

Tekninen määrittely

Nappula Oy

versio 1.0

|  |  |  |
| --- | --- | --- |
| Karelia | TITE/TIKO | Ohjelmistojen määrittely ja suunnittelu LTP7030 |
| Tekijä: Jaakko Räsänen (1500911)  Jussi Hukkanen (1500646) | | Tulostettu: 44.3.2017 |
| Jakelu: Petri Laitinen | | |
| Dokumentin tila: valmis | | Muokattu: 24.3.2017 |

Versiohistoria

|  |  |  |  |
| --- | --- | --- | --- |
| Versio | Päiväys | Tekijät | Selite (muutokset, korjaukset...) |
|  |  |  |  |
| 0.1 | 14.03.2017 | Jaakko Räsänen | Aloitettu dokumentin työstäminen. Dokumentin pohjan valmistelu. |
|  |  |  |  |
| 0.2  0.3  1.0 | 17.03.2017  22.03.2017  24.03.2017 | Jussi Hukkanen  Jussi Hukkanen  Jaakko Räsänen | Kieliasun parantelua  Dokumentin työstäminen. Kuvien ja kaavioiden lisäys.  Dokumentin työstäminen. Dokumentin viimeistely |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sisällysluettelo

1. JOHDANTO 4

1.1 Tarkoitus 4

1.2 Kattavuus 4

1.3 Määritelmät, termit ja lyhenteet 4

1.4 Yleiskatsaus dokumenttiin 4

2. Järjestelmän yleiskuvaus 5

2.1 Yleiskuvaus 5

2.2 Laitteistokuvaus 5

2.3 Ohjelmistokuvaus 5

2.4 Reunaehdot 5

2.5 Järjestelmän liittyminen ympäristöön 5

3. Arkkitehtuurin kuvaus 7

3.1 Ratkaisuperiaatteet 7

3.2 Tietokanta-arkkitehtuuri 7

3.3 Ohjelmistoarkkitehtuuri, moduulit ja prosessit 7

3.4 Uudelleenkäytettävät komponentit 7

3.5 Komponenttikaavio 8

3.6 Sekvenssikaaviot 9

3.6.1 Käyttötapausesimerkki 1: Tuotteen hinnan päivittäminen 9

3.6.2 Käyttötapausesimerkki 2: Hintalapussa näkyvän hinnan tarkistaminen 9

4. Moduuli- ja prosessikuvaukset 10

4.1 LCD-näyttö ja Arduino Uno 10

4.2 Raspberry Pi 3 10

4.3 Microsoft Dynamics NAV 11

5. Siirrettävyys 12

6. Virhetilanteiden käsittely 13

7. testattavuus 14

# JOHDANTO

## Tarkoitus

Tämän projektin tarkoituksena on tuottaa toimiva, automatisoitu hintalappujärjestelmä haluttuun kohteeseen. Järjestelmästä on tarkoitus tehdä mahdollisimman helposti uudelleenkäytettävä.

## Kattavuus

Tämä dokumentti käsittelee kyseisen järjestelmän teknistä toteutusta.

## Määritelmät, termit ja lyhenteet

Ks. Vaatimusmäärittely -dokumentti, kappale 2

## Yleiskatsaus dokumenttiin

Dokumentti esittelee aluksi järjestelmän yleisellä tasolla. Tämän jälkeen dokumentissa puretaan järjestelmä osiin ja esitellään kukin näistä osista erikseen.

# Järjestelmän yleiskuvaus

## Yleiskuvaus

Järjestelmän tarkoitus on automatisoida kohdekaupan hintalappujen päivitys. Hintalappuihin päivitettävä data saadaan asiakkaan omasta Microsoft Dynamics NAV - toiminnanohjausjärjestelmästä.

## Laitteistokuvaus

Alla on lueteltu järjestelmään kuuluvat laitteet. Tarvittava laitteiden määrä on riippuvainen kaupan koosta:

* LCD-hintalappu \* X
* Arduino Uno -mikro-ohjain \* Y
* Raspberry Pi 3 -korttitietokone \* Z
* langaton XBee-vastaanotin/lähetin \* P

Muut laitteet ovat asiakkaan omia.

## Ohjelmistokuvaus

Arduino Unon ohjelmointiin käytetään sen omaa Arduino IDE -alustaa.

Raspberry Pi 3:ssa pyörii sen oma käyttöjärjestelmä. Raspberry Pihin kirjoitettava koodi tehdään Python 3 -kielellä.

Microsoft Dynamics NAV on asiakkaan oma järjestelmä ja siihen kytkeydytään SOAP-rajapinnasta.

## Reunaehdot

* Järjestelmän tulee pystyä päivittämään hinnat vähintään kerran päivässä, sellaiseen kellonaikaan kun kauppa on suljettu.
* Hintojen tulee täsmätä Asiakkaan tietokannan hintoihin.

## Järjestelmän liittyminen ympäristöön

Laitteet itsessään liitetään ympäristöön tarvittavilla kiskoilla ja telineillä. Niiden asennukseen käytetään hakoja ja ruuveja.

Laitteet ovat keskenään yhteydessä XBee-lähettimillä.

Raspberry Pi 3 on luodun järjestelmän sisällä ainoa laite, joka on yhteydessä asiakkaan järjestelmään. Raspberry Pi on yhteydessä Microsoft Dynamics NAV:n SOAP-rajapintaan.

# Arkkitehtuurin kuvaus

## Ratkaisuperiaatteet

Tässä kappaleessa esiteltyihin ratkaisuihin on päädytty sillä perusteella että ne ovat projektin jäsenille tuttuja ennalta ja ne on todettu käyttökelpoisiksi.

## Tietokanta-arkkitehtuuri

Tietokantana käytetään asiakkaan omaa, valmista tietokantaa.

## Ohjelmistoarkkitehtuuri, moduulit ja prosessit

Ohjelmistoarkkitehtuuri on kuvattu karkeasti kappaleessa 3.5 komponenttikaavion avulla. Toteutus on jaettu kolmeen eri moduuliin.

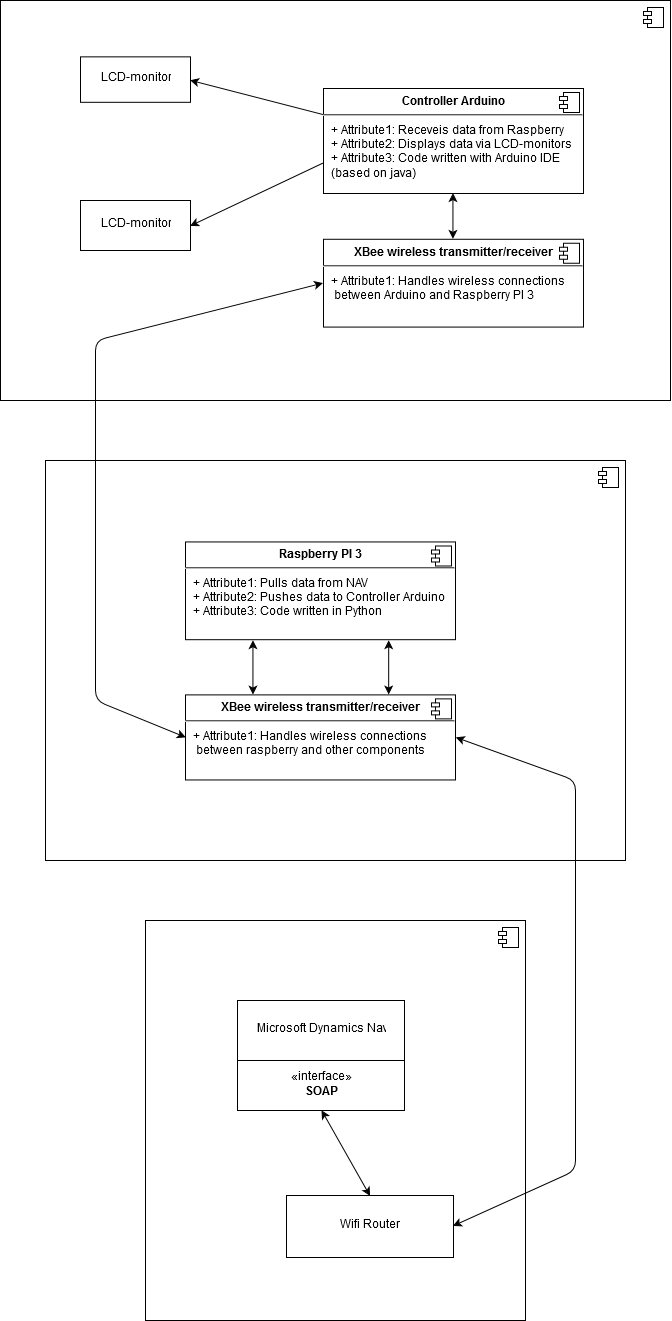
Ensimmäinen moduuli käsittelee LCD – hintalappunäyttöjä ja niille datan syöttävää Arduinoa. Arduinon lähettimenä/vastaanottimena toimii Xbee- langaton lähetin/vastaanotin.

Toinen moduuli käsittää Raspberry PI 3:n ja sen lähettimeksi valitun Xbee- lähettimen. Raspberryn tarkoitus on välittää data Arduinoille ja noutaa data Microsoft Dynamic NAV:ista.

## Uudelleenkäytettävät komponentit

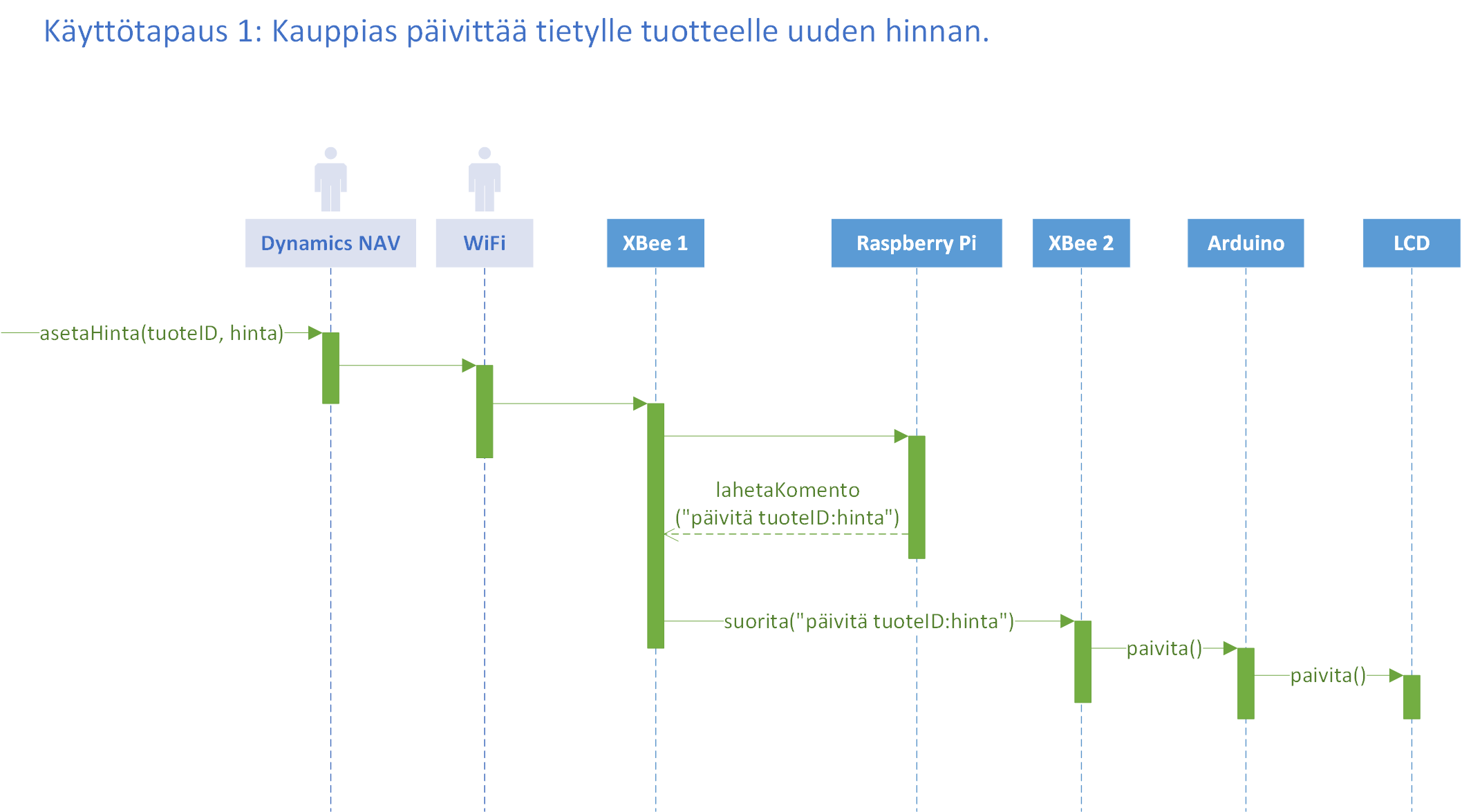
Kaikki komponentit ovat sellaisenaan uudelleenkäytettäviä, mikäli toiminnanohjausjärjestelmänä on Microsoft Dynamics NAV. Jos toiminnanohjausjärjestelmä muuttuu, tulee myös Raspberry Pin koodi päivittää.

## Komponenttikaavio

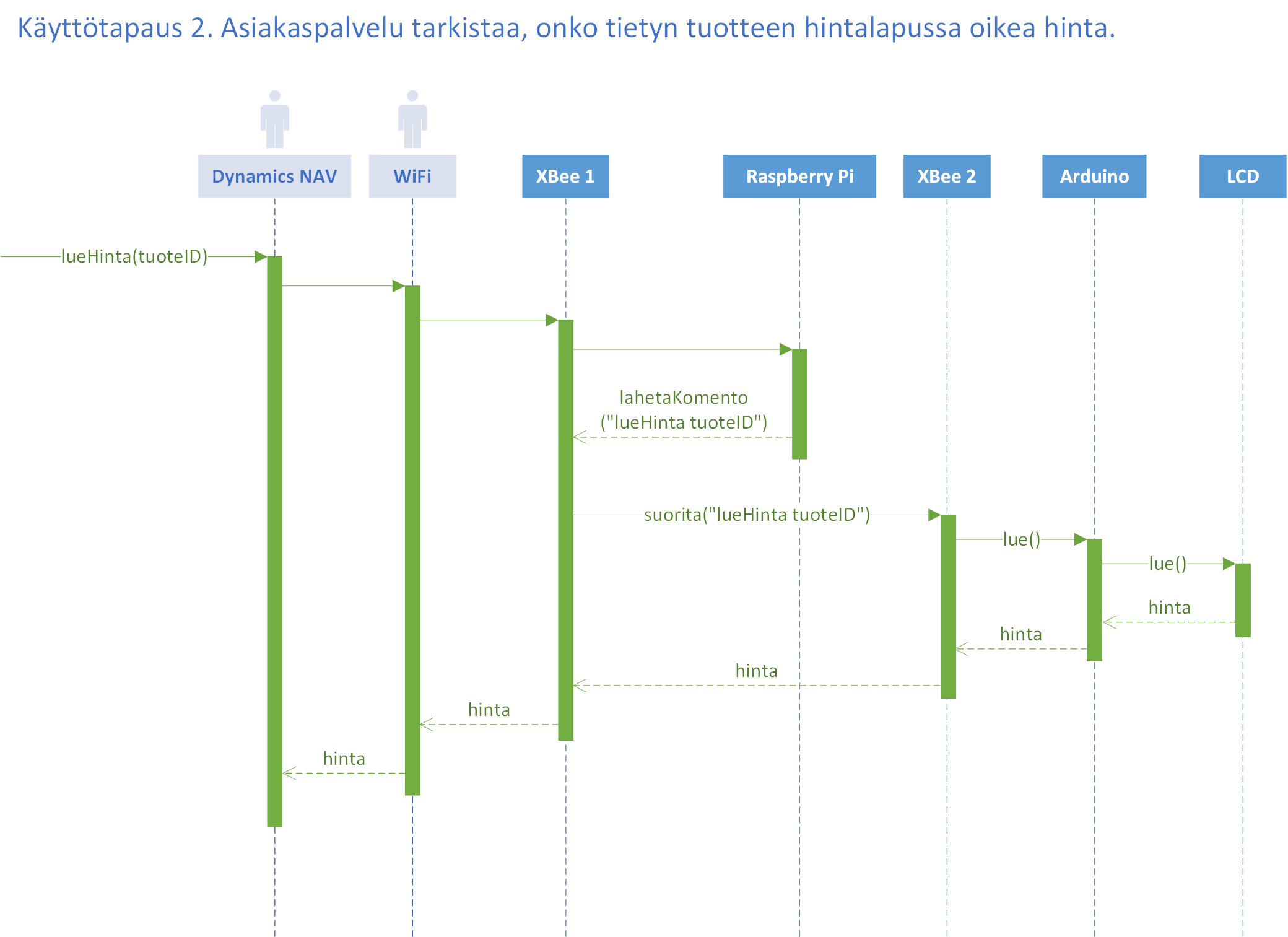


## Sekvenssikaaviot

### Käyttötapausesimerkki 1: Tuotteen hinnan päivittäminen



### Käyttötapausesimerkki 2: Hintalapussa näkyvän hinnan tarkistaminen



# Moduuli- ja prosessikuvaukset

## LCD-näyttö ja Arduino Uno

Tämä moduuli on tarkoitettu vastaanottamaan dataa Raspberry Pi:ltä. Arduino Uno on moduulin keskipisteenä ja siihen liitetään suoraan kaikki moduulin muut osat (lähetin ja näyttö). Kun uusi hinnan päivitys tulee Raspberry Pi:ltä, toimii moduuli seuraavalla tavalla:

* Uusi hinta lähetetään Raspberry Pi:ltä
* Lähetys vastaanotetaan Xbee- vastaanottimella
* Data otetaan vastaan esim. datanvastaanotto( )- funktiolla
* Vastaanotettu data tallennetaan muuttujaan *hinta*
* Muuttuja *hinta* laitetaan näkymään LCD – näytölle funktion paivita( ) – kautta.
* kuittaus( ) – funktio lähettää kuittausviestin Raspberry Pi:lle mikäli prosessi on suoritettu onnistuneesti loppuun. (Mikäli on tapahtunut virhe, napataan se kiinni ja lähetetään virheviesti Raspberry Pi:lle.)

Näin hinta on päivitetty Arduinoon ja sen kautta LCD – näyttöön.

## Raspberry Pi 3

Tämä moduuli on koko järjestelmän tärkein osa. Se noutaa dataa Microsoft Dynamics NAV:in tietokannasta ja lähettää sen edelleen Arduinoille, joiden kautta hinnat saadaan näkyville. Datan nouto ja päivitys prosessi toimii automaattisesti tai se voidaan ajaa manuaalisesti. Automaattinen päivitysprosessi toimii seuraavasti:

* Raspberry lähettää datapyynnön NAV:iin SOAP – rajapinnan välityksellä.
* Päivittyneet hinnat lähetetään Raspberrylle.
* Hinnat vastaanotetaan datanvastaanotto( )– funktiolla.
* Hinnat lähetetään eteenpäin suorita(”päivitä tuoteID:hinta”) – funktiolla.
* Arduinolta jäädään odottamaan kuittausta.
* Mikäli kuittausta ei tule aikarajan sisällä tai Arduino ilmoittaa virheestä, kirjataan tapaus lokiin.

Manuaalinen päivitysprosessi toimii muuten samoin, mutta prosessin aloitus ei ole automatisoitu. Manuaalisessa päivityksessä voidaan päättää, mitkä hinnat päivitetään. Manuaalinen päivitys on kuvattu kappaleessa 3.6.1.

## Microsoft Dynamics NAV

Tämä moduuli on Asiakkaan oma, jo olemassa oleva toiminnanohjausjärjestelmä. Tähän moduuliin ollaan yhteydessä sen SOAP – rajapinnan kautta ja se lähettää päivittyneitä hintoja Raspberry Pi:lle, jos päivitys pyyntö/komento vastaanotetaan.

# Siirrettävyys

Järjestelmä voidaan siirtää sellaisenaan uuteen käyttökohteeseen, mikäli toiminnanohjausjärjestelmänä myös uudessa kohteessa on Microsoft Dynamics NAV. Uudessa kohteessa tulee kuitenkin huomioida hintalappujen sijoittelu, jotta oikeat hinnat näkyvät oikeassa paikassa.

# Virhetilanteiden käsittely

Virhetilanteissa tallennetaan epäonnistunut toiminto lokiin ja annetaan ilmoitus käyttöliittymään.

Ohjelma yrittää automaattisesti suorittaa epäonnistuneen toiminnon kertaalleen, jonka jälkeen vaaditaan manuaalista korjausta.

Mikäli päivitys katkeaa odottamattomasti, raportoidaan päivittämättömät tuotteet lokiin.

Sähkökatkotilanteissa suositellaan järjestelmän käynnistämistä uudelleen virhetilanteiden välttämiseksi. Tämän jälkeen suositellaan manuaalista päivitystä, varsinkin jos katko on sattunut päivityksen hetkellä.

# testattavuus

Järjestelmä voidaan testata noutamalla dataa tietokannasta ja tarkistamalla, näkyykö oikea data oikean tuotteen kohdalla.