

LAPPEENRANNAN TEKNILLINEN YLIOPISTO
TIETOTEKNIKAN OSASTO
TIETOLIIKENNETEKNIKAN LABORATORIO

VIIVAKOODIN LUKEMINEN MATKAPUHELIMELLA

KANDIDAATINTYÖ
LOPPURAPORTTI

Seminaarityön aihe on hyväksytty 27.9.2007

Ohjaaja: DI Tommi Kallonen

Valvoja: TkT Jari Porras

Lappeenranta, 28. Joulukuuta 2007

Juho Vartiainen

juho.vartiainen@lut.fi

ABSTRACT

Lappeenranta University of Technology

Department of Information Technology

Juho Vartiainen

Viivakoodin lukeminen matkapuhelimella

Bachelor's Thesis

2007

28 pages, 27 figures, 2 tables

Supervisor: DI Tommi Kallonen

Keywords: barcode, reading, mobile phone, camera

Purpose of this work was to study different types of barcodes and possibilities of reading them with mobile phones camera. Another goal was to transfer product information from a server to a mobile phone according to a barcode.

There is existing software to read barcodes with mobile phones camera. They also open a www – page to mobile phones browser according to a URL address interpreted from a barcode.

I simulated a use-case scenario, in which product delivery related information was controlled by a mobile phone. The information was located in a MySQL database and it was controlled with www – pages through mobile phones browser. I also made tests about how possible difficulties effect on successful interpretation of a barcode.

TIIVISTELMÄ

Lappeenrannan Teknillinen Yliopisto

Tietotekniikan osasto

Juho Vartiainen

Viivakoodin lukeminen matkapuhelimella

Kandidaatintyön loppuraportti

2007

28 sivua, 27 kuvaa, 2 taulukkoa

Ohjaaja: DI Tommi Kallonen

Hakusanat: viivakoodi, lukeminen, matkapuhelin, kamera

Tämän työn tavoitteena oli tutkia viivakoodien ominaisuuksia sekä niiden lukumahdollisuuksia matkapuhelimen kameran avulla. Tämän lisäksi luetun viivakoodin perusteella oli tarkoitus siirtää tuotetietoja palvelimelta matkapuhelimeen.

Viivakoodin lukemiseen matkapuhelimella löytyy valmiita sovelluksia, jotka avaavat viivakoodista tulkitun URL osoitteen perusteella www – sivun matkapuhelimen selaimen.

Toteutin työssä käyttötapauksen, jossa tuotteen toimitukseen liittyviä tietoja hallitaan matkapuhelimella. Toimituksen tiedot sijaitsevat MySQL tietokannassa ja niitä hallitaan www – sivujen avulla matkapuhelimen selaimesta. Lisäksi tein testauksia lukutapahtuman virheensiedosta simuloiden mahdollisia haittatekijöitä.

SISÄLLYSLUETTELO

1 JOHDANTO.....	5
2 TEORIAOSUUS	6
2.1 1D VIIVAKOODITYYPIT.....	6
2.2 2D VIIVAKOODITYYPIT	13
2.3 VIIVAKOODIEN LUKEMINEN MATKAPUHELIMEN KAMERALLA	16
2.4 TIEDON SIIRTÄMINEN TIETOKANNASTA MATKAPUHELIMEEN.....	17
3 KÄYTÄNNÖN OSUUS.....	19
3.1 TYÖN KUVAUS	19
3.2 TIETOKANNAN KUVAUS.....	19
3.3 LUKUTESTIEN KUVAUS.....	20
3.4 LUKUTESTIEN TULOKSET	21
3.5 KÄYTTÖTAPAUS: TOIMITUKSEN SEURAAMINEN.....	22
4 YHTEENVETO	26
LÄHDELUETTELO	27

LYHENNELUETTELO

EAN European Article Numbering

EM Expectation-Maximization

HIBC Health Care Industry Barcode

HTML HyperText Markup Language

ISBN International Standard Book Numbering

ISSN International Standard Serial Numbering

J2ME Java 2 Platform, Micro Edition

PHP Hypertext Preprocessor

RSS Reduced Space Symbology

SCC Shipping Container Code

SSCC Serialized Shipping Container Code

UPC Universal Product Code

URL Uniform Resource Locator

XML Extensible Markup Language

1 JOHDANTO

Tässä työssä keskityn viivakoodien avulla yksilöityjen tuotteiden tietojen hallintaan matkapuhelimen avulla. Viivakoodeja olisi tarkoitus tulkita matkapuhelimen kameran avulla, ja tuotetiedot tulisi saada siirrettyä tulkitun koodin perusteella tietokannasta matkapuhelimeen.

Työn tavoitteena on aluksi selvittää erilaisten viivakoodityyppien ominaisuudet ja niiden lukumahdollisuudet matkapuhelimen kameran avulla. Tämän jälkeen on selvitettävä miten tulkitun viivakoodin perusteella saadaan siirrettyä tuotteeseen liittyviä tietoja palvelimelta matkapuhelimeen. Tietoja tulee myös olla mahdollista päivittää matkapuhelimen avulla.

Tulen myös tekemään käytännön testauksia lukutapahtuman virheensiedosta simuloiden mahdollisia tapauksia joissa viivakoodi voi olla jollain tapaa vääristynyt.

Tämä työ on osa Mobilding – hanketta, jonka tavoitteena on selvittää millä menetelmillä voidaan hallita rakennukseen liittyvät tiedot koko sen elinkaaren aikana. Hallinnassa on tarkoitus hyödyntää nykyaikaista mobiilitekniikkaa vastaamaan eri toimijoiden asettamiin tarpeisiin tiedolle. [1]

Työn Osio 2 sisältää katsauksen eri viivakoodityyppeihin, niiden lukemiseen matkapuhelimen kameralla, sekä tiedon siirtämiseen tietokannasta matkapuhelimeen. Osio 3 sisältää työn käytännön osuuden, eli työn ja testien kuvaukset sekä tulokset. Osio 4 sisältää yhteenvedon työstä, sekä pohdintaa sen onnistumisesta ja tulosten merkityksestä.

2 TEORIAOSUUS

Viivakoodien tarkoituksena on yksilöidä tuotteita. Ne eivät tyypillisesti sisällä informaatiota itse tuotteesta, vaan pelkästään identifioivan numerosarjan, jonka avulla voidaan hakea tietoa tuotteesta. [2]

Viivakoodit sisältävät yleensä tarkistussumman, jonka avulla saadaan selville jos viivakoodin sisältö on korruptoitunut. 1D viivakoodissa tarkistussumma on usein yhden merkin mittainen, ja se lasketaan viivakoodin sisältämien merkkien ja niiden paikkojen perusteella. 2D viivakoodit pystyvät sisältämään huomattavasti suuremman määrän dataa kuin 1D viivakoodit, ja sisältävät usein virheentunnistus- sekä virheenkorjausalgoritmeja.

2.1 1D VIIVAKOODITYYPIT

1D viivakoodit koostuvat eri levyisistä pystysuorista viivoista sekä väleistä. Näitä yhdessä kutsutaan ”elementeiksi”, ja ne niiden yhdistelmät edustavat tiettyjä merkkejä [2].

Seuraavaksi käyn läpi 1D viivakoodistandardeja pääpiirteittäin.

Code 39 – viivakoodityyppi on erittäin laajasti käytetty. Tässä standardissa jokaista merkkiä esittää 5 viivaa ja 4 väliä. Kolme näistä on aina leveitä ja kuusi kapeita. Jokainen sarja alkaa ja loppuu tietyillä aloitus/lopetus – merkeillä. Code 39 – viivakoodityyppi on esitetty kuvassa 1. [3]



Kuva 1: Code 39 [3]

Code 128 – viivakoodityyppi tukee kolmea eri merkistöä, jotka kattavat kaikki 128 ASCII merkkiä pienin poikkeuksin. Koodi sisältää myös tarkistussumman, joka lasketaan summaamalla ensimmäinen merkki jokaisen merkin ja niiden paikan tulolla, ja ottamalla saadusta luvusta jakojäännös luvun 103 kanssa. Code 128 – viivakoodityyppi on esitetty kuvassa 2. [3]



Kuva 2: Code 128 [3]

Code 93 – viivakoodityyppi on korkeatiheyksinen komplementti Code 39 – viivakoodityypille. Koodi alkaa ja loppuu asteriskiinkin sekä sisältää tarkistussumman. Code 93 – viivakoodityyppi on esitetty kuvassa 3. [3]



Kuva 3: Code 93 [3]

EAN 13 – viivakoodityyppiä käytetään maailmanlaajuisesti merkitsemään jälleenmyyntituotteita. Koodi koostuu neljästä osasta. Ensimmäinen osa on kahden tai kolmen numeron mittainen ja se identifioi koodin myöntäneen auktoriteetin. Toinen osa on viisi numeroa pitkä ja identifioi tuottajan. Kolmas osa on myös viisi numeroa pitkä ja identifioi tuotteen. Viimeisenä osana on yhden numeron mittainen tarkistussumma. EAN 13 – viivakoodityyppi on esitetty kuvassa 4. [3]



Kuva 4: EAN 13 [3]

EAN 8 – viivakoodityyppi on lyhennetty versio EAN 13 – viivakoodityypistä. Se sisältää kaksi ilmaisinnumeroa, viisi datanumeroa ja yhden tarkistussummanumeron. EAN 8 – viivakoodityyppi on esitetty kuvassa 5. [3]



Kuva 5: EAN 8 [3]

I2OF5 – viivakoodityyppiä käytetään pääasiallisesti jakelu- ja varastoteollisuudessa. Koodissa käytetään sekä viivojen että välien leveyttä merkkien esittämiseen. Koodista on olemassa myös versio S2OF5, joka käyttää merkkien esittämiseen pelkästään viivojen leveyksiä. I2OF5 – viivakoodityyppi on esitetty kuvassa 6. [3]



Kuva 6: I2OF5 [3]

ISBN ja ISBN 13 – viivakoodityyppiä käytetään maailmanlaajuisesti identifioimaan kirjoja. Nämä ovat käytännössä samanlaisia kuin EAN 13 – viivakoodityyppi. Samankaltainen on myös julkaisuja identifioiva ISSN – viivakoodityyppi. [3]

ITF14 – viivakoodityyppi perustuu samaan merkkien esittämistapaan kuin I2OF5 – viivakoodityyppi. Sitä käytetään merkitsemään kartonkeja, laatikkoja ja lavoja, joilla on UPC tai EAN identifikaationumero. Koodin ylä- ja alapuolella on nk. kantoviivat (Bearer bars), joiden avulla ehkäistään osittaiset lukemiset. ITF14 – viivakoodityyppi on esitetty kuvassa 7. [3]



Kuva 7: ITF14 [3]

MSI – viivakoodityyppi on MSI Data Corporationin kehittämä ja sen pääasiallinen käyttökohde on jälleenmyyntiyritysten hyllyt inventaariota varten. Koodi sisältää aloitus- sekä lopetus merkit, minkä tahansa määrän sisältö merkkejä sekä tarkistussumman. MSI – viivakoodityyppi on esitetty kuvassa 8. [3]



Kuva 8: MSI [3]

Postnet – viivakoodityyppi on Yhdysvaltojen postilaitoksen käyttämä koodi, joka käyttää viittä viivaa ja neljää väliä. Postnet - viivakoodityyppi on esitetty kuvassa 9. [3]



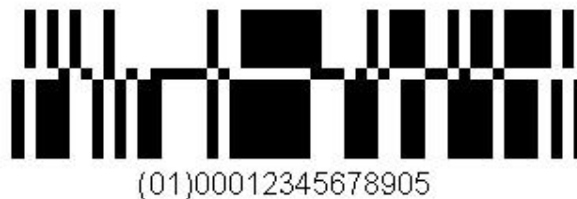
Kuva 9: Postnet [3]

Rational Codabar – viivakoodityyppi pystyy esittämään 16 eri merkkiä sekä 4 aloitus- ja lopetusmerkkiä. Rational Codabar – viivakoodityyppi on esitetty kuvassa 10, ja sitä käyttävät Yhdysvaltojen veripankki, valokuva-laboratoriot sekä FedEx.



Kuva 10: Rational Codabar [3]

RSS – viivakoodityyppiä käytetään yleensä elintarvike- sekä terveydenhoitoaloilla pienuutensa takia. RSS – viivakoodityyppiä ovat RSS 14, RSS Limited, RSS Stacked sekä RSS Truncated. Nämä eroavat hieman ominaisuuksiltaan, esimerkiksi kuvassa 11 esitetty RSS Stacked – viivakoodityyppi voidaan tulostaa kaksiriviseksi. [3]



Kuva 11: RSS Stacked [3]

SCC 14 – viivakoodityyppiä käytetään laivauskonteissa ja sitä esitetään yleisimmin kuvassa 12 esitetyllä UCCEAN 128 – formaatilla. [3]



Kuva 12: UCCEAN 128 [3]

SSCC 18 – viivakoodityyppi käyttää UCC/EAN-128 merkistöä, ja sitä käytetään koko toimitusketjussa tavaroiden seuranta varten. SSCC 18 – viivakoodityyppi on esitetty kuvassa 13. [3]



Kuva 13: SSCC 18 [3]

UCC/EAN 128 – viivakoodityyppi kehitettiin tarjoamaan maailmanlaajuinen formaatti. Koodin etuosassa on kaksinumeroinen applikaatiotunnus. Kuvassa 13 esitetyssä SSCC 18 – tyyppin viivakoodissa applikaatiotunnus on 00.

UPC – viivakoodityypit sisältävät sekä tavan koodata että merkistön. UPC – viivakoodityyppejä ovat UPCA, UPCE, UPCEXT2 sekä UPCEXT5. UPCA ja UPCE – viivakoodityyppejä käytetään esittämään sekä tuottaja että tuote. UPCEXT2 – viivakoodityyppejä käytetään ylimääräisenä viivakoodina esittämään lehden numero. Tällä tavoin lehden tuotekoodi voi pysyä samana. UPCEXT5 – viivakoodityyppejä käytetään myös ylimääräisenä ja sillä esitetään suositeltu jälleenmyyntihinta. UPCA – viivakoodityyppi on esitetty kuvassa 14, UPCE – viivakoodityyppi on esitetty kuvassa 15, UPCEXT2 – viivakoodityyppi on esitetty kuvassa 16 ja UPCEXT5 – viivakoodityyppi on esitetty kuvassa 17. [3]



Kuva 14: UPCA [3]



Kuva 15: UPCE [3]



Kuva 16: UPCEXT2 [3]



Kuva 17: UPCEXT5 [3]

HIBC – viivakoodityyppi on kehitetty erityisesti terveydenhoitoalaa varten. Terveydenhoitoalan tuotteet vaativat mahdollisimman virheettömän tavan merkitä tuotteita koska kyseessä on ihmisten turvallisuus. HIBC – viivakoodityyppi on esitetty kuvassa 18. [3]



Kuva 18: HIBC [3]

2.2 2D VIIVAKOODITYYPIT

1D viivakoodien hyödyllisyyttä datan siirtämisessä rajoittaa niiden pieni kapasiteetti [4]. Nykyään yhä useammat applikaatiot vaativat pidempiä viivakoodeja koodatakseen suuremman määrän dataa. Tätä varten suunniteltiin 2D viivakoodit, jotka pystyvät sisältämään huomattavasti enemmän tietoa kuin 1D viivakoodit. [5]

Seuraavaksi esittelen 2D viivakoodityyppejä sekä niiden ominaisuuksia.

PDF417 – viivakoodityyppi 17 moduulista, joissa jokaisessa on 4 viivaa sekä väliä. Viivakoodi käyttää Reed-Solomon virheenkorjausta, joka estää pieniä vaurioita hävittämästä dataa. PDF417 – viivakoodityyppi on esitetty kuvassa 20. [3]



Kuva 20: PDF417 [3]

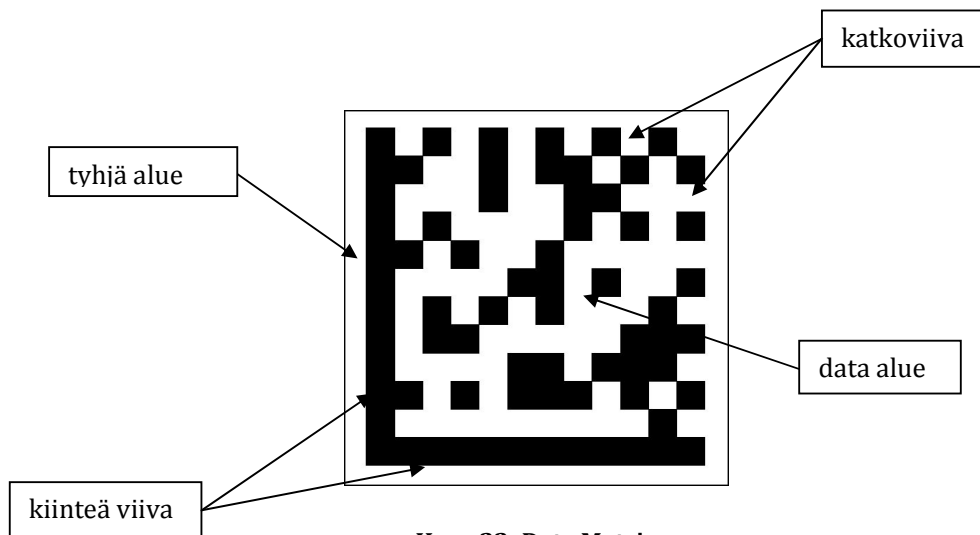
QR Code – viivakoodityyppi suunniteltiin ajatuksena nopea lukeminen. Viivakoodin etsintäkuvio koostuu kolmesta neliöstä, joissa on vuoronperään sekä mustaa että valkoista. Nämä neliöt mahdollistavat nopean ja tarkan lukemisen. QR Code – viivakoodityyppi on esitetty kuvassa 21. [3]



Kuva 21: QR Code [3]

Data Matrix – viivakoodityyppi koostuu neliön tai suorakaiteen muotoisista moduuleista [3]. Viivakoodin sisältö on sijoitettu etsintäkuvion sisään, joka koostuu kiinteästä viivasta, katkoviivasta sekä tyhjistä alueista viivakoodin ympärillä [6]. Data Matrix – viivakoodeja on olemassa kahdella erityyppisellä virheenkorjauksella, konvoluutiokoodilla sekä Reed-Solomon virheenkorjauksella. Data Matrix – viivakoodityyppiä käytetään laajalti sovelluksissa joissa kameralla varustettu matkapuhelin ohjataan koodin avulla URL osoitteeseen. 12*12 pikselin kokoinen Data Matrix viivakoodi on esitetty kuvassa 22. [3]

Reed-Solomon virheenkorjausmenetelmä perustuu polynomiyhtälöiden avulla laskettaviin tarkistussummiin. ECC200 versio Data Matrix – viivakoodityypistä sisältää virheentunnistus sekä virheenkorjaus algoritmeja, joiden avulla viivakoodi on mahdollista tulkita jopa 60% vaurioitumiseen saakka. Data Matrix viivakoodityypin ECC200 version koon suhdetta datamäärään ja virheenkorjaukseen esittää taulukko 1. Taulukko kertoo viivakoodin koon sekä korjattavien virheiden enimmäismäärän suhteen lisäksi enimmäismäärät merkkejä, joita viivakoodi pystyy sisältämään eri koodaus tavoilla. [7]



Kuva 22: Data Matrix

Taulukko 1: Data Matrix viivakoodin koon suhde dataan ja korjattaviin virheisiin

Koko (pikseliä)	enimmäismäärä numeroita	enimmäismäärä kirjaimia sekä numeroita	enimmäismäärä binäärilukuja	enimmäismäärä korjattavia virheitä
10x10	6	3	1	2
12x12	10	6	3	3
14x14	16	10	6	5
16x16	24	16	10	6
18x18	36	25	16	7
20x20	44	31	20	9
22x22	60	43	28	10
24x24	72	52	34	12
26x26	88	64	42	14
32x32	124	91	60	18
36x36	172	127	84	21
40x40	228	169	112	24
44x44	288	214	142	28
48x48	348	259	172	34
52x52	408	304	202	42
64x64	560	418	278	56
72x72	736	550	366	72
80x80	912	682	454	96
88x88	1152	862	574	112
96x96	1392	1042	694	136
104x104	1632	1222	814	168
120x120	2100	1573	1048	204
132x132	2608	1954	1302	248
144x144	3116	2335	1556	310

2.3 VIIVAKOODIEN LUKEMINEN MATKAPUHELIMEN KAMERALLA

Tekniikat 1D viivakoodien tunnistamiseen perustuvat perinteisesti toisiinsa derivoituihin, huippujen sijainteihin, valikoivaan otantaan sekä EM algoritmiin. Nämä tekniikat ovat kuitenkin suunniteltu suuren resoluution omaaville lukijalaitteilla, eivätkä ole välttämättä toimivia matkapuhelimen kameraa käytettäessä. Wang, Zou ja Wang esittelevät artikkelissaan [8] metodin, joka perustuu kuviodien tunnistamiseen niiden tilastollisten ominaisuuksien perusteella. He käyttivät testeissään EAN 13 – viivakoodityyppejä. Tilastollisten ominaisuuksien kartoittamiseen he käyttivät yli tuhatta kuvaa viivakoodeista. Kuvat olivat otettu Nokian puhelinmallilla 3650. Testien tuloksena he saivat tunnistettua onnistuneesti 250 symbolia 292:sta.

Suurin osa nykyisistä kameralla varustetuista matkapuhelimista tukee J2ME – sovellusympäristöä [9], jonka avulla voidaan käsitellä otettuja kuvia. Kahani ja Falas esittelevät artikkelissaan [6] J2ME – sovellusympäristöä hyväksikäyttävän tavan tulkita 2D viivakoodeja matkapuhelimen kameralla otetusta kuvasta. Menetelmä sisältää useita esiprosessointi askelia, joiden jälkeen viivakoodi on helpompi tulkita kuvasta. Dekoodaus vaiheessa menetelmä etsii aluksi viivakoodin etsintäkuvion, jonka jälkeen se tulkitsee viivakoodin data osuuden.

Viivakoodin tulkintaa matkapuhelimen kameralla otetusta kuvasta saattaa vaikeuttaa ääriään myötä aiheuttama vääristyminen. Chu, Yang ja Chen esittelevät artikkelissaan [10] algoritmin, joka helpottaa 2D viivakoodin tulkitsemista ääriään myötä vääristämästä kuvasta. He myös mainitsivat aikovansa kääntää menetelmän eri alustoille, kuten Nokian käyttämille Symbian pohjaisille järjestelmille.

2D viivakoodeja esittelevässä osiossa tuli esiin että Data Matrix – viivakoodityyppi on suosittu matkapuhelimen kameraa käyttävissä sovelluksissa, joissa käyttäjä ohjataan URL osoitteeseen. Data Matrix – viivakoodityyppi omaa myös 2D koodeista parhaimman data-kapasiteetti/koko – suhteen. [6]

Artikkeli [11] vertailee Data Matrix sekä QR Code - viivakoodityyppejä mobiileissa sovelluksissa käytettäväksi. Vertailun lopputulos oli että Data Matrix on parempi,

perusteina myös artikkelissa [6] esille tulleet laajempi käyttö sekä parempi data-kapasiteetti/koko – suhde.

Tutkimukset ovat osoittaneet että puhelimen kameran resoluutiolla ei ole juurikaan merkitystä viivakoodien luettavuuteen, mutta tarkennuksella on. Hyvin paljon yleistynyt automaattinen tarkennus takaa hyvän luettavuuden, mutta sen pitää olla ohjelmallisesti ohjattavissa. [6]

Viivakoodien lukemiseen matkapuhelimen kameran avulla löytyy valmiita sovelluksia, kuten UpCode[12] ja MobileTag[13]. Molemmat tarjoavat mahdollisuuden lukea viivakoodeja matkapuhelimen kameran avulla tai vaihtoehtoisesti syöttämällä viivakoodin numeron puhelimen näppäimistöllä, olettaen että numero löytyy viivakoodin yhteydestä. UpCoden sovellus pystyy tulkitsemaan Data Matrix-, QR Code- sekä joitain 1D viivakoodityyppejä. Mobiletag sovellus käyttää Data Matrix – viivakoodityyppiä. Sovellukset tarjoavat erilaisia mahdollisuuksia käyttää viivakoodista tulkittua tietoa, kuten ohjata matkapuhelimen selaimen tulkittuun URL osoitteeseen tai lisätä viivakoodin sisältämä yhteystieto matkapuhelimen osoitekirjaan.

2.4 TIEDON SIIRTÄMINEN TIETOKANNASTA MATKAPUHELIMEEN

Vaihtoehtoina tiedonsiirtämiseen tietokannasta matkapuhelimeen on tehdä oma asiakasohjelma tai www – sivu, jonka avulla tietoja hallitaan. On myös olemassa sovelluksia, kuten DataX[14], jonka toiminta perustuu tiedon siirtämiseen XML – muodossa välityspalvelimen avulla. DataX sovelluksen käyttäminen vaatisi ainakin erillisen välityspalvelimen hankkimisen.

Oman asiakasohjelman tekeminen vaatisi Symbian tai Java ohjelmointitaitoja. Matkapuhelimen puolella asiakasohjelma pitäisi saada tulkitsemaan viivakoodi, esimerkiksi artikkelissa [6] esitetyllä tavalla J2ME – sovellusympäristöä hyväksikäyttäen. Tämän jälkeen matkapuhelimen pitäisi muodostaa yhteys palvelimen puolella sijaitsevaan ohjelmaan, joka toimisi rajapintana palvelimella sijaitsevaan tietokantaan. Tämä tapa

kuluttaisi matkapuhelimen resursseja koska asiakasohjelma joutuisi käsittelemään pyyntöjä sekä tulkitsemaan ja esittämään dataa.

Käytettäessä valmista sovellusta joka ohjaa matkapuhelimen selaimen viivakoodin osoittamaan URL osoitteeseen, on ainoana järkevänä mahdollisuutena tehdä www – sivu jonka avulla tietoja hallitaan. Tässä lähestymistavassa tietokannan sisältävä palvelin tekisi prosessoinnin ja matkapuhelimen tarvitsisi vain näyttää www – sivu. Sivun voi toteuttaa PHP – kielellä, josta löytyy valmiita funktioita tietokannan hallintaa varten.

3 KÄYTÄNNÖN OSUUS

3.1 TYÖN KUVAUS

Työn tarkoituksena oli aluksi tutkia eri viivakoodeja, niiden ominaisuuksia sekä lukumahdollisuuksia matkapuhelimen kameralla. Tämä tapahtui työn teoriaosuudessa. Toisena tavoitteena oli toteuttaa menetelmä, jolla tuotteeseen liittyviä tietoja voidaan hallita matkapuhelimen avulla. Päädyin käyttämään työssä UpCoden ohjelmistoa, jolla viivakoodi voidaan liittää URL osoitteeseen. Kun ohjelma käynnistetään, se alkaa etsiä Data Matrix viivakoodin etsintäkuviota. Kun sovellus löytää etsintäkuvion, se tulkitsee koodin dataosasta numerosarjan. Numerosarjan perusteella sovellus hakee UpCoden palvelimelta siihen linkitetyn URL osoitteen ja aukaisee sen puhelimen selaimen. UpCoden palvelimella on hallintasivut, joiden avulla viivakoodeihin liitetään osoite.

Tätä työtä varten minulla oli käytettävissä MySQL tietokanta. Tein tietokannan sisältävälle palvelimelle www-sivut joiden avulla tietoja voidaan hallita tietokoneella sekä matkapuhelimella. Sivut toteutin HTML ja PHP kielillä. Matkapuhelimella tapahtuvassa hallinnassa viivakoodin osoittama URL osoite sisältää id numeron, jonka perusteella tuotteeseen liittyvät tiedot luetaan tietokannasta. Osoite on muotoa *http://iposoite/manage.php?id=x*”, jossa *x* on tilaukseen liittyvän tuotteen id tunnus. Sivun ulkoasultaan mahdollisimman yksinkertainen, koska sitä käytetään matkapuhelimella ja silloin käytettävyys ja mahdollisimman vähäinen tiedonsiirto ovat oleellisia asioita. Tietokoneella on tarkoitus hallita sivuja, joilla tietokantaan lisätään tuotteita ja tilauksia.

3.2 TIETOKANNAN KUVAUS

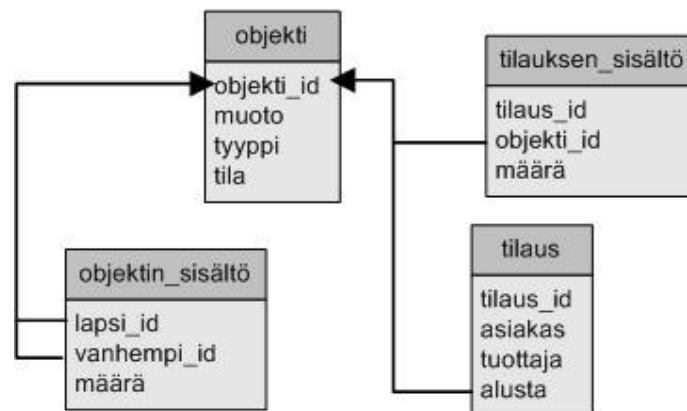
Tavarat ovat tietokannassa objekteja, jotka on yksilöity id numeron avulla. Objektit voivat sisältää toisia objekteja, esimerkiksi jos kuljetusalustana on lava ja sen päällä on tuotteita. Tämä on kuvattu lapsi-vanhempi – suhteella. Lisäksi objekteilla on muoto, tyyppi ja tila.

Tyyppi voi olla esimerkiksi 16mm pultti ja muoto voi olla esimerkiksi 2kg laatikko. Tila voi olla esimerkiksi tilattu tai vastaanotettu. Objektit kuuluvat aina tiettyyn projektiin, jotka on myös yksilöity id numeroilla.

Tilaukset sisältävät asiakkaan, tuottajan sekä kuljetusalustan, joka on objekti. Tilauksilla on sisältö, joka koostuu objekteista ja niiden määristä.

Kun tietokannassa muutetaan jotakin, esimerkiksi objektin tilaa, tallentuu muutokseen aikaleima. Tiedot tehdyistä muutoksista jää talteen ja niistä uusin on aina voimassa. Joihinkin muutoksiin tallentuu myös sen henkilön tunnus, joka muutoksen teki. Henkilöiden tiedot löytyvät tietokannasta.

Kuvassa 23 on esitetty yksinkertaistettu malli tietokannasta, joka havainnollistaa tärkeimpien taulujen suhteita.

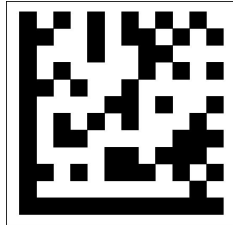


Kuva 23: Tietokanta

3.3 LUKUTESTIEN KUVAUS

Testien tarkoituksena oli selvittää miten mahdolliset olosuhteiden aiheuttamat häirtatekijät vaikuttavat lukemisen onnistumiseen. Testeissä käytin Nokian N70- sekä 6600 – mallin matkapuhelimia ja UpCoden ohjelmistoa. Viivakoodina käytin kuvan 24 esittämää koodia,

jonka olin UpCoden hallintasivujen kautta liittänyt testisivuun. Tulostettu viivakoodi oli kooltaan 2cm*2cm ja siinä oli 12*12 pikseliä.



Kuva 24: Testeissä käytetty viivakoodi

Testaaminen tapahtui liikuttamalla matkapuhelinta viivakoodia kohti ja kirjaamalla ylös miltä etäisyydeltä lukutapahtuma onnistui. Erilaisia häirtatekijöitä simuloitiin mittaamalla lukuetaisyyttä kun viivakoodi oli kuiva, märkä, muovin takana, valoisassa, hämärässä sekä 45 asteen kulmassa.

3.4 LUKUTESTIEN TULOKSET

Testien tulokset on esitetty taulukossa 2. Tuloksista nähdään että muovilla peittäminen ei vaikeuttanut lukemista käytännössä ollenkaan. Viivakoodin kasteleminen heikensi hieman luettavuutta, mutta lukeminen kuitenkin onnistui. Valoisuuden puute vaikutti luettavuuteen huomattavasti, ja täysin pimeässä en saanut lukemista onnistumaan ollenkaan. Myös kulma vaikuttaa erittäin suuresti, sillä 45 asteen kulmassa lukeminen ei onnistunut myöskään ollenkaan. Matkapuhelimen ollessa 10cm etäisyydellä viivakoodista, lukeminen onnistuu noin 35 asteen kulmassa. Kokeilin lukemista myös Nokian 6600 – mallin matkapuhelimella, jossa kameran resoluutio on huomattavasti pienempi. Kuivan viivakoodin lukeminen valoisassa onnistui tällä puhelinmallilla 18,5cm etäisyydeltä. Tämä tukee teoriaa siitä että matkapuhelimen kameran resoluutiolla ei ole suurta merkitystä lukutapahtuman onnistumiseen.

Taulukko 2: Matkapuhelimen etäisyys viivakoodista, kun lukutapahtuma onnistuu(viivakoodi 2cm*2cm/12px*12px)

	Kuiva	Märkä	Muovin takana
Valoisassa	23cm	19cm	22,5cm
Hämärässä	18cm	13cm	18cm
45° kulmassa valoisassa	x	x	x

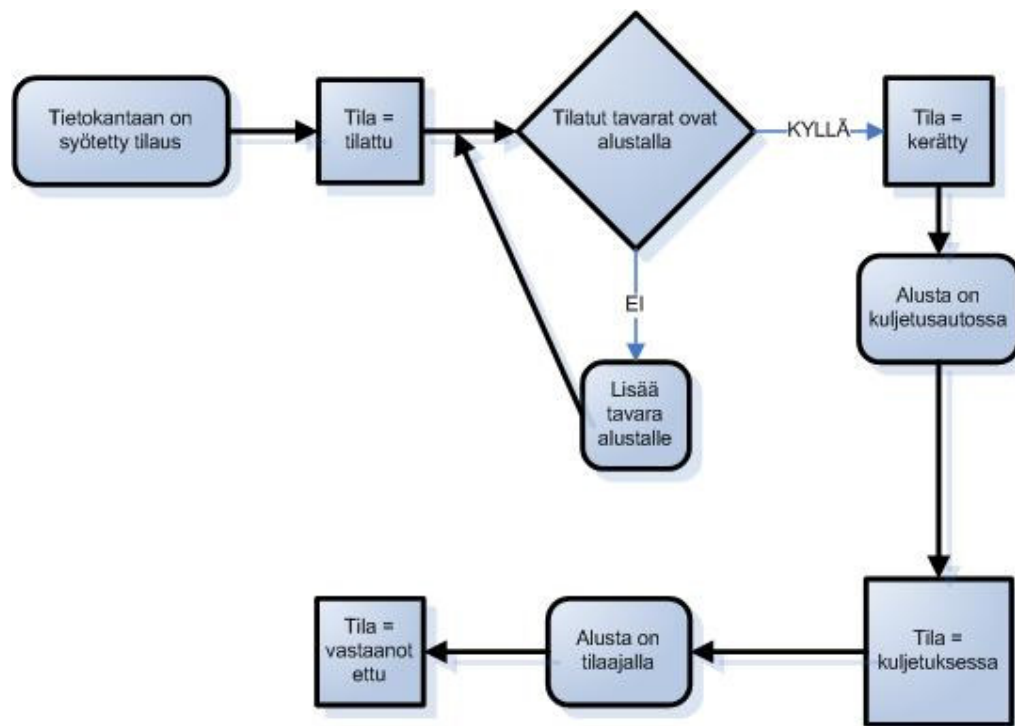
Tuloksista voidaan päätellä että jos viivakoodi on mahdollista peittää muovilla, se on kannattavaa. On mahdollista että jos muovi on kurtussa, se taittaa valoa ja heikentää luettavuutta, mutta käytännössä lukemista suorittavalla henkilöllä on luultavasti mahdollisuus suoristaa muovi. Joissakin puhelinmalleissa saattaa olla mahdollisuus valaista kuvattavaa kohdetta, mutta testeissä käyttämälläni Nokian mallilla kuvatessa hämärässä saattaisi olla järkevää pitää mukana jonkinlaista valaisinta. Kohteisiin olisi myös hyvä laittaa viivakoodeja mahdollisimman usealle sivulle varmistuakseen että lukeminen päästään tekemään kohtisuoraan.

3.5 KÄYTTÖTAPAUS: TOIMITUKSEN SEURAAMINEN

Simuloin seuraavanlaisen käyttötapauksen:

Yritys B tilaa yritykseltä A 4 kpl 2kg laatikoita 8mm pultteja sekä 2 kpl 4kg laatikoita 16mm pultteja. Tilaus syötetään tietokantaan. Lava liitetään tilaukseen ja siihen tulostetaan viivakoodi. Tässä vaiheessa tilaus menee "tilattu" tilaan. Työntekijä lukee matkapuhelimella viivakoodin ja katsoo mitä tilaukseen kuuluu. Hän laittaa tilaukseen

kuuluvat pulttिलातिकot lavalle ja vaihtaa tilauksen "kerätty" tilaan. Viivakoodi luetaan jälleen kun lava laitetaan kuljetusautoon ja vaihdetaan tila "kuljetuksessa" tilaan. Tilaus saapuu yritykselle B ja työntekijä lukee matkapuhelimella viivakoodin ja vaihtaa tilauksen tilaksi "vastaanotettu". Kaiken tämän aikana on tallentunut talteen kuka työntekijä on ollut kyseessä sekä ajat kun tilat ovat vaihtuneet. Työntekijän tunnus saadaan kun hän kirjautuu järjestelmään tunnuksellaan ja salasanallaan. Käyttötapausten mukaista toimituksen seuranta esittää kuva 25.



Kuva 25: Toimituksen seuranta

Aloitan simulaation menemällä tietokoneen selaimella sivulle, jonka avulla tietokantaan lisätään tavaroita. Ohjaudun kirjautumissivulle, jolle syötän käyttäjätunnuksen ja salasanan. Tämän jälkeen ohjaudun takaisin ja minulla on sessio voimassa seuraavan 30 minuutin ajan. Kirjautumisen yhteydessä sessioon tallentui käyttäjätunnukseni.

Seuraavaksi lisään tietokantaan tarvittavat tavarat ja siirryn sivulle, jonka avulla tietokantaan lisätään tilauksia. Syötän tilaukseen asiakkaan, tuottajan sekä kuljetusalustan, joka oli tässä tapauksessa lava. Seuraavaksi lisään tilaukseen käyttötapausten mukaiset

tavarat ja valitsen ”valmis”. Nyt tilaus menee tilaan ”tilattu” ja ruudulle tulostuu URL osoite, jonka käyn UpCoden hallintasivuilla liittämässä viivakoodiin. Tulostan viivakoodin, ja nyt olen vaiheessa jossa tuottajan työntekijä alkaa kerätä tilausta. Tästä eteenpäin kaikki tapahtuu matkapuhelimella.

Avaan matkapuhelimella UpCoden sovelluksen ja osoitan kameralla viivakoodia. Joudun taas syöttämään tunnukset, koska matkapuhelimella ei ole sessiota voimassa. Nyt olen tilauksen hallintasivulla, josta päivitän tuotteiden määrät alustalle. Nyt hallintasisivu on kuvan 26 mukainen.

Tilauksen hallinta
Tilauksen id 2:

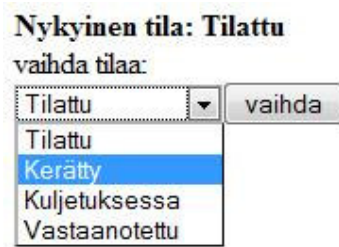
valinta	muoto	tilattu	alustalla
<input type="radio"/>	Pultti 8mm, 2 kg laatikko	4	4
<input type="radio"/>	Pultti 16 mm, 4 kg laatikko	2	2

muuta valinnan määrää alustalla:
uusi määrä

Nykyinen tila: Tilattu
vaihda tilaa:

Kuva 26: hallintasisivu

Tässä vaiheessa tilaus on kerätty, joten vaihdan kuvan 27 esittämällä valitsimella tilauksen tilaksi ”kerätty”. Seuraava vaihe käyttötapauksessa on kun kuljetusauto tulee hakemaan tilauksen. Nyt vaihdan tilaksi ”kuljetuksessa”.



Kuva 27: tilan vaihtaminen

Seuraavaksi lava on asiakkaalla joka sen tilasi. Vaihdan tilaksi ”vastaanotettu”, ja nyt käyttötapaus on suoritettu.

4 YHTEENVETO

Työn tavoitteena oli tutkia eri viivakoodityyppejä sekä niiden lukumahdollisuuksia matkapuhelimen kameran avulla. Tulkitun viivakoodin avulla oli tarkoituksena siirtää tuotteeseen liittyviä tietoja palvelimelta matkapuhelimeen.

Työlle asetetut tavoitteet tulivat onnistuneesti täytettyä. Työ esitteli eri viivakoodityyppejä ja niiden ominaisuuksia, ja tuli myös ilmi mikä viivakoodityyppi on hyvä vaihtoehto matkapuhelimen kameran avulla lukemista varten. Valinta perustui laajaan käyttöön sekä suurimpaan data-kapasiteetti/koko – suhteeseen.

Simuloin myös työssä toimituksen seurantaan liittyvän mahdollisen käyttötapauksen onnistuneesti. Järjestelmä vaikutti kätevältä mutta oikeassa käyttötilanteessa saattaisi nousta esiin volyymin mukana tuomia haasteita, kuten viivakoodien linkittäminen ja tulostaminen laajemmassa mittakaavassa.

Suoritetut lukutestit tuottivat suuntaa antavaa tietoa viivakoodien sijoittamista koskien. Testit olisivat voineet olla laajempia, mutta testit ovat kuitenkin riittäviä käyttötapauksen mukaiseen käyttöön, koska lukemisen suorittavalla henkilöllä on oletettavasti mahdollisuus suorittaa lukeminen vaikka matkapuhelin kiinni viivakoodissa. Lisäksi viivakoodien yhteyteen on mahdollista liittää viivakoodin sisältämä numero tekstimuodossa, jolloin sen voi syöttää sovellukseen manuaalisesti.

LÄHDELUETTELO

- [1] Mobilding. <http://www.it.lut.fi/project/mobilding/> (4.12.2007)
- [2] Worth Data Inc. *Bar Code Primer*. Saatavissa: <http://www.barcodehq.com/manuals/primer.pdf> (4.12.2007)
- [3] TechnoRiver. *Barcode Symbology*. Saatavissa: <http://www.technoriversoft.com/barcode.html> (4.12.2007)
- [4] Robert B. Johnston, Alvin Khin Choy Yap. *Electronic Data Interchange using Two Dimensional Bar Code*. Thirty-First Annual Hawaii International Conference on System Sciences-Volume 4 p. 0083
- [5] Hao Wang, Yanming Zou. *Camera Readable 2D Bar Codes Design and Decoding for Mobile Phones*. IEEE ICIP2006.
- [6] Tasos Falas, Hossein Kashani. *Two-Dimensional Bar-code Decoding with Camera-Equipped Mobile Phones*. Fifth Annual IEEE International Conference on Pervasive Computing and Communications Workshops(PerComW'07).
- [7] IDAutomation. *Datamatrix Barcode FAQ & Tutorial*. Saatavissa: <http://www.idautomation.com/datamatrixfaq.html> (17.12.2007)
- [8] Kongqiao Wang; Yanming Zou; Hao Wang. *Bar code reading from images captured by camera phones*. Mobile Technology, Applications and Systems, 2005 2nd International Conference on 15-17 Nov. 2005
- [9] J2ME. <http://java.sun.com/javame/index.jsp> (27.12.2007)
- [10] Chung-Hua Chu, De-Nian Yang, Ming-Syan Chen. *Image Stabilization for 2D Barcode in Handheld Devices*. MM'07, September 23-28, 2007.
- [11] Semacode technical white paper. *Choosing the best 2D barcode format for mobile apps*. Saatavissa: http://semacode.com/documents/best_2d_code.pdf (8.12.2007)
- [12] UpCode Ltd. <http://www.upc.fi/en/upcode/> (26.9.2007)
- [13] MobileTag. <http://www.mobiletag.com/beta/en/> (26.9.2007)
- [14] Hui Lei, Marion Blount, Carl Tait. *DataX: an Approach to Ubiquitous Database Access*. Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA apos;99. Second IEEE Workshop.25-26 Feb 1999 Page(s):70 - 79