object Notation) -kuvauskieltä. Sovellus on testattu käyttäen Nokia N900 ja 5800 XpressMusic-puhelimia.

Opinnäytetyö sisältää perehtymistä langattomiin sensoriverkkoihin, Qt-ohjelmointikielen opiskelua ja SeeMoto-palvelun rajapinnan testausta. Työ painottuu mobiilisovelluksen toteutukseen ja lopulta sen testaukseen todellisuutta vastaavassa asennusympäristössä.

Lopussa tarkastellaan rajapinnan ja mobiilisovelluksen erilaisia jatkokehitysmahdollisuuksia ja pohditaan eri vaihtoehtoja langattomien sensoriverkkojen asennuksen helpottamiseksi.

2. SENSORIVERKKO

Sensoriverkkoja asennetaan moniin eri käyttökohteisiin. Verkot voivat olla liikkuvia, jolloin kyseessä ovat ajoneuvot ja varastot. Kiinteitä verkkoja ovat esimerkiksi rakennukset tai alueet. Sensoriverkko koostuu verkon tukiasemasta (Gateway), ja sen kanta-alueella sijaitsevista langattomista antureista.

Verkossa tapahtuva tiedonsiirto on toteutettu käyttäen ZigBee-standardin mukaista tiedonsiirtotekniikkaa. Anturit lähettävät mittaustietoa tukiasemalle joko suoraan tai verkossa olevien reitittimien kautta (Kuvio 1). Tukiasema lähettää tiedot GSM-yhteyden avulla SeeMoto-palveluun, josta tietoa voidaan tutkia ja selata Internet-selaimen avulla. Tukiasema sisältää GPS-paikantimen, jonka avulla jokaiselle mittaustiedolle saadaan paikkatieto. Tässä opinnäytetyössä tieto kysytään rajapinnan avulla SeeMoto-palvelusta, jota hyödyntäen toteutetaan mobiilisovellus.



Kuvio 1: SeeMoto-sensoriverkon rakenne. (lähde: www.seemoto.com)

2.1 ZigBee

ZigBee IEEE 802.15.4 -standardi määrittelee lyhyen kantaman vähän virtaa kuluttavan tietoliikenneverkon. Ensimmäisten ZigBee-verkkojen suunnittelu aloitettiin vuonna 1998, jolloin olemassa olevien tekniikoiden kuten WiFin ja Bluetoothin arvioitiin olevan riittämättömiä kaikkiin käyttötarkoituksiin. Ensimmäinen ZigBee-standardi valmistui syyskuussa 2003. Tämän standardin kehittämisestä vastaa ZigBee-allianssi. /1/

Tekniikkaa hyödyntäen voidaan toteuttaa pienitehoisia langattomia laitteita, joiden paristojen käyttöikä voi olla useita vuosia. Standardin mukaan ZigBee-verkot voivat sisältää jopa yli 65 000 laitetta. Aktivoituessaan laitteet kytkeytyvät todella nopeasti verkkoon ja ovat muun aikaa vähän virtaa kuluttavassa sleep-tilassa. Bluetooth-tekniikkaan verrattaessa ZigBee-verkoilla keskimääräinen verkkoon kytkeytymisaika on selvästi nopeampi ja verkon laitteiden maksimimäärä suurempi. /2/

2.2 MeshWorks Wireless Oy

MeshWorks Wireless Oy on vuonna 2007 perustettu Tampereella sijaitseva langattomaan tiedonsiirtotekniikkaan erikoistunut yritys. Yritys tuottaa monille eri toimialoille langattomia ratkaisuja, jotka mahdollistavat reaaliaikaisen näkymän liiketoiminnan kannalta tärkeään mittaustietoon. MeshWorks Wireless Oy:n tuotemerkkeihin kuuluvat mm. SeeMoto ja FeelPlace. /3/

Vuodesta 2010 MeshWorks Wireless Oy on kuulunut osaksi E-Bros Oy:tä, jonka osamisalueina ovat logistiikka- ja varastojärjestelmien hallintasovellukset.

2.2.1 SeeMoto

SeeMoto on liikkuvaan kalustonhallintaan keskittynyt kokonaisratkaisu, jonka avulla yritykset voivat hyödyntää paikannus- ja etähallintajärjestelmiä. Palvelu koostuu paikannus- ja tiedonkeruulaitteista, selainpohjaisesta käyttöliittymästä sekä tiedonsiirtorajapinnasta.

SeeMoto tarjoaa reaaliaikaista tietoa yrityksille eri toimintaympäristöissä. Palvelun avulla yritykset voivat säästää kustannuksissa, kehittää palvelun laatua ja parantaa tehokkuutta. Palvelussa olevan tiedon voi tarvittaessa jakaa edelleen asiakkaille ja yhteistyökumppaneille. /4/

Tuotteet ja palveluratkaisut soveltuvat monille toimialoille, kuten elintarviketeollisuuteen, kuljetukseen, huolto- ja asennuspalveluihin, rakentamiseen, kiinteistöihin, prosessiteollisuuteen, kone- ja valmisteollisuuteen. Palvelu lisää tietoisuutta kaluston ja kohteiden sijainnista, tilasta, mittausarvoista, kunnosta ja ympäristöstä.

2.2.2 FeelPlace

FeelPlace tähtää energiatehokkuuden parantamiseen kiinteistöissä. Palvelu tarjoaa asiakkaalle mahdollisuuden seurata reaaliaikaisesti mm. energian kulutusta. FeelPlace on yhdistelmä langattomia antureita, tiedonkeruuta, raportteja, tiedon jakamista ja yhdistämistä muihin palveluihin. /4/

2.3 Laitteet

MeshWorks Wireless Oy tarjoaa useita eri laitteita, jotka mahdollistavat eri mittaustietojen keräämisen monista eri ympäristöistä. Verkkoon kuuluu tukiasema, johon sen kantama-alueella olevat anturit ottavat yhteyttä. Anturit mittaavat mm. lämpötilaa, kosteutta, hiilidioksidipitoisuutta, virrankulutusta ja polttoainetietoja.

Mitattuaan tiedot, anturit lähettävät tiedot tukiasemalle joko suoraan tai verkossa olevan reitittimen kautta. Anturin ollessa verkon kantama-alueen ulkopuolella se tallentaa mittaustiedot muistiin ja seuraavan kerran liittyessään verkkoon lähettää tiedot tukiasemalle. Tällä voidaan varmistaa katkeamaton tiedonkeräys esimerkiksi sähkökatkon sattuessa. Verkon laitteet käyttävät mesh-topologiaa, joka mahdollistaa tiedon kulkemisen useaa eri siirtotietä hyväksi käyttäen.

Tukiasema välittää tiedot palvelimelle, jossa ne parsitaan ja joista sitten muodostetaan käyttäjille erilaisia kuvaajia ja raportteja. Tämän jälkeen tiedot ovat kyseltävissä rajapinnan kautta palvelusta. Alla on esitelty SeeMoto-palvelun tuoteperheeseen kuuluvia laitteita.

2.3.1 TGG – Transceiver Gateway

TGG (Transceiver Gateway) on yksi tukiasemana toimiva laite (Kuvio 2). Laite sisältää tiedonkeruun lisäksi GPS-paikantimen, joka mahdollistaa ajoneuvojen paikannuksen ja mittaustietojen leimaamisen ajan lisäksi myös paikkatiedolla. TGG muodostaa linkin antureiden ja SeeMoto-palvelun välillä.



Kuvio 2: Transceiver Gateway (lähde: <http://www.seemoto.com/>)

2.3.2 TS - Temperature Sensor

TS (Temperature Sensor) on SeeMoto-järjestelmän langaton lämpötila-anturi (Kuvio 3). Anturi on suunniteltu tuore- ja pakaste-elintarvikkeiden lämpötilan seurantaan kuljetuksen, jakelun ja säilytyksen aikana. Anturi on vesitiivis paristoilla toimiva laite, jonka toiminta-aika on useita vuosia. Mikäli anturi on tukiaseman kantaman ulottumattomissa se voi tallentaa yli 8000 mittausarvoa muistiin ja lähettää ne tukiasemalle saadessaan seuraavan kerran yhteyden verkkoon.



Kuvio 3: Temperature Sensor (lähde: <http://www.seemoto.com/>)

2.3.3 THS – Temperature and Humidity Sensor

THS (Temperature and Humidity Sensor) on anturi, jonka toiminta on muutoin samanlainen kuin lämpötila-anturin, mutta se mittaa lämpötilan lisäksi myös ilmankosteutta (Kuvio 4). Käyttökohteita ovat esimerkiksi sisätilan ilmankosteuden ja lämpötilan valvonta. Ilmankosteuden mittaustavan takia anturi ei voi olla vesitiivis.



Kuvio 4: Temperature and Humidity Sensor (lähde: <http://www.seemoto.com/>)

2.3.4 PS – Pulse Sensor

PS (Pulse Sensor) on anturi, jolla voidaan mitata relelähdön tilanvaihto (Kuvio 5). Anturia käytetään usein myös pulssilaskurina, jolloin sillä voidaan laskea esimerkiksi kulutetun sähkön määrä. Käyttökohteita ovat mm. kiinteistöjen sähkön mittaus, murtohälytykset ja taloautomaatio.



Kuvio 5: Pulse Sensor (lähde: <http://www.seemoto.com/>)

3. KÄYTTÖLIITTYMÄN SUUNNITTELU

Mobiilisovelluksen käyttöliittymää suunniteltaessa tarkoituksena oli tehdä toimiva ja yksinkertainen käyttöliittymä, jolloin käyttäjä pystyy keskittymään varsinaiseen asennustyöhön. Suunnittelu aloitettiin tutustumalla ZigBee-laitteiden eri käyttömahdollisuuksiin, sekä kartoittamalla sovelluksen tarpeet.

3.1 Sovelluksen käyttötarkoitus

Mobiilisovelluksen käyttötarkoitus on toimia aputyökaluna langattomien sensoriverkkojen asennuksessa. Sovellus toteutuksessa hyödynnettiin SeeMoto-palvelun JSON-rajapintaa.

Sovellusta suunniteltaessa antureiden yhteydenoton tarkistamiseen ei ollut olemassa erillisiä asennusta helpottavia työkaluja. Antureiden asennuksen jälkeen täytyi yhteyden toimivuus tarkistaa kannettavan tietokoneen avulla palvelusta. Tämä hidasti asennusta, koska aina ei ollut tietokoneita lähellä, joten oma tietokone täytyi olla mukana.

Mobiilisovelluksen tarkoituksena on helpottaa ja nopeuttaa asennuksessa tapahtuvan langattoman verkon antureiden yhteyden tarkastamista. Verkon asentaja voi tarkistaa laitteiden yhdistymisen puhelimestaan, jolloin erillistä tietokonetta ei tarvitse kantaa mukana.

Sovellus on toteutettu niin, että sen käyttäjä (asentaja) pääsee näkemään oman tunnukseen alle liitetyt langattomat laitteet reaaliajassa. Sovelluksen avulla asentaja voi tarkistaa, että kohteeseen sijoitetut laitteet ovat yhteydessä palveluun ja näin varmistaa, että asennus on suoritettu oikein. Käyttöliittymän tarkoituksena on olla niin yksinkertainen, ettei käyttäjällä tarvitse olla mitään kokemusta langattomista sensoriverkoista tai niissä käytetyistä tekniikoista.

3.2 Käytetyt tekniikat

Mobiilisovellus on toteutettu käyttäen alustariippumatonta Qt-ohjelmointikieltä. Tämä mahdollistaa saman lähdekoodin kääntämisen usealle eri alustalle. Langattomien laitteiden tiedot kysytään käyttäen SeeMoto-palvelun JSON-rajapintaa, jonka tiedonsiirto on toteutettu käyttäen HTTP POST-pyyntömetodia.

3.2.1 Qt

Qt on alustariippumaton ohjelmistojen ja käyttöliittymien kehitysalusta, joka sisältää C++ -luokkakirjaston ja alustariippumattoman ohjelmointiympäristön. Qt:n omistaa nykyään Nokia, mutta sitä on ennen kehittänyt norjalainen Trolltech-yhtiö jo vuodesta 1996 lähtien. Qt:n kehittämisen ajatuksena oli tehdä kirjasto, jonka avulla ohjelmat olisi helppo kääntää erilaisille alustoille. /5/

Tuettuina alustoina ovat esimerkiksi Linux/Unix, Mac OS X ja Windows-alustat. Lisäksi tuettuina ovat jotkin sulautetut alustat, kuten Maemo ja Symbian. Tunnettuja Qt:n avulla kehitettyjä ohjelmistoja ovat mm. Internet-selain Opera ja Googlen omistama Google Earth-palvelu. /5/

3.2.2 JSON

JSON (Java Script Object Notation) on yksinkertainen kuvauskieli, jota on helppo käyttää JavaScript-ohjelmissa. Nimestään huolimatta JSON on JavaScriptistä riippumaton, eli sitä voidaan käyttää myös muilla ohjelmointikielillä.

JSON on tullut tunnetuksi JavaScriptin kautta, koska se on helppo tapa eri Internet-sivustojen kommunikointiin ilman serverikieliä. JSON saattaa tulevaisuudessa olla myös XML:n (eXtensible Markup Language) korvaaja, koska se on tiiviimpää kuin XML ja helpommin purettava. /6/

3.2.3 JavaScript

JavaScript on alun perin Netscape Communications Corporationin kehittämä, pääasiassa web-ympäristössä käytettävä, komentosarjakieli. Se on yksinkertainen, suoraan html-koodin sekaan kirjoitettava skriptikieli. Se mahdollistaa interaktiivisten toimintojen lisäämisen www-sivuille, esimerkiksi lomaketietojen tarkastamisen tai erilaiset valikko-ohjatut www-sivuratkaisut. /7/

3.3 Käytetyt työkalut

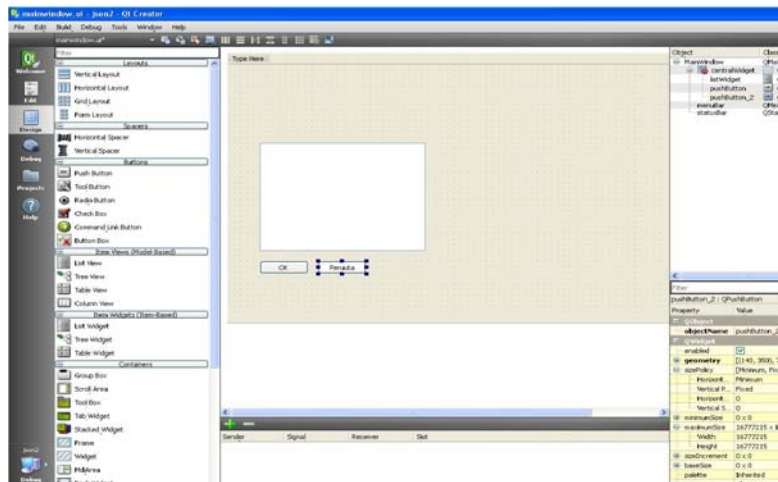
Sovelluskehitys aloitettiin etsimällä sopivat työkalut sensoriverkkosovelluksen toteutusta varten. Sovelluskehitysympäristönä Qt on varsin uusi, mutta nykyään kaikki tarvittavat kehitystyökalut löytyvät yhdestä paketista. Kehitysalustaksi valittiin Nokia Qt SDK:n sisältämä Qt Creator ja testausta varten valittiin N900-puhelimelle asennettava Mad Developer.

3.3.1 Nokia Qt SDK

Nokia Qt SDK sisältää valmiit työkalut ohjelmistokehitykseen usealle eri alustalle. Sen avulla sovelluskehittäjät voivat suoraan luoda sovelluksia Maemolle, Symbianille ja Meegolle. Tarkoitus on helpottaa sovelluskehittäjän työtä ja ohjelmiston jakelumahdollisuuksia, koska sama ohjelmakoodi saadaan toimimaan suoraan monella alustalla. Nokia Qt SDK:n mukana tulee myös Qt Creator –kehitysympäristö. /8/

3.3.2 Qt Creator

Qt Creator (kuviot 6) on Nokian kehittämä alustariippumaton ohjelmistojen kehitysympäristö. Sen tarkoitus on olla työkaluna Qt-ohjelmoinnissa, mutta sitä voidaan käyttää myös esimerkiksi tavalliseen C++-kehitykseen. Qt Creator on osa Qt-SDK:ta, joka sisältää myös kääntäjäympäristöt Symbianille ja Maemo5:lle. /8/



Kuvio 6: Qt Creator

3.3.2 Mad Developer

Mad Developer (Kuvio 7) on Maemolle kehitetty sovellus, jonka tarkoitus on nopeuttaa Qt-sovellusten kehitystä ja testausta. Mad Developerin ansiosta sovellus voidaan suorittaa suoraan kohdelaitteessa, eikä siitä tarvitse erikseen luoda asennuspakettia, siirtää laitteeseen ja asentaa.



Kuvio 7: Mad Developer

Rajapinta palauttaa kyselyn määritysten mukaisesti tiedot tietokannasta. Sovellus parsii tiedot omaan tietorakenteeseensa käyttäen hyväksi Qt:n script-kirjastoa. Tämän jälkeen tieto on helposti haettavissa sovelluksen kaikissa näkymissä.

Kyselyn rakenne muodostuu neljästä osasta (esimerkki 2): käyttäjän tunnistamisesta (identification), laitteiden suodatukselta (device_filter), attribuuttien suodatukselta (attribute_filter) ja historiatietojen hakemisesta (history_filter). Käyttäjän tunnistus on ainoa pakollinen tieto kyselyssä. Mikäli suodattimia ei kyselyssä käytetä, rajapinta palauttaa kaikki käyttäjälle määritettyjen laitteiden tiedot. Tämä mahdollistaa monipuolisten kyselyiden tekemisen rajapintaan.

```

{
  "request": {
    "identification": {
      ...
    },
    "device_filter": {
      ...
    },
    "attribute_filter": {
      ...
    },
    "history_filter": {
      ...
    }
  }
}

```

Esimerkki 2: Kyselyn rakenne

4.1.1 Device Filter

Device Filter mahdollistaa vastauksen suodattamisen laitteen osoitteen, tyypin ja ryhmän (asset) mukaan. Tämän avulla voidaan kysyä esimerkiksi yhden tai useamman laitteen tai ryhmän tiedot (esimerkki 3).

```

"device_filter": {
  "address": [
    "0050c2fffeacdb29"
  ]
}

```

Esimerkki 3: Device Filter

4.1.2 Attribute Filter

Attribute Filterin avulla voidaan suodattaa kyselyssä palautettavien laitteiden attribuuttien määrää. Tämä mahdollistaa esimerkiksi vain lämpötila-arvojen palauttamisen kaikkien arvojen sijaan. Lisäksi voidaan rajoittaa mittausarvojen lisätietojen määrää. Tässä työssä Attribute Filterin esimerkki 4) avulla rajapinnasta kysytään ainoastaan asetin id (assetid), nimi (assetname) ja laitteiden nimi (devname , osoite(address) ja mittaukset (meas). Mittausarvoille kysytään vain aikaleima (date).

```
"attribute_filter": {
  "data": [
    "assetid",
    "assetname",
    "devname",
    "address",
    "meas"
  ],
  "attr": [
    "date"
  ]
}
```

Esimerkki 4: Sovelluksessa käytetty attribuuttien suodatus (Attribute Filter)

4.1.3 History Filter

History Filterin avulla voidaan kysyä mittauksetietojen historiatietoja. Rajauksia historiatiedolle voidaan tehdä mittausarvojen sekvenssien, aikarajauksen ja tulosten määrien mukaan (esimerkki 5).

```
"history_filter": {
  "markers": "10"
}
```

Esimerkki 5: Kymmenen viimeisimmän mittausarvon kysyminen (History Filter)

4.2. Käyttöliittymä

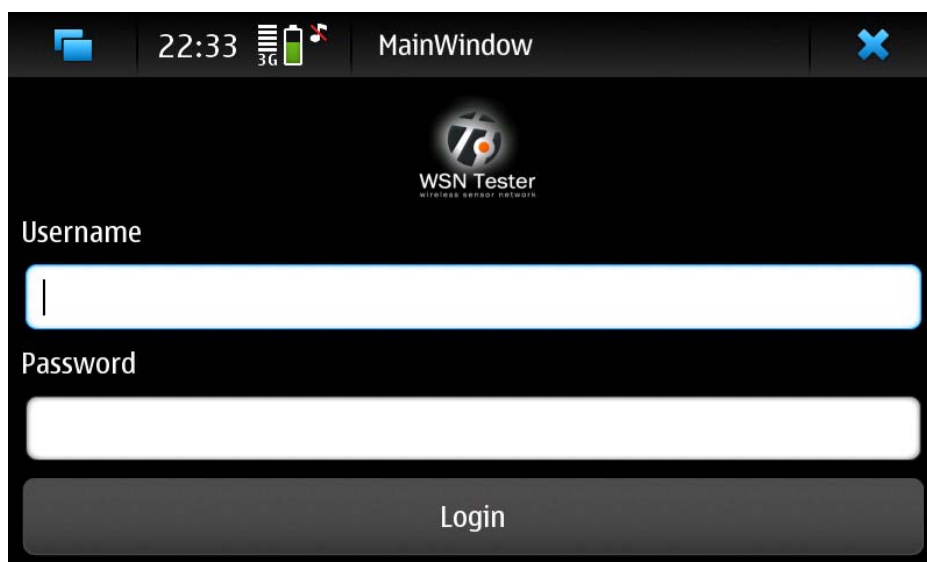
Käyttöliittymä koostuu neljästä eri näkymästä. Sovellusta käynnistettäessä käyttäjälle avautuu ensimmäisenä kirjautumisnäkyvä, johon syötetään käyttäjätunnus ja salasana. Ainoastaan kirjautumisnäkyvä on identtinen kaikilla käyttäjillä.

Kirjautumisnäkyvän lisäksi käyttöliittymän kuuluu kolme näkymää: asset-näkymä, laitenäkymä ja mittaustietonäkymä. Nämä ovat yksilöllisiä jokaiselle käyttäjälle, koska rajapinta palauttaa vain käyttäjälle määriteltyjen laitteiden tiedot.

4.2.1 Kirjautuminen

Sovellukseen kirjaudutaan käyttäjätunnuksen ja salasanan avulla (Kuvio 8). Painettaessa login-nappia sovellus lähettää käyttäjätunnuksen ja salasanan palvelimelle.

Sovellus tarkistaa, että käyttäjätunnus ja salasana löytyvät palvelimelta ja siirtyy sen jälkeen seuraavaan näkymään. Mikäli käyttäjätunnus- tai salasana-kenttien nimet jätetään tyhjäksi, ohjelma palauttaa virheilmoituksen, jossa kehoitetaan käyttäjää syöttämään tiedot uudestaan.

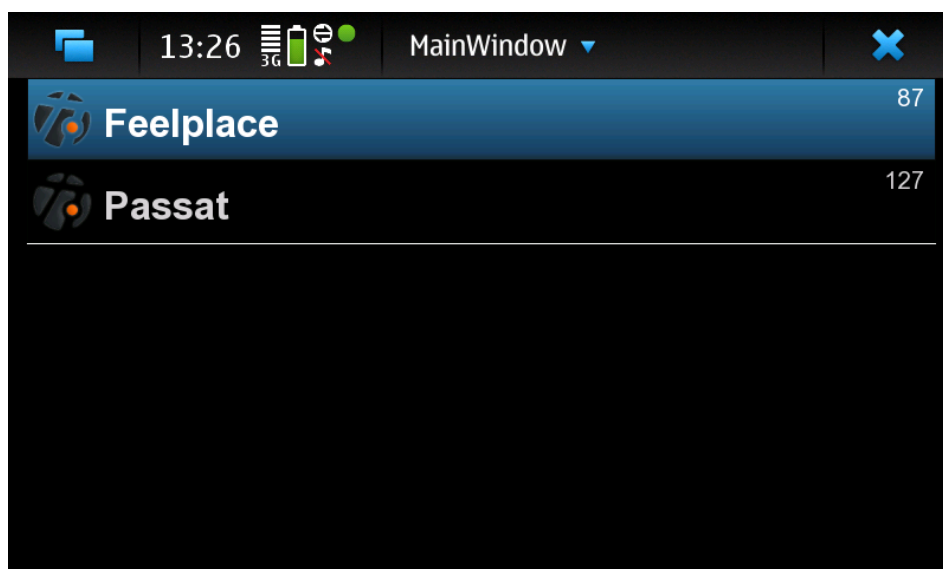


Kuvio 8: Kirjautumisnäkyvä

4.2.2 Asset-näkymä

Kirjautumisnäkyvän jälkeen avautuu asset-näkymä (Kuvio 9), johon listautuvat käyttäjälle määritellyt assetit eli seurattavat kohteet. Esimerkiksi asset-näkymässä näkyvä ”Passat” on yksi asset, joka sisältää kaikki kyseiseen kohteeseen liittyvät laitteet.

Jokaisella assetilla voi myös olla paikkatieto, eli siihen voi olla liitettynä paikantava laite (esimerkiksi tukiasema). Jokainen listalla oleva asset toimii linkkinä seuraavaan näkymään, jossa ovat tiedot asettiin kuuluvista laitteista.



Kuvio 9: Asset-näkymä.

4.2.3 Laitenäkymä

Laitenäkymässä näkyvät kaikki kohteeseen liittyvät laitteet (Kuvio 10). Näkymän tarkoituksena on näyttää, mitä laitteita kyseinen kohde sisältää ja mitkä laitteet ovat lähettäneet mittaustietoaan palvelimelle. Näkymän vasemmassa laidassa näkyvät eriväriset pallot, jotka kertovat yhteyden tilan. Yhteydenoton tarkistaminen tehdään vertaamalla viimeisimmän mittaustiedon kellonaikaa nykyaikaan. Ruudulla näkyvä yhdistelmä numeroita ja kirjaimia tarkoittaa mittaavan laitteen mac-osoitetta, jonka avulla asentaja voi varmistua oikeasta laitteesta.

Mikäli laite on lähettänyt tietoa viimeisen kymmenen minuutin aikana, näkymän vasemmassa laidassa oleva pallo loistaa vihreänä. Jos viimeisestä lähetyksestä on yli 10 minuuttia, mutta vähemmän kuin 20, loistaa pallon väri keltaisena. Muussa tapauksessa pallon väri on punainen, myös siinä tapauksessa kun mittaustietoa ei ole saatu ollenkaan.

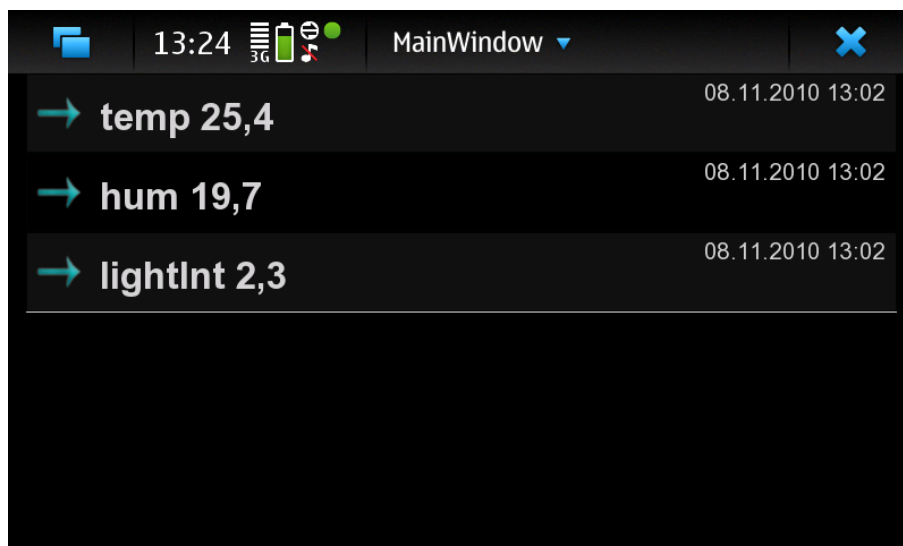
Laitenäkymän tarkoitus on helpottaa laitteiden asennusvaihetta kertomalla yhteydenotosta palvelimelle pallojen värien avulla. Klikkaamalla laitetta siirrytään mittaustietonäkymään, josta nähdään mitä mittaustietoja laite on lähettänyt.



Kuvio 10: Laitenäkymä

4.2.4 Mittaustietonäkymä

Mittaustietonäkymän tarkoituksena on kertoa mitä mittaustietoja laite lähettää (Kuvio 11). Anturityypin mukaan mitattavien arvojen määrä voi vaihdella, mikä on otettu mittaustietonäkymän toteutuksessa huomioon.



Kuvio 11: Mittaustietonäkymä

4.3 Asennus

Qt-kehitysympäristö sisältää alustariippumattoman ohjelmointiympäristön, joten se voidaan kääntää lukuisille eri ohjelmistoalustoille. Tässä työssä toteutettu mobiilisovellus käännettiin kahdelle eri alustalle: Maemolle ja Symbianille.

Nokia julkaisi vuoden 2010 kesällä Nokia Qt SDK:n, joka kerää tarvittavat työkalut yhteen, joten niiden avulla sovelluskehittäjä voi suoraan luoda sovelluksia Maemo- ja Symbian -alustoille. SDK:ta asennettaessa voidaan valita, mitkä kehitysalustat valitaan asennettavaksi käyttötarpeen mukaan. Maemo-alustan omaavaksi kehitysalustaksi valitsin työhön Nokian N900-puhelimen ja Symbian-kehitysalustan puhelimeksi Nokian 5800 XpressMusic -puhelimen.

4.3.1 Nokia N900

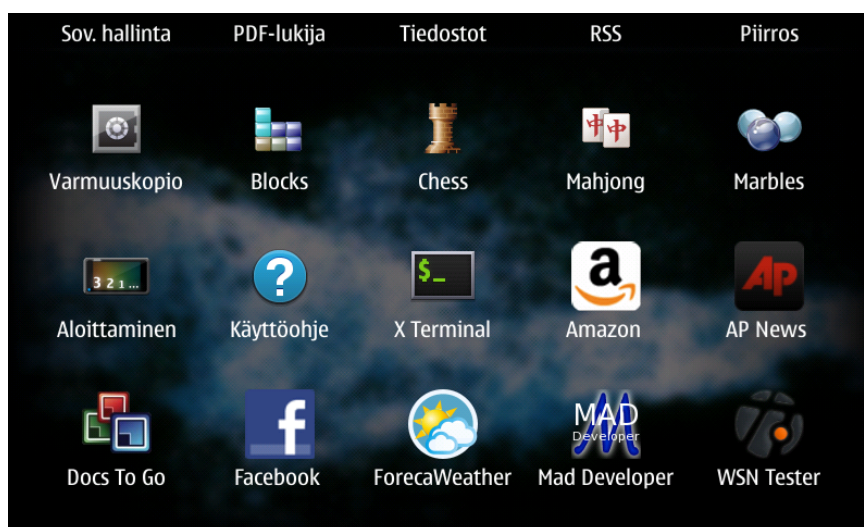
Nokia N900 on ensimmäinen Internet-tabletti, jossa yhdistyy Maemo-käyttöjärjestelmä ja puhelinominaisuudet. Nokia on tavallisesti julkaissut lähes kaikki viimeisimmät matkapuhelimet Symbian-käyttöjärjestelmällä, mutta tässä N900 tekee poikkeuksen. Maemo on Linux-pohjainen kehitysalusta kämmenmikroille. N900 on saanut suosiota sovelluskehittäjien joukossa, koska lähdekoodi on avoin. Tämä tarkoittaa sitä, että sovellusten tuominen järjestelmään on helppoa.

Uuden Nokia Qt SDK:n mukana tuleva Qt Creator luo automaattisesti debian-pakettitiedoston (.deb), kun sovellusta käännetään. Sovellusta voidaan käynnöksen jälkeen ajaa suoraan puhelimesta Mad Developer-työkalun avulla. Mad Developer myös samalla asentaa sovelluksen puhelimeen, joten sitä voidaan suorittaa laitteessa myös jälkeinpäin.

Nykyisellä Qt Creatorin versiolla .deb-paketin luominen onnistuu, mutta sovellus joudutaan edelleen käynnistämään komentorivin (x-terminal) kautta. Sovelluksen käynnösvaiheessa voidaan myös määritellä erikseen .desktop-tiedosto (esimerkki 6), jonka avulla asennettu sovellus saadaan näkyvään N900-puhelimen sovellusluettelossa (Kuvio 12). /9/

```
[Desktop Entry]
Encoding=UTF-8
Version=1.0
Type=Application
Name=<MyAppName>
Exec=/usr/local/bin/<myappname>
Icon=<appname>
StartupWMClass=
X-Window-Icon=<myappname>
X-HildonDesk-ShowInToolbar=true
X-Osso-Type=application/x-executable
Terminal=false
```

Esimerkki 6. Desktop-tiedoston sisältö



Kuvio 12: Sensoriverkkosovellus ”WSN Tester” sovellusluettelossa

4.3.2 Nokia 5800 XpressMusic

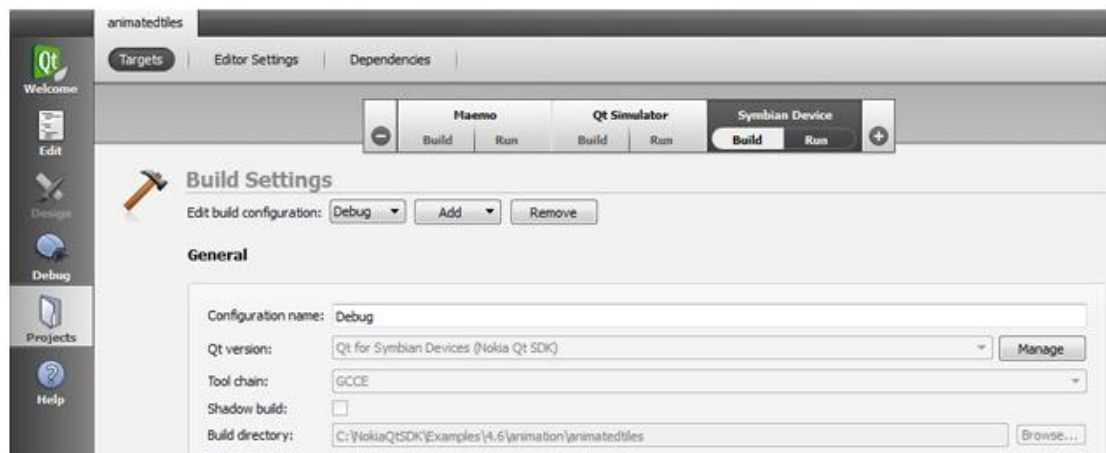
Nokia 5800 XpressMusic -puhelin on toteutettu S60 5th edition -sovellusalustalla, joka on rakennettu Symbian Os-käyttöjärjestelmän päälle. Verrattuna aikaisempiin sovellusalustoihin S60 5th edition tarjoaa hyödyllisiä ominaisuuksia sovelluskehittäjän näkökulmasta ajatellen.

Sovellusalustassa on mm. rajapinta, joka tukee erilaisten sensorien käyttöä laitteessa, sekä tuki kosketusnäytölle. Nokia Qt SDK tarjoaa monia aputyökaluja Symbian-kehitykseen, minkä vuoksi sitä käytettiin myös sensoriverkkosovelluksen asentamiseen puhelimelle.

Ennen Qt-sovelluksen suorittamista Symbian -käyttöjärjestelmässä, puhelimeen asennettiin Qt-kirjastot. Asentaminen onnistui suoraan Nokia Qt SDK:n aputyökalulla. Seu-

raavaksi asennettiin TRK-debugging, jolloin voitiin testata sovellusta suoraan päätelaitteella Qt Creatorin avulla, puhelimen ollessa liitettynä tietokoneeseen.

Qt Creatorista valittiin kehitysalustaksi Symbian (Kuvio 13). Tämän jälkeen sovellus voidaan testimelessä suorittaa puhelimesta. Lisäksi Qt creator luo ohjelman käännösvaiheessa sis-asennuspaketin, joka voidaan myöhemmin kopioida ja asentaa puhelimeen, jolloin se toimii kuin mikä tahansa puhelimeen asennettu sovellus.



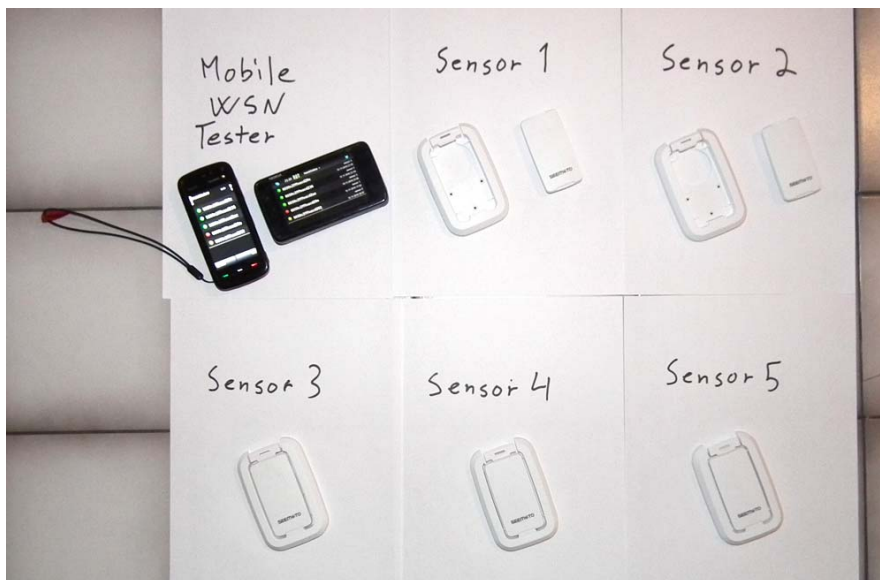
Kuvio 13: Kehitysalustaksi on valittu Symbian

5. MOBIILISOVELLUKSEN TESTAUS

Testin tarkoituksena oli kokeilla mobiilisovelluksen toimintaa simuloidussa asennusympäristössä. Sovelluksen tulisi näyttää värien avulla, mitkä anturit ovat yhteydessä tukiasemaan ja sitä kautta SeeMoto-palveluun. Antureiden tietojen lisäksi sovelluksesta tulee näkyä anturin keräämät mittaustiedot.

5.1 Testin toteutus

Mobiilisovelluksen testaus toteutettiin käyttäen TGG-tukiasemaa ja viittä lämpötilaa mittaavaa TS-anturia (Kuvio 14). Antureille annettiin nimet ”sensor 1”, ”sensor 2” jne. Tukiasemaan kytkettiin virta ja anturit yhdistivät tukiasemaan. Tämän jälkeen aloitimme sovelluksen testauksen.



Kuvio 14: Anturit, tukiasema ja testattavat puhelimet

5.2 Testaus

Testauksen aluksi ”sensor 1” irrotettiin taustalevystä, jolloin se lakkasi lähettämästä tietoa tukiasemalle ja aloitti tiedon tallentamisen omaan muistiinsa. Tämän jälkeen odotettiin yli kymmenen minuuttia, jonka jälkeen irrotettiin ”sensor 2”. Tämän jälkeen odotettiin vielä yli kymmenen minuuttia, jonka jälkeen käynnistettiin mobiilisovellus. Tämän menetelmän avulla pystyttiin simuloimaan tilanne, jossa ”sensor 1” ei ole saanut yhteyttä verkkoon yli 20 minuuttia (punainen väri). Lisäksi kymmenen minuuttia myöhemmin irrotettu ”sensor 2” oli väriltään keltainen. Loput kolme anturia, jotka ovat olleet koko ajan yhteydessä tukiasemaan, näkyvät vihreinä (Kuvio 15).



Kuvio 15: Sovellus käynnissä Nokia 5800 -ja N900-puhelimilla.

5.3 Lopputulos

Testauksen aikana testattiin kaikki asentajan kannalta tärkeimmät käyttöliittymän toiminnot. Käyttöliittymässä näkyvän anturien vasemmalla puolella sijaitsevien pallojen värit toimivat kaikissa testitapauksissa oikein, myös aikatiedot pitivät paikkansa. Lisäksi testattiin erilaisen uuden asettin luomista, antureiden liittämistä uuteen kohteeseen ja antureiden näkymistä käyttöliittymässä. Sovellus toimi testauksen aikana suunnitellulla tavalla ja on valmis hyötykäyttöön langattomien sensoriverkkojen asennuksessa.

6. YHTEENVETO

Langattomat sensoriverkot yleistyvät nopeasti ja niitä voi hyödyntää lähes kaikilla liiketoiminnan eri osa-alueilla. Nykyään tieto on monesti keskitetty Internet-pohjaisiin palveluihin, josta se on helposti selattavissa miltä tahansa tietokoneelta tai mobiililaitteelta. Monet selainpohjaiset palvelut ovat kuitenkin puhelimella hitaita ja hankalia käyttää pienillä näytöillä selattaessa. Tässä apuun tulevat erilaiset mobiilisovellukset, jotka nopeuttavat tiedon selausta.

Tässä työssä tehty mobiilisovellus toimii nopeana työkaluna sensoriverkkojen laitteiden mittaustietojen tarkastelussa. Sovellus on suunniteltu lähinnä verkon asennusta ja laitteiden yhteyksien tarkastamista varten. Käyttöliittymältään sovellus on mahdollisimman helppokäyttöinen, joten verkon tilan tarkastaminen onnistuu nopeasti asennuksen aikana.

Qt-kehitysympäristö oli oikea valinta sovelluksen kehittämiseen alustariippumattomuuden takia. Uusi Nokia Qt SDK helpotti sovelluskehitystä huomattavasti, koska kaikki tarvittava ohjelmien kääntämiseen usealle eri alustalle tulee samassa paketissa. Kehitysympäristön valitsemiseen vaikutti myös aikaisempi kokemus Qt:stä.

Toteutettu mobiilisovellus on tällä hetkellä toimiva ratkaisu verkon asennuksen aputyökaluksi. Sovellusta voisi kehittää myös asiakkaille sopivaksi työkaluksi, jolloin erilaisia jatkokehitysmahdollisuuksia löytyy useita. Esimerkiksi paikkatiedon hyödyntäminen voisi olla yksi lisä sovelluksen toimintaan. Myös antureiden mittaustietojen historia-arvojen kysely ja tarkastelu voisivat olla tarpeellisia ominaisuuksia varsinkin verkon käyttäjien kannalta.

LÄHTEET

Sähköiset lähteet

- /1/ Wikipedia: ZigBee [www-dokumentti] [Viitattu 4.10.2010].
Saatavissa: <http://fi.wikipedia.org/wiki/ZigBee>
- /2/ ZigBee Alliance [www-dokumentti] [Viitattu 15.10.2010].
Saatavissa: <http://www.zigbee.org>
- /3/ MeshWorks Wireless [www-dokumentti] [Viitattu 28.10.2010]
Saatavissa: <http://meshworkswireless.com/>
- /4/ SeeMoto [www-dokumentti] [Viitattu 27.10.2010].
Saatavissa: <http://www.seemoto.com/>
- /5/ Wikipedia: Qt [www-dokumentti] [Viitattu 19.10.2010]
Saatavissa: <http://fi.wikipedia.org/wiki/Qt>
- /6/ JSON [www-dokumentti] [Viitattu 24.10.2010]
Saatavissa: <http://www.json.org/>
- /7/ JavaScript [www-dokumentti] [Viitattu 24.10.2010]
Saatavissa: https://developer.mozilla.org/en/About_JavaScript
- /8/ Forum Nokia [www-dokumentti] [Viitattu 28.10.2010]
Saatavissa: <http://www.forum.nokia.com/>
- /9/ Maemo.org [www-dokumentti] [Viitattu 10.11.2010]
Saatavissa <http://talk.maemo.org/showthread.php?t=62858>