

---

# Automaatioprojektien tarjouslaskennan työkalujen kehittäminen

---

Niko Pikkarainen

Opinnäytetyö

Neste Jacobs Oy





Koulutusala Tekniikka, Varkaus			
Koulutusohjelma Automaatiotekniikka			
Työn tekijä(t) Niko Pikkarainen			
Työn nimi Automaatioprojektien tarjouslaskennan työkalujen kehittäminen			
Päiväys	28.01.2011	Sivumäärä/Liitteet	29+6
Ohjaaja(t) Tero-Markus Jankko			
Toimeksiantaja/Yhteistyökumppani(t) Neste Jacobs Oy			
Tiivistelmä			
<p>Päättötyön tavoitteena oli kehittää tarjouslaskennan työkalu Neste Jacobsin automaatio- ja sähköistysosaston käyttöön.</p> <p>Työssä käytettiin Microsoftin Access 2003 ohjelmistoa, jolla kehitettiin jo olemassa olevaa tietokantapohjaa. Tarkoituksena oli saada aikaan tietokanta, joka tuottaa käytettyjen lähtötietojen pohjalta projektien työmäärä- ja hinta-arvot ja täten yhtenäistää ja helpottaa yrityksen tarjouslaskentaa. Tietokanta toimii myös raportointityökaluna.</p> <p>Alkutietojen mukaan toteutettiin tietokanta, jota muutettiin sitten käyttäjän antaman palautteen myötä.</p> <p>Tietokantaa testattiin aluksi syöttämällä sinne lukuisia tarjouksia kuvitteellisille asiakkaille, mutta viimeinen testaus tapahtuu julkaistavan version myötä. Tietokannan ja työkalun kehitystä jatketaan edelleen tämän työn valmistuttua. Jatkokehitys on rajattu tämän työn ulkopuolelle.</p>			
Avainsanat Automaatioprojekti, tarjouslaskenta, kehittäminen, tietokanta			

Field of Study Business and Engineering, Varkaus			
Degree Programme Automation technology			
Author(s) Niko Pikkarainen			
Title of Thesis Development of the Quotation Tools for Automation Projects			
Date	28.01.2011	Pages/Appendices	29+6
Supervisor(s) Tero-Markus Jankko			
Project/Partners Neste Jacobs Oy			
<p>Abstract</p> <p>The goal of the project was to create a quotation tool for the Automation- and Electrification department of Neste Jacobs.</p> <p>In this project Microsoft Access 2003 Database program was used. The starting point was an existing preliminary database that was developed to meet the demands. The pricing tool is supposed to calculate quotations and the workload caused by the project. The tool must have several different types of reports like monthly quotations, monthly orders etc.</p> <p>The tool is supposed to help the sales team of Neste Jacobs to make more accurate offers more easily and to improve the pricing knowledge of the whole Automation and Electrification-department.</p> <p>Testing was made by feeding many fictitious customers/quotations to the system. The final testing will be made by users. The future tool will also include some new features, but those features are out of the scope of this final thesis.</p>			
<p>Keywords Database, Automation, Electrification, VBA, Quotation tool</p>			

## SISÄLTÖ

1	JOHDANTO .....	7
1.1	NESTE JACOBS.....	7
2	MICROSOFT ACCESS.....	9
2.1	Tietokanta .....	9
2.1.1	Relaatiotietokanta .....	9
2.2	Tietokannan suunnittelun perusteita .....	10
3	TYÖN TOTEUTUS.....	12
4	YHTEENVETO.....	27
	LÄHTEET .....	29

## LIITTEET

- Liite 1 Valid Quotations Report
- Liite 2 Hour report
- Liite 3 Työkalun käyttöohje
- Liite 4 Lohkokaavio tietokannan rakenteesta



## 1 JOHDANTO

Työn tarkoituksena oli kehittää tietokantapohjainen työkalu Neste Jacobsin sähkö- ja automaatio-osaston käytettäväksi. Hanke on osa suurempaa myynnin kehittämis/yhtenäistämiprojektia. Tartuin haastavalta kuulostavaan aiheeseen, koska minulla ei ole aikaisempaa kokemusta tarjouslaskennasta, mutta kova halu oppia uusia asioita. Tiesin myös sen, että kannustava ja auttava työyhteisömme auttaa varmasti, kun ongelmatilanteita ilmaantuu.

Alkutietoina ohjelmalla oli, että sen olisi laskettava työhön tarvittavat tunnit ja muut projekti-kustannukset Neste Jacobsille sekä määritellyn marginaalin avulla koko projektin tarjoushinta asiakkaalle. Ohjelma tallentaa tunnit eri työlajeille sekä eri työntekijäkategorioihin (jaettu grouppeihin tehtävien mukaan). Tämän jaottelun ja tietokantaan tallennetun hinnaston avulla ohjelma laskee koko projektille hinnan.

Ohjelmasta saadaan ajettua myös raportteja. Kuukausiraportissa pystytään tarkastelemaan edellisen kuukauden voimassa olevia tarjouksia, tilaukseen menneitä tarjouksia ja hävittyjä tarjouksia. Tämän lisäksi voidaan tarjouskohtaisesti tulostaa tuntiraportteja, joissa näkyy myös kustannukset.

Työn aikana opin paljon VBA-ohjelmoinnista ja muutenkin Accessin toiminnasta, aikataulutetussa projektissa toimimisesta sekä automaatioprojektien tarjousprosessin vaiheista.

### 1.1 NESTE JACOBS

Neste Jacobsin historia ulottuu vuoteen 1956, jolloin se oli Neste Oy:n suunnitteluosasto. Vuonna 1999 suunnitteluosasto yhtiöllistettiin ja nimeksi muodostui Neste Engineering Oy. Vuonna 2004 Jacobs Engineering Inc. osti osan Neste Engineeringiä, uuden yhtiön nimeksi muodostui Neste Jacobs.<sup>[1]</sup>

Neste Jacobsin päätoimialat ovat:

- Öljy ja kaasu
- Biorefining
- Petrokemian teollisuus
- Kemian teollisuus (mukaan lukien laitokset ja life science sektori) <sup>[1]</sup>

Neste Jacobsilla on kaksi omistajaa (Neste Oil Oyj 60 % ja Jacobs Engineering Group 40 %).<sup>[1]</sup>

### **Neste Oil Corporation**

Neste Jacobsin emoyhtiö on öljynjalostus- ja markkinointiyhtiö, joka on erikoistunut vähäpäästöisiin ja korkealaatuisiin liikennepolttoaineisiin. Yhtiön strategia perustuu öljynjalostuksen ja Premium-laatuisten uusiutuvan dieselin toimialan kasvattamiseen. Neste Oilin öljynjalostamot sijaitsevat Porvoossa ja Naantalissa, ja niiden yhdistetty raakaöljyn jalostuskapasiteetti on noin 260 000 barreliä päivässä. <sup>[1]</sup>

### **Jacobs Engineering Group Inc.**

Jacobs Engineering Group Inc. on yksi maailman suurimmista ja monipuolisimmista teknisten palveluiden tarjoajista. Yli 12 miljardin dollarin liikevaihdolla yhtiö tarjoaa koko palveluiden kirjon teollisuudelle, kaupallisille ja valtioasiakkaille useilla markkina-alueilla. Palvelut sisältävät tieteellistä ja erityiskonsultointia, opastusta/näkökantoja suunnittelussa ja rakentamisessa sekä käyttö- ja kunnossapitopalveluita. <sup>[1]</sup>

Maaliskuussa 2008 Neste Jacobs osti kokonaan Rintekno Groupin (Rintekno Oy, Systecon Oy, Turun Sähkösuunnittelu Oy, Kotka Control Oy ja Rintekno AB), joka on nykyään integroitu yhtiöön. Neste Jacobsilla on toimistoja Suomessa Porvoossa (pääkonttori), Turussa, Naantalissa ja Kotkassa, sekä Ruotsissa Göteborgissa. Suuremmissa projekteissa perustetaan väliaikaisia konttoreita työmaiden yhteyteen.

Neste Jacobsin missio on "Excellence in Engineering", eli olla alan johtava palveluntarjoaja. Visio on "Preferred solution provider for hydrocarbon and biorefining industries".

Arvot: Responsibility, Cooperation, Innovation ja Excellence.<sup>[1]</sup>

Neste Jacobs työllistää n. 700 henkilöä erilaisissa tehtävissä sekä kotimaassa että ulkomailla. <sup>[1]</sup>



## 2 MICROSOFT ACCESS

Microsoft Access on Office Enterprise ja Professional -ohjelmistopakettien mukana tuleva tietokannan hallinta- ja kehitysohjelmisto.

Tietokannan hallintaohjelmisto kuten Microsoft Access järjestee ja tallentaa tietoja samalla tavalla kuin arkistokaappi järjestee paperisia tietolomakkeita. Tietokantaohjelmistolla voidaan lisätä, tarkistaa ja noutaa tietoa nopeasti ja tehokkaasti. Tietokantaohjelmisto jakaa tiedon sisältöä kuvaaviin paikkoihin, joten jokaisessa tietokannassa taulukot sisältävät tietynlaista tietoa (kuten asiakastiedot tai tilaukset).<sup>[8]</sup>

Microsoft Access ei ole ainoa tietokantasovellus, mutta Office-paketin mukana tullessaan se on luultavasti laajimmin käytössä oleva. Muita tietokannan hallintaohjelmistoja ovat esim. OpenOffice.org -paketin mukana tuleva Base, Apache Derby ja Oracle Database.

### 2.1 Tietokanta

Tietokanta on kokoelma tietoja, joilla on yhteys toisiinsa. Tietokannat on käytännöllisintä luokitella tuetun ohjelmointimallin mukaan. Hierarkinen malli toteutettiin ensimmäisenä. Sen jälkeen tuli verkkomalli. Sitten relaatiomalli ohitti ne niin kutsutun "Flat-File"-mallin kanssa, joka oli helppo toteuttaa vaatimattomiinkin alustoihin. Hierarkinen, verkko- ja Flat-tiedostomalli eivät perustu vahvaan teoreettiseen pohjaan niin kuin relaatiomalli, vaan ne ovat syntyneet laitteiston ja ohjelmointitekniisten rajoitteiden ohjatessa mallin kehitystä.<sup>[3]</sup>

#### 2.1.1 Relaatiotietokanta

Vuonna 1970 E.F.Codd esitteli relaatiomallin. Tämä oli siihenastisista tietokantamalleista yksinkertaisin ja joustavin ja toteutti parhaiten tietokannalle asetettavat vaatimukset. Ikävänä puolena relaatiomallissa oli suuri koneresurssien tarve.<sup>[4]</sup>

Relaatiotietokannassa tiedot esitetään tauluina (engl. table), joita kutsutaan myös relaatioiksi. Yhtä riviä kutsutaan tietueeksi (engl. record). Taulun jokaisella rivillä on yhtä monta tietoa eli kenttää (engl. field). Jokaisella rivillä täytyy olla yksikäsitteinen perus-

avain, joka vastaa jotakin reaali maailman kohdetta. Kuhunkin kohteeseen liitetään vain siihen välittömästi liittyvät ominaisuudet. Kukin yksittäinen tieto relaatiotietokannassa voidaan hakea ainakin ilmoittamalla taulun nimi, perusavaimen kentän nimi ja avaimen arvo sekä haettavan tiedon kentän nimi. Lisäksi on olemassa lukemattomia muita tapoja hakea tietoa. Relaatiotietokannasta tietoa haetaan vain tiedon nimien ja arvojen perusteella, ei siis koskaan tiedon sijainnin tai järjestyksen mukaan.<sup>[4]</sup>

## 2.2 Tietokannan suunnittelun perusteita

Oikein suunniteltu tietokanta mahdollistaa sen, että saadaan käyttöön päivitettyjä ja tarkkoja tietoja. Koska oikea suunnittelu on oleellista tavoitteiden saavuttamisessa tietokantaa käytettäessä, hyvän suunnittelun periaatteiden opettelemiseen kannattaa panostaa aikaa. Lopputuloksena on mitä todennäköisimmin sellainen tietokanta, joka vastaa tarpeita ja jota voi mukauttaa myöhemmin.<sup>[6]</sup>

Tietyt periaatteet ohjaavat tietokannan suunnitteluprosessia. Ensimmäinen periaate on se, että kaksoistiedot (kutsutaan myös toistuviksi tiedoiksi) ovat huonoja, koska ne vievät turhaa tilaa ja lisäävät virheiden ja epä johdonmukaisuuksien todennäköisyyttä. Toisen periaatteen mukaan tietojen oikeellisuus ja eheys ovat tärkeitä. Jos tietokanta sisältää virheellisiä tietoja, raportit, jotka hakevat tietoja tietokannasta, sisältävät silloin myös virheellisiä tietoja. Tämän tuloksena kyseisiin raportteihin perustuvat päätökset ovat vääriä. Mikäli tiedot ovat menettäneet eheydensä, niitä ei välttämättä voida käydä lukemassa. Tästä johtuen käyttäjä ei välttämättä saa tarvitsemaansa tietoa tietokannasta.<sup>[6]</sup>

Tietokannalle siis asetetaan ainakin seuraavat vaatimukset:

- Kukin tieto tallennetaan kannassa vain yhteen paikkaan eli tietokannassa ei esiinny turhaa toistoa (engl. redundancy).
- Tietoja pystytään hakemaan joustavasti erilaisin perustein. Myös sellaisin, joita ei tietokantaa suunnitellessa ole pystytty ennakoimaan.
- Tietokannan rakenteen muuttaminen on joustavaa.
- Sovellusohjelmat ovat riippumattomia tietojen fyysisestä tallennusrakenteesta, mitä kutsutaan tietoriippumattomuudeksi.<sup>[5]</sup>

Suunnitteluprosessi koostuu seuraavista vaiheista:

### **Tietokannan tehtävän määrittäminen**

Tämä auttaa valmistautumaan jäljellä oleviin vaiheisiin. <sup>[6]</sup>

### **Vaadittavien tietojen etsiminen ja järjestäminen**

Kerää kaikenlaiset tiedot, jotka halutaan ehkä kirjoittaa tietokantaan, kuten tuotteiden nimet ja tilausnumerot. <sup>[6]</sup>

### **Tietojen jakaminen tauluihin**

Jaetaan tieto-osat pääkokonaisuuksiksi tai -aiheiksi, kuten Tuotteet tai Tilaukset. Jokaisesta aiheesta tulee tällöin taulu. <sup>[6]</sup>

### **Tieto-osien muuntaminen sarakkeiksi.**

Päätetään, mitkä tiedot halutaan tallentaa kuhunkin tauluun. Jokaisesta osasta tulee kenttä, ja se näytetään sarakkeena taulussa. Työntekijät -taulu saattaa esimerkiksi sisältää kentät Sukunimi ja Aloittanut. <sup>[6]</sup>

### **Perusavainten määrittäminen**

Valitaan jokaisen taulun perusavain. Perusavain on sarake, jota käytetään tunnistamaan jokainen rivi yksilöllisesti. Esimerkkinä voidaan mainita Tuotetunnus tai Tilaustunnus. <sup>[6]</sup>

### **Taulukoiden välisten yhteyksien määrittäminen**

Tarkastellaan jokaista taulua ja päätetään, miten yhden taulun tiedot liittyvät toisten taulujen tietoihin. Selvennetään yhteyksiä tarpeen mukaan lisäämällä kenttiä tauluihin tai luomalla uusia tauluja. <sup>[6]</sup>

### **Rakenteen hienosäätäminen**

Tutkitaan, onko rakenteessa virheitä. Luodaan tauluja ja lisätään niihin joitakin uusia mallitietotietueita. Katsotaan, saako tauluista haluttuja tuloksia. Säädetään rakennetta tarpeen mukaan. <sup>[6]</sup>

### **Normalisointisääntöjen käyttöönotto**

Tarkastetaan, onko taulut muodostettu oikein, ottamalla käyttöön normalisointisäännöt. Säädetään tauluja tarpeen mukaan. <sup>[6]</sup> Normalisoinnilla pyritään vähentämään ylimääräisiä tietoja (tyhjiä kenttiä tauluissa tai tietojen turhaa toistamista), vähentämään ongelmia päivityksessä sekä lisäämään selkeyttä ja laajennettavuutta. <sup>[11]</sup>

### 3 TYÖN TOTEUTUS

Työ aloitettiin tutustumalla Reijo Nokelaisen tekemään tietokanta-aihiin, jota käytettiin pohjana. Nokelaisen tietokanta osoittautui sopivaksi lähtökohdaksi halutulle työkalulle, joten sitä lähdettiin kehittämään eteenpäin. Projektin asiakastietojen kysely jätettiin entiselleen, mutta hintatietojen syöttö automatisoitiin.

Vanhassa pohjassa oli suoraan kysytty projektin hintaa, mikä nyt piti muuttaa siten, että ohjelma laskee hinnan automaattisesti. Piti siis rakentaa lomake, jossa kysytään eri työlajien arvioidut tunnit, sekä tarkentava lomake, joka jaottelee tunnit vielä eri kustannuslajien kesken.

Tiedot tallennettiin ensin Hour\_by\_Service nimiseen tauluun, jossa on käytetty indeksointiavaimena tarjousnumeroa. Tarkentavan lomakkeen tunnit tallennettiin Tuntien\_ryhmajaottelu nimiseen tauluun, jotta pystytään mahdollistamaan projektin työ kustannusten laskeminen. Tunnit syötetään jokaiselle laskutusluokalle manuaalisesti, jonka jälkeen ohjelma laskee jokaiselle laskutusluokalle kulut (mitä laskutusluokan tunnit maksaa firmalle) ja hinnan (mitä firma laskuttaa asiakkaalta). Tämän jälkeen kaikki tunnit, kulut ja hinnat lasketaan yhteen ja lomakkeelta poistuttaessa tallennetaan Hour\_by\_Service lomakkeelle. Sähköistyksen tuntiensyöttölomake on tehty samalla pohjalla kuin automaationkin (Hour\_by\_Service), ja sinne päästään painamalla Hour\_by\_Service lomakkeen yläreunassa olevaa painiketta "Go to Electrical Hours" (Katso kuva1).

The image displays two screenshots of the 'Hour\_Feed' software interface, which is used for calculating automation and instrumentation hours. The interface is divided into two main sections: the left side (top-left) and the right side (top-right).

**Left Screenshot (Top-Left):** This section shows a detailed breakdown of hours and costs for various engineering tasks. The tasks are grouped into categories like Design Basis, Application Engineering, Instrumentation Engineering, and Analyzers. Each task has a 'Suggested Hours' and a 'Cost (€)' column. The total suggested hours for the Design Basis category is 66,650.00. The tasks listed include Control and safety system philosophy, F and G detection system philosophy, Machinery protection system philosophy, Field instrumentation philosophy, Control descriptions, Control diagrams, Interlock descriptions, Logic diagrams, Sequence descriptions, Process computer systems, Simulators, High level controls, Data base (SPI), Cabling and wiring engineering (SPI), Field instrument location engineering, Control room and end rack room layout engineering, Calculations (Instrument sizing, power, instrument air, Etc), Instrument installation engineering, Purchase requisitions, SIL verification, and Job definitions.

**Right Screenshot (Top-Right):** This section shows a summary of the total hours and costs for different project phases. The phases are grouped into categories like Automation System Engineering, Site Engineering, Quality, and Coordination. Each phase has a 'Total (h)', 'Total cost', 'NJ Hours (h)', and 'NJ Cost (€)' column. The total hours for the Automation System Engineering phase is 0.00. The phases listed include DCS specification and purchase requisition, DCS system implementation, FAT, SIS specification and purchase requisition, SIS system implementation, FAT, Product tracking and recipe system specification, Product tracking and recipe system implementation, GDS specification and purchase requisition, Site support during installation, Functional testing of SIFs, Precommissioning, Review meetings, Commenting, Checking etc., Participation in quality audits, Execution Planning, Management, and Coordination meetings.

Both screenshots show a 'Go to Electrical hours' button in the top right corner. The bottom of each screenshot shows a record navigation bar with 'Record: 14 of 1'.

Kuva 1 Automaation ja instrumentoinnin tuntiensiötyölomake (vas. yläosa ja oik. alaosa)

Laskenta Accessissa oli hankala. Vaikka taulut näyttävät Excel-taulukoilta, ei niillä pysty laskemaan samalla tavalla kuin Excel-taulukoilla. Access-laskennasta tekee vaikeamman se, että soluja joita haluat kaavassa käyttää, ei voi klikkaamalla valita. Laskenta Accessissa tulisi toteuttaa kyselyiden kautta (Englanniksi Query). On kuitenkin helpompaa toteuttaa laskenta VBA-koodilla kuin kyselyillä. Internetistä löytyy paljon ohjeita erilaisiin koodausongelmiin, kun laittaa hakusanaksi "VBA" ja sitten määrittelee ongelman. Tämän lisäksi Access-osaajamme auttoi vaikeimpien pulmien ratkaisussa. Koko ohjelman laskenta on toteutettu VBA-koodilla, koska se oli helpompi ja nopeampi tapa. VBA:n käyttöä tuki myös se, että joitakin automatisointeja ei olisi voinut toteuttaa ilman koodia. Taulukossa 1 on esitelty joitakin ominaisuuksia, joita voidaan tehdä VBA:lla, mutta jotka on vain osittain toteutettavissa makroilla.

VBA	Makrot
Case-rakenne	Osittain toteutettavissa Condition-sarakkeella
Loop-silmukat	Osittain toteutettavissa Repeat-toiminnolla käyttämällä laskurina esim. lomakkeen kentän arvoa
Vakiot ja muuttujat	Osittain toteutettavissa käyttämällä lomakkeen kentän arvoa
Funktiot: omat ja API	
Transaction-datan eräajot	RollBack esim. väliaikaisilla tauluilla ja Delete-kyselyillä
Tietokanta-objektien luominen	Taulujen muokkaus SQL DDL-kyselyillä
Virheiden käsittely	
OLE ja DDE-operaatiot	Osittain korvattavissa linkityksellä Office-ohjelmiin

TAULUKKO 1 VBA toimintoja, jotka ovat vain osittain toteutettavissa makroilla <sup>[12]</sup>

Työn eteneminen hidastui jälleen, kun yksinkertainen kertolasku ei toiminut. Ohjelmassa oli lomake, jossa oli kentät Group 50 ja GR50\_Cost, ja näiden lisäksi täytyi saada tauluun tallennetusta hinnastosta haettua kustannus. Kaava toimi muuten, mutta tauluun viittaus kaatoi ohjelman. Ensin yritettiin käyttää samanlaista viittausta kuin lomakkeiden Tekstiruutuihin viitatessa käytetään

( [Forms]![Formin nimi]![Kentän nimi] ). Koska tämä viittaus kaatoi ohjelman, täytyi selvittää mikä olisi oikea tapa viitata taulun tietoihin.

Tilanteeseen löytyi ratkaisu lauseesta DLookup("kentän nimi","taulu","missä tieto on"), joka hakee tauluun tallennettuja tietoja sulkujen sisällä olevien kriteerien mukaan.<sup>[2]</sup> Tällä lauseella ohjelma hakee hinnan, joten laskenta oli kunnossa.

Aloituspalaverin opettajan kanssa pidettiin poikkeuksellisesti melkein kuukauden kulu-  
tua itse työskentelyn aloituksesta. Ennen aloituspalaveria pidettiin esimieheni ja Harrin  
kanssa tilannekatsaus, jossa katsottiin mitä kaikkea on saatu siihen mennessä valmiiksi  
sekä saatiin tärkeää palautetta käyttöliittymän käytettävyydestä. Joitakin muutoksia käyt-  
töliittymään tuli, kuten se, että tuntien syöttö vaihtui omasta valikostaan tarjouksenteko-  
lomakkeelle (punaisella ympyröity kuvassa 2). Ohjelmaan myös tuli lisäyksiä, kuten Pro-  
ject Managementin tuntien syöttö ja erittely sekä laitteiden ja antureiden listaus- ja hin-  
noittelulomake.

Kuva 2 Uuden tarjouksen tekolomake

Muutoksien teko alkoi linkitysten vaihdolla NEW\_QUOTATION\_Form lomakkeelle, koska se oli helppo ja nopea homma tehdä. Tämän jälkeen luotiin Equipment lomake, johon voidaan syöttää listaus projektissa käytetyistä laitteista, paljonko laitteet maksavat sekä valita drop down -valikosta marginaaliprosentti. Kun painetaan Calculate-painiketta, ohjelma laskee hinnan määritettyä marginaalia käyttäen.

Kuva 3 Laitteiden syöttölomake

Ohjelmaan luotiin myös kuvan 3 mukainen yksinkertainen lomake, johon saadaan syötettyä projektin muita kuluja (esim. matkustuskulut yms.).

Expences

### Feed Expenses

Feed additional expences like meetings and travelling etc.

Quotation number: AE001-005

Autom: 0.00

Instr: 0.00

Sähkö: 0.00

Tekno: 0.00

Total (€): 0.00

Record: 1 of 1

Kuva 4 Matkakulujen yms. syöttölomake

Myös projektin johtoon kuuluvat tunnit täytyi saada huomioitua, joten niille luotiin oma tekstiruutu NEW\_QUOTATION\_Form -lomakkeelle, josta klikkaamalla pääsee ProjManagement nimiseen lomakkeeseen. Lomakkeella on eri työlajeille omat tekstiruutunsa, joita painamalla pääsee Tuntien\_ryhmajaottelu\_management nimiselle lomakkeelle. Tänne syötetään tunnit eri laskutusluokille, minkä jälkeen ohjelma laskee jokaisen laskutusluokan kulut (paljon se maksaa firmalle) ja hinnan (paljon firma laskuttaa asiakasta). Kun lomake suljetaan, tunnit tallentuvat automaattisesti oman työvaiheen tunteihin ProjManagement -lomakkeella. Sekä kustannukset että hinta summautuvat myös ProjManagementin Total Cost(€) ja Total Price(€) tekstiruutuihin. Kun ProjManagement -lomakkeessa on kaikki tarvittavat tiedot täytetty, painetaan Calculate and Save -nappia, joka nimensä mukaisesti laskee tunnit yhteen (sekä syöttää ne Total (h) tekstiruutuun ja NEW\_QUOTATION\_Form lomakkeen Work Hours tekstiruutuun ja syöttää myös Total Price (€):n tiedot NEW\_QUOTATION lomakkeen Project Management Price tekstiruutuun) ja tallentaa tiedot.

ProjManagement

Quotation number: AE001-005

Project Management: 0.00

Assistant Services: 0.00

Change Management: 0.00

Permits, Authority Eng: 0.00

Project Controls: 0.00

ICT: 0.00

Document Control: 0.00

Quality Management: 0.00

HSE Management: 0.00

Total (h): 0.00

Total Cost(€): 0.00

Total Price (€): 0.00

Calculate and Save

Record: 1 of 1

Kuva 5 Projektin johdon tuntien syöttölomake



Group	Hours	Cost	Price
Group 50	0.00	0.00	0.00
Group 51	0.00	0.00	0.00
Group 52	0.00	0.00	0.00
Group 53	0.00	0.00	0.00
Group 54	0.00	0.00	0.00
Group 55	0.00	0.00	0.00
Group 56	0.00	0.00	0.00
Group 57	0.00	0.00	0.00
Group 58	0.00	0.00	0.00
Group 59	0.00	0.00	0.00
Group 60	0.00	0.00	0.00
Group 61	0.00	0.00	0.00
Group C	0.00	0.00	0.00
Total	0.00	0.00	0.00

Kuva 6 Projektin johdon tuntien erittelylomake

Kun kaikki tarvittavat tiedot oli saatu tuotua NEW\_QUOTATION\_Form -lomakkeelle, painetaan Total Price € -tekstiruutua, jolloin siihen päivittyy koko tarjouksen kokonaishinta. Sen jälkeen painetaan Rate Price (h) -tekstiruutua, jolloin siihen päivittyy myös keskituntihinta.

Lomakkeen alareunasta löytyy myös Change Pricing -niminen painike, jolla päästään muuttamaan hinnastoa. Hinnaston muutos on muistettava tehdä ennen tuntien syöttämistä, koska muuten hinnat lasketaan vanhan hinnaston mukaan.

Kun nämä toimenpiteet on tehty, on itse tarjouksentekolomake valmis ja voidaan painaa Accept -painiketta, joka tallentaa tiedot NJ\_AE\_Quotations\_table -nimiseen tauluun. Kun painetaan aloitussivun View and Edit All Quotations -painiketta, avautuu NJ\_AE\_Quotations\_Form niminen lomake, jossa on kaikki työkalulla luodut tarjoukset. Tähän lomakkeeseen lisättiin kolme painiketta, joista pääsee tarkastelemaan automaation, sähköistyksen ja projektin johdon tunteja.

Quotation number	Date	Author	Check	Customer	Project Description	Service	Activity	Valid	Order	Lost	Project Start D	Project End D	Total Price €
AE001-001	3/24/2010	LAHTIHAR	LAITIKAL	Bayer Schering Pharr	Factory xx extension	Basic Engineering		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	3/2/2011	5/2/2016	38,700 €
AE001-002	10/1/2010	aalohan	aalohan	MIR-Sähkö Oy	Testing		Instrumentation Eng	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>			445,192 €
AE001-003	10/2/2010	aalohan	kaskekar	Senaatti-kiinteistöt	Testing2		Electrical Engineeri	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	12/15/2009	12/31/2011	683,753 €
AE001-004	10/8/2010	espoker	kaskekar	Mildola Oy, Kirkkonur	Test!!!		Instrumentation Eng	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	11/1/2010	6/3/2011	491,324 €
*	1/10/2010							<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			0 €

kuva 7 Vanhojen tarjousten tarkastelu- ja editointilomake

Viimeisenä osa-alueena oli raportoinnin tekeminen. Ensin yritettiin muodostaa raportti, jossa olisi yhdessä raportissa sekä voimassa olevat, tilatut että hävityt tarjoukset. Tällä tavoin raportista tuli kuitenkin hyvin vaikealukuinen, joten päädyttiin luomaan jokaiselle kategorialle omat raporttinsa. Raportit on "nidottu" toisiinsa VBA-koodilla, eli kun painetaan Show Report -painiketta, kaikki raporttiin kuuluvat erillisraportit tulevat näkyviin kerralla. Tämä helpottaa käyttäjän toimintaa huomattavasti, kun jokaista raporttia ei tarvitse käydä erikseen etsimässä ja avaamassa.

Raporttien rajaus vaati myös paljon perehtymistä, koska raportin suodatus ei alkanut toimimaan halutulla tavalla. Kuukausiraportoinnin aikarajaus koettiin saada asetettua siten, että kun painetaan Show Report -painiketta, Access hakee edellisen kuukauden tarjoukset raportille. Internetistä löytyi useita erilaisia suodatin malleja, mutta mikään niistä ei tuottanut toivottua lopputulosta. Pitkän etsinnän jälkeen suodatin vihdoinkin löytyi eräältä keskustelupalstalta, jossa neuvottiin tekemään uusi moduli. Moduliin tulee seuraavanlainen koodi:

```
Option Compare Database
```

```
Function BeginLastMonth() As Variant
```

```
BeginLastMonth = DateAdd("m", -1, DateSerial(Year(Date), Month(Date), 1))
```

```
End Function
```

```
Function EndOfLastMonth() As Variant
```

```
EndOfLastMonth = DateSerial(Year(Date), Month(Date), 0)
```

```
End Function
```

Rajausmodulissa käytetty koodi.<sup>[7]</sup>

Tämän jälkeen lisätään kyselyn Paivays-kenttään kriteeriksi Between BeginLastMonth() and EndOfLastMonth. Tällöin Access osaa automaattisesti suodattaa edellisen kuukauden tarjoukset raporteille.<sup>[7]</sup> Tämän lisäyksen jälkeen kuukausiraportointi alkoi toimia.

Tuntiraportoinnin hakuvaihtoehtoon lisättiin aluksi turha suodatin, joka oli tyhjä (filteri=" "). Tämä suodatin suodattikin kaikki tiedot. Suodatin poistettiin käytöstä, jolloin tuntiraportoinnin haku alkoi toimia halutulla tavalla. Nyt kun valitsee raportin tyyppiä Project Hours Report, tulee hakulehdelle näkyviin Quotation number -listboxi, josta valitaan halutun tarjouksen numero. Kun painetaan Show Report-painiketta, ohjelma avaa kolme raporttia (automaation ja instrumentoinnin tunnit, sähköistyksen tunnit, sekä projektin johtamisen tunnit), joihin on haettu tarvittavat tiedot hakuvalinnasta riippuen. Esimerkki-raportin voi katsoa Liitteistä 1 ja 2

17.11.2010 pidettiin palaveri, jossa katsottiin jälleen, kuinka työ on edistynyt ja saatiin hieman parannusehdotuksia.

Tarjouksen tietoja täytyi päästä muokkaamaan myös tarjouksen teon jälkeen. Ohjelmaan kopioitiin lomakkeet, jotka on nimetty lisäämällä alkuperäisiin nimiin "Edit" -sana. Aluksi tuntiensyöttölomake avattiin koodilla, jonka piti rajata tiedot tietyillä arvoilla ja tämä näennäisesti myös toimi. Toimintoa testatessa enemmän kävi kuitenkin ilmi, että ohjelma muuttaa ensimmäisen tarjouksen (AE001-001) tarjousnumeroa muokattavan tarjouksen numeroksi. Apua haettiin Access-osaajaltamme ja hän teki suodattimen, joka suodattaa kaikki muut tiedot pois (eli aikaisemmalla koodilla aukesi kaikki syötetyt tiedot, jolloin Access meni sekaisin ja korvasi ensimmäisen tarjouksen tarjousnumeron uusilla). Tämän suodattimen lisäämisen jälkeen tuntien muokkaus onnistui ongelmitta.

NEW\_QUOTATION -lomakkeelle tuli myös mahdollisuus lisätä uusi asiakas. Tehtiin lomake joka nimettiin NEW\_CUSTOMER nimiseksi, jossa kysellään asiakkaan tiedot. Tästä ne tallentuvat suoraan Customers\_Table nimiseen tauluun. Kun asiakas on lisätty, se (asiakas) ilmestyy NEW\_QUOTATION -lomakkeen Firm-nimiseen alavetolaatikkoon. Tämän lisäksi tarjous tarvitsi vapaan kommenttikentän (nykyisten lisäksi), joka tuli Additional info -painikkeen alle (katso kuva 8).

Kuva 8 NEW\_QUOTATION -lomake muutosten jälkeen

Katsottiin taas yhdessä parannusehdotuksia ja missä ohjelman saa kaatumaan. Tämän tuloksena käyttöliittymän rakenne muuttui siten, että tarjousta tehtäessä (NEW\_QUOTATION -lehdellä) ei syötetäkään tunteja, vaan perustetaan asiakas ja tarjousnumero. Tunnit syötetään Edit Quotationin kautta kuten aiemminkin. Tällä muutok-

sella estettiin se, ettei käyttäjä voi vahingossa tehdä tuplatietoja. Tästä johtuen NEW\_QUOTATION -lomakkeelta poistui kaikki tunteihin ja hintoihin liittyvät kentät.

Myös ennen comboboxina ollut Activity-kenttä muuttui vaihtuessaan listboxiksi. Tämä muutos tehtiin siksi, että suunnittelulajeista oli saatava valittua useampi arvo kerralla. Lisäksi käyttäjä ei enää voi itse vaikuttaa tarjousnumeroon (ennen ohjelma kysyi haluatko tämän tarjousnumeron, tällöin tarjousnumeroa pystyi muokkaamaan), vaan ohjelma ottaa tarjoukselle seuraavan vapaan numeron.

Microsoftilla oli epämääräisesti laitettu Properties-ominaisuuteen, että vaihtoehdoista voi valita useamman, mutta heillä ei ollut omaa suoraa työkalua, jolla tämän arvon olisi voinut tallentaa. Tähän tilanteeseen löytyi ratkaisu seuraavasta koodista, joka löytyi Microsoftin omilta tukisivuilta (en ymmärrä, miksi tätä ominaisuutta ei ole voitu laittaa ohjelmaan vakioksi):

```
Private Sub Form_Current()  
    Dim oltem As Variant  
    Dim bFound As Boolean  
    Dim sTemp As String  
    Dim sValue As String  
    Dim sChar As String  
    Dim iCount As Integer  
    Dim iListItemsCount As Integer  
  
    sTemp = Nz(Me.Activity.Value, " ")  
    iListItemsCount = 0  
    bFound = False  
    iCount = 0  
  
    Call clearListBox  
  
    For iCount = 1 To Len(sTemp) + 1  
        sChar = Mid(sTemp, iCount, 1)  
        If StrComp(sChar, ",") = 0 Or iCount = Len(sTemp) + 1 Then  
            bFound = False  
            Do  
                If StrComp(Trim(Me.Activity.ItemData(iListItemsCount)), Trim(sValue)) = 0  
Then
```

```

        Me.Activity.Selected(iListItemsCount) = True
        bFound = True
    End If
    iListItemsCount = iListItemsCount + 1
    Loop Until bFound = True Or iListItemsCount = Me.Activity.ListCount
    sValue = ""
Else
    sValue = sValue & sChar
End If

Next iCount
End Sub

```

Koodi joka lisättiin "NEW\_QUOTATION" lomakkeen "on Current" toimintoon<sup>[9]</sup>

```

Private Sub clearListBox()
    Dim iCount As Integer

    For iCount = 0 To Me.Activity.ListCount
        Me.Activity.Selected(iCount) = False
    Next iCount

End Sub

```

clearLisbBox() aliohjelma, jota kutsutaan "On Current" koodissa <sup>[9]</sup>

```

Private Sub Talleta_Click()
    Dim oltem As Variant
    Dim sTemp As String
    Dim iCount As Integer

    iCount = 0

    If Me.Activity.ItemsSelected.Count <> 0 Then
        For Each oltem In Me.Activity.ItemsSelected
            If iCount = 0 Then
                sTemp = sTemp & Me.Activity.ItemData(oltem)
                iCount = iCount + 1
            Else
                sTemp = sTemp & "," & Me.Activity.ItemData(oltem)
            End If
        Next oltem
    End If

```

```

iCount = iCount + 1
End If
Next oltem

```

```

Else
  MsgBox "Nothing was selected from the list", vbInformation
  Exit Sub 'Nothing was selected
End If

```

```

Me.Activity.Value = sTemp
DoCmd.Close acForm, "NEW_QUOTATION_Form", acSavePrompt
blRet = MouseWheelON
End Sub

```

Tallenna-painikkeen koodi <sup>[9]</sup>

Edelläkuvattujen koodien lisäämisen jälkeen ohjelma tallentaa valinnat tietokantaan muodossa valinta1, valinta2 jne.

View Quotations -lomakkeelta poitettiin tunti-lomakkeisiin johtaneet napit, koska ne olivat turhia (sieltä pääsi ainoastaan selaamaan tunteja, ei muokkaamaan niitä). Nyt ainut tapa tutkia tarjouksen tunteja on joko ajaa siitä raportti tai käydä editointitilassa katselemissa.

Myös Equipment-lomakkeeseen tuli muutoksia. Entisen yhden suuren kirjoitusalueen sijaan rakenne muuttui hieman enemmän tuntiensyöttö-lomakkeen tyyliksi. Lomakkeeseen tehtiin kolme "osiota" (Automaatio, Instrumentointi ja Sähköistys), joille kaikille tuli viisi riviä. Jokaisella rivillä on oma hintamarginaalivalintansa. Kirjoitetaan vain ensimmäiseen kenttään mitä tavaraa tarjotaan, ja seuraavaan kenttään kyseisen tavaran hinnan. Määrittellään marginaali ja paina Calculate and Save -nappia, joka laskee asiakkaalle tarjottavan hinnan ja tallentaa tiedot Equipments-tauluun. Kuva 8 esittää uudistetun Equipment-lomakkeen ulkoasun. Tarjouksen teon vaiheet käyvät ilmi liitteen 3 käyttöohjeesta, sekä tietokannan rakennetta kuvaava lohkokaavio löytyy liitteestä 4.

Kuva8 Uudistettu Equipment-lomake

Ennen tarjousten välillä pystyi liikkumaan "rullaamalla" hiirellä. Tämä ominaisuus osoitautui kuitenkin hankalaksi, koska käyttäjä saattoi vaihtaa muokattavaa tarjousta vahingossa (esim. yritettäessä siirtyä lomakkeella alaspäin). Accessin omista valikoista ei löytynyt asetusta, jossa voisi määrittää rullauksen toimintaa. Koska valikoista ei löytynyt mitään mainintaa kyseisestä ominaisuudesta, elvitettiin asiaa internetistä. Pian selvisikin, ettei (ainakaan) Access 2003-versiossa ole kyseiselle ominaisuudelle säätömahdollisuutta. Onneksi myös muilla on ollut sama ongelma. Erään keskustelupalstan kautta löytyi [www.lebans.com](http://www.lebans.com) nettisivut, jossa oli hiiren hallintaan tarvittava VBA-koodi MouseHook. Sivustoilta löytyy myös paljon muita hyödyllisen oloisia lisäosia, joita kannattaa käydä katsomassa. MouseHookin käyttäminen on melko yksinkertaista. Kopioidaan MouseHook.dll-tiedosto samaan kansioon tietokannan kanssa. Tehdään moduli, johon kopioidaan seuraavanlainen koodi:

```
Option Compare Database
```

```
Option Explicit
```

```
Private Declare Function LoadLibrary Lib "kernel32" _
```

```
Alias "LoadLibraryA" (ByVal lpLibFileName As String) As Long
```

```
Private Declare Function FreeLibrary Lib "kernel32" _
```

```
(ByVal hLibModule As Long) As Long
```

```
Private Declare Function StopMouseWheel Lib "MouseHook" _
```

```
(ByVal hWnd As Long, ByVal AccessThreadID As Long, Optional ByVal bIsGlobal As Boolean = False) As Boolean
```

```
Private Declare Function StartMouseWheel Lib "MouseHook" _
    (ByVal hWnd As Long) As Boolean
```

```
Private Declare Function GetCurrentThreadId Lib "kernel32" () As Long
```

```
' Instance returned from LoadLibrary call
```

```
Private hLib As Long
```

```
Public Function MouseWheelON() As Boolean
```

```
MouseWheelON = StartMouseWheel(Application(hWndAccessApp))
```

```
If hLib <> 0 Then
```

```
    hLib = FreeLibrary(hLib)
```

```
End If
```

```
End Function
```

```
Public Function MouseWheelOFF(Optional GlobalHook As Boolean = False) As Boolean
```

```
Dim s As String
```

```
Dim blRet As Boolean
```

```
Dim AccessThreadID As Long
```

```
On Error Resume Next
```

```
' Our error string
```

```
s = "Sorry...cannot find the MouseHook.dll file" & vbCrLf
```

```
s = s & "Please copy the MouseHook.dll file to your Windows System folder or into the same folder as this Access MDB."
```

```
' OK Try to load the DLL assuming it is in the Window System folder
```

```
hLib = LoadLibrary("MouseHook.dll")
```

```
If hLib = 0 Then
```

```
    ' See if the DLL is in the same folder as this MDB
```

```
    ' CurrentDB works with both A97 and A2K or higher
```

```
    hLib = LoadLibrary(CurrentDBDir() & "MouseHook.dll")
```

```
    If hLib = 0 Then
```



```

    MsgBox s, vbOKOnly, "MISSING MOUSEHOOK.dll FILE"
    MouseWheelOFF = False
    Exit Function
End If
End If

' Get the ID for this thread
AccessThreadID = GetCurrentThreadId()
' Call our MouseHook function in the MouseHook dll.
' Please not the Optional GlobalHook BOOLEAN parameter
' Several developers asked for the MouseHook to be able to work with
' multiple instances of Access. In order to accomodate this request I
' have modified the function to allow the caller to
' specify a thread specific(this current instance of Access only) or
' a global(all applications) MouseWheel Hook.
' Only use the GlobalHook if you will be running multiple instances of Access!
MouseWheelOFF = StopMouseWheel(Application.hWndAccessApp, AccessThreadID,
GlobalHook)

End Function

***** Code Begin *****
'Code courtesy of
'Terry Kreft & Ken Getz
'
Function CurrentDBDir() As String
Dim strDBPath As String
Dim strDBFile As String
    strDBPath = CurrentDb.Name
    strDBFile = Dir(strDBPath)
    CurrentDBDir = Left$(strDBPath, Len(strDBPath) - Len(strDBFile))
End Function
***** Code End *****

```

modMouseHook modulin koodi <sup>[10]</sup>

Tämän jälkeen lomakkeen aukaisevassa painikkeessa voidaan esitellä boolean-muuttuja `bIRet` (`Dim bIRet As Boolean`). Lomakkeen avausrivin jälkeen annetaan `bIRet`:lle arvo `bIRet=MouseWheelOFF(True)`. Mikäli halutaan, että muualla rulla toimii, lomakkeelta poistuttaessa annetaan muuttujalle arvo `False`. Tämä on erittäin hyödyllinen ominaisuus `Access`ille, koska ainakin itse yritän aina "rullata" työkalun sivuilla alaspäin, mikä johti sittemmin tahattomaan siirtymiseen väärään tilaukseen.

#### 4 YHTEENVETO

Työssä käytettiin menetelminä pääasiassa VBA-ohjelmointia, koska taustalla tapahtuvat automatisoinnit olivat usein helpompia toteuttaa koodin avulla. Joissakin tapauksissa, kuten joissakin suodattimissa, käytin Accessin kyselyitä (englanniksi Query).

Käytin työssäni paljon internetistä löytyvää materiaalia, koska kirjoja en tahtonut löytää mistään. Kuten jo aiemmin mainitsin, VBA:han liittyviä keskustelupalstoja on todella monia, ja niistä löytyy aika vaivattomasti esimerkkitapauksia ja ratkaisuja.

Työkalusta saatiin ensimmäisen version toimimaan. Työkalua kehitetään edelleen, ja siihen lisätään paljon ominaisuuksia. Jossakin vaiheessa tämä työkalu linkitetään mahdollisesti kaupalliseen asiakkuuksien hallintaohjelmistoon.

Työn aikana opin paljon automaatioprojektien tarjoustentekoprosessista ja aikataulute-  
tussa ohjelmistonkehitysprojektissa työskentelemisestä. Työn haastavuus tuli siitä, että aikaisemmin olen käyttänyt hyvin vähän Accessia.

Microsoft Accessilla pystyy tekemään kohtuullisen helposti tietokantasovelluksia. Monimutkaisempien tietokantojen kohdalla kannattaa opetella VBA-ohjelmointi, mielestäni tämä on pakollista, mikäli työkalun toimintojen halutaan olevan automaattisia.

Ohjelmasta tuli pilottiversiona toimiva ohjelmisto, johon tulee jatkossa vielä lisäominaisuuksia. Asiat olisi voinut toki tehdä eri tavallakin, mutta mielestäni käyttämäni tapa oli selkeä ja jatkojalostuksen kannalta hyvä, koska työstämäni koodia on helppo muokata.

Access sopii tällaisiin pilottiohjelmiin mielestäni varsin hyvin, koska se on kohtuullisen vapaasti muokattavissa. Ohjelmisto on myös melkein joka yrityksessä jo käytössä, joten lisenssiä ei tarvitse ostaa erikseen. Tästä johtuen myös käyttäjiä on olemassa paljon, joten internetissä on todella paljon keskustelupalstoja, joista saa apua moneen ongelmaan.



## LÄHTEET

1. Neste Jacobs (2010) *Neste Jacobsin WWW-sivut*, saatavissa: <http://www.nestejacobs.com> (luettu 27.09.2010)
2. Allen Browne (2006) *Microsoft Access Tips fo Casual Users*, saatavissa: <http://allenbrowne.com/casu-07.html> (luettu 04.10.2010)
3. Wikipedia vapaa tietosanakirja (2010) *Tietokanta*, saatavissa: <http://fi.wikipedia.org/wiki/Tietokanta> (luettu 06.11.2010)
4. Jyväskylän yliopiston IT-tiedekunan WWW-sivut (2004) *Relaatiotietokannat*, saatavissa: <http://appro.mit.jyu.fi/doc/tiedonhallinta/tietokannat/index1.html> (luettu 06.11.2010)
5. Jyväskylän yliopiston IT-tiedekunan WWW-sivut (2004) *Tietokannat*, saatavissa: <http://appro.mit.jyu.fi/doc/tiedonhallinta/tietokannat/index0.html> (luettu 06.11.2010)
6. Microsoft (2010) *Microsoft Office -työkalujen ohjesivut*, saatavissa: <http://office.microsoft.com/fi-fi/access-help/tietokannan-suunnittelun-perusteet-HA001224247.aspx> (luettu 06.11.2010)
7. Google Groups (2004) *Automated Monthly Report*, saatavissa: [http://groups.google.com/group/microsoft.public.access/browse\\_thread/thread/1fc2fe568c3c49d0/8666885a040da0e3?lnk=st&q=access+monthly+reports+automatically&rn um<BR>=3&hl=en#8666885a040da0e3](http://groups.google.com/group/microsoft.public.access/browse_thread/thread/1fc2fe568c3c49d0/8666885a040da0e3?lnk=st&q=access+monthly+reports+automatically&rn um<BR>=3&hl=en#8666885a040da0e3) (luettu 09.11.2010)
8. Kate J Chase; Scott Palmer, (2006) *Access 2003 for Starters: The Missing Manual* Saatavissa: [http://books.google.fi/books?id=FET0a2a\\_pWwC&printsec=frontcover&dq=Access+2003&hl=fi&ei=02PaTKyeFcOYOuTL3cAJ&sa=X&oi=book\\_result&ct=book-preview-link&resnum=8&ved=0CFkQuwUwBw#v=onepage&q&f=false](http://books.google.fi/books?id=FET0a2a_pWwC&printsec=frontcover&dq=Access+2003&hl=fi&ei=02PaTKyeFcOYOuTL3cAJ&sa=X&oi=book_result&ct=book-preview-link&resnum=8&ved=0CFkQuwUwBw#v=onepage&q&f=false)
9. Microsoft (2010) *Microsoftin tuotetuki*, saatavissa: <http://support.microsoft.com/kb/827423> (luettu 29.11.2010)
10. Lebans, Stephen (2009) *Access lisäosien hyödyllinen sivusto*, saatavissa: <http://www.lebans.com> (luettu 31.11.2010)
11. Asmala, Hannu; (SAMK): *Relaatiotietokannan normalisointi*, saatavissa: <http://www.tp.spt.fi/~salabra/ha/Relaatiotietokannat/normalisointi.html> (luettu 08.12.2010)
12. Mikkola, Matti(2010) *Access-vinkkejä*, saatavissa: <http://www.mattimikkola.com/vinkitaccess.htm> (luettu 08.11.2010)



## NESTEJACOBS

## Valid Quotations

Date	Quotation number	Customer	Project Description	Author	Work Hours	Total Price €	Engineering Price €	Equipment Price €	Expences (€)	Project Management	
24.11.2010	AE001-005	Watrec Ltd	Test 2	KASKEKA	2 926	903 067 €	123 413 €	22 364 €	22 364,00	756 800,00	
25.11.2010	AE001-006	Testifima Oy	Testi 3	AALTOHA	5 674	741 079 €	109 471 €	379 927 €	379 927,00	165 701,00	
<b>Sum:</b>						<b>8 600,00</b>	<b>1 644 146,00</b>	<b>232 884,00</b>	<b>402 291,00</b>	<b>402 291,00</b>	<b>922 501,00</b>





NESTEJACOBS

***Automation and Instrumentation Hours***Quotation number 

<i>Control and safety philosophy</i>	<input type="text" value="16790"/>
<i>F and G detection system philosophy</i>	<input type="text" value="200"/>
<i>Machinery protection system philosophy</i>	<input type="text" value="0"/>
<i>Field instrumentation philosophy</i>	<input type="text" value="0"/>
<i>Control Descriptions</i>	<input type="text" value="0"/>
<i>Control Diagrams</i>	<input type="text" value="0"/>
<i>Interlock descriptions</i>	<input type="text" value="0"/>
<i>Logic diagrams</i>	<input type="text" value="210"/>
<i>Sequence descriptions</i>	<input type="text" value="0"/>
<i>Sequence diagrams</i>	<input type="text" value="0"/>
<i>Process computer systems</i>	<input type="text" value="0"/>
<i>Simulators</i>	<input type="text" value="0"/>
<i>High level controls</i>	<input type="text" value="0"/>
<i>Database (SPI)</i>	<input type="text" value="0"/>
<i>Cabling and wiring engineering</i>	<input type="text" value="1500"/>
<i>Field instrument location engineering</i>	<input type="text" value="385"/>
<i>Control room and rack room layout engineering</i>	<input type="text" value="385"/>
<i>Calculations (Instrument sizing, power, instrument air, Exi)</i>	<input type="text" value="0"/>
<i>Instrument installation engineering</i>	<input type="text" value="0"/>
<i>Purchase requisitions</i>	<input type="text" value="300"/>
<i>SIL verification</i>	<input type="text" value="0"/>
<i>Job definitions</i>	<input type="text" value="0"/>
<i>System functional description</i>	<input type="text" value="0"/>
<i>Analyzer measurement arrangement drawings</i>	<input type="text" value="0"/>
<i>Sample line detail engineering</i>	<input type="text" value="0"/>
<i>Analyzer house engineering</i>	<input type="text" value="0"/>



<i>Purchase requisitions</i>	<input type="text" value="0"/>
<i>FAT</i>	<input type="text" value="0"/>
<i>DCS specification and purchase requisition</i>	<input type="text" value="0"/>
<i>DCS system implementation</i>	<input type="text" value="0"/>
<i>FAT</i>	<input type="text" value="0"/>
<i>SIS specification and purchase requisition</i>	<input type="text" value="0"/>
<i>SIS system implementation</i>	<input type="text" value="0"/>
<i>FAT</i>	<input type="text" value="0"/>
<i>Product tracking and recipe system specification</i>	<input type="text" value="0"/>
<i>Product tracking and recipe system implementation</i>	<input type="text" value="0"/>
<i>GDS specification and purchase requisition</i>	<input type="text" value="0"/>
<i>Site support during installation</i>	<input type="text" value="0"/>
<i>Functional testing of SIFs</i>	<input type="text" value="0"/>
<i>Precommissioning</i>	<input type="text" value="0"/>
<i>Review meetings</i>	<input type="text" value="0"/>
<i>Commenting, Checking, Cross checking</i>	<input type="text" value="0"/>
<i>Participation in quality audits</i>	<input type="text" value="0"/>
<i>Execution Planning</i>	<input type="text" value="0"/>
<i>Management</i>	<input type="text" value="0"/>
<i>Coordination meetings</i>	<input type="text" value="0"/>
<i>DCS</i>	<input type="text" value="Metso DNA"/>
<i>SIS</i>	<input type="text" value="Schneider Electric"/>
<i>I/O count</i>	<input type="text" value="1500"/>
<i>Hours Total</i>	<input type="text" value="4 185,00"/>
<i>Cost Total</i>	<input type="text" value="255 220,00"/>
<i>NJ Hours</i>	<input type="text" value="4 085,00"/>
<i>NJ Cost (€)</i>	<input type="text" value="248 720,00"/>
<i>Cons Hours</i>	<input type="text" value="100,00"/>
<i>Cons Cost (€)</i>	<input type="text" value="6 500,00"/>



Niko Pikkarainen

01.12.2010

## Tarjoustyökalun käyttöohje

### 1. Tarjouksen tekeminen

Uuden tarjouksen tekeminen aloitetaan painamalla aloituslomakkeella New Quotation -painiketta. Tämän jälkeen avautuu NEW\_QUOTATION -lomake, jossa on jo esitetyt päivämäärä ja tarjousnumero. Syötä tälle lomakkeelle asiakkaan perustiedot, minkä jälkeen paina Accept -painiketta. Mikäli et löydä asiakasta Firm -listasta, voit perustaa sen itse. Paina NEW CUSTOMER -painiketta, jolloin aukeaa lomake, jonka täyttämällä ja tallentamalla uusi asiakas ilmestyy firm listalle.

**HUOM!** Muista laittaa myös Expiration date, koska ohjelma poistaa automaattisesti Valid- täpän perustuen tähän päivämäärään.

Tämän itse tarjouksen laskeminen tapahtuu valitsemalla aloituslomakkeelta Edit Quotations -painike, jolloin avautuu Edit\_QUOTATIONS\_Form -lomake. Lomakkeen alareunassa on navigointipainikkeet, joilla voit liikkua tarjouksesta toiseen. Etsi tekemäsi tarjous. Voit muokata täällä myös perustietoja, mikäli huomaat niissä jonkin virheen. Tuntien syöttäminen tapahtuu klikkaamalla Engineerin price (internal) -kenttää, jolloin aukeaa väli-ikkuna. Valitse tästä, haluatko syöttää tunnit Automaatiolle ja instrumentoinnille vai Sähköistykselle.

Kun tuntiensyöttölomake aukeaa, täytetään ensin järjestelmä/moottori ym. kentät, jolloin saadaan kaikille kentille ilmestymään nolla alkuarvo. Mikäli tehdään toisessa järjestyksessä (Eli Final Hours kenttä on täysin tyhjä) ei ohjelma osaa siirtää tarjousnumeroa ja työvaihenumeroa tuntien jaottelu lomakkeelle. Kun olet syöttänyt edellä mainitut tiedot (nollat ovat ilmestyneet Final Hours -kenttiin), voidaan aloittaa itse tuntien syöttäminen.

Klikkaa Final Hours -kenttää, jolloin aukeaa lomake, jossa on jaoteltu kyseinen työvaihe Groupeille. Syötä tunnit tähän lomakkeeseen (kun painat Enteriä, siirryt seuraavaan laatikkoon, ja oletusarvo "maalautuu" automaattisesti). Toista tämä kaikille tarvittaville työvaiheille.

Kun kaikki lomakkeen tunnit on syötetty, voidaan painaa Calculate And Save -painiketta, joka nimensä mukaisesti laskee tunnit, kustannukset ja hinnat, sekä tallentaa ne. Tämän jälkeen lomakkeelta voi poistua painamalla "ovi" -painiketta.

Edit\_QUOTATIONS\_Form -lomakkeella on samankaltaiset syöttämistoiminnot myös Equipment -kentässä, Expences -kentässä ja Project Management -kentässä. Kun olet syöttänyt kaikki tunnit ja hinnat, klikkaa Total Price ja Rate Price -kenttiä, jolloin nämä päivittyvät.

Mikäli Rate Price ei ole hyvä, voit hienosäätää sitä laittamalla täpän Fix the Rate Price with Margin-laatikkoon, jolloin lomakkeelle ilmestyy kenttä, johon voit syöttää marginaaliprosentin (muista valita myös haluatko lisätä vai vähentää tuon marginaalin verran). Marginaali vaikuttaa Engineerin Price (internal) -summaan, ja sitä kautta Rate Priceen.



Niko Pikkarainen

01.12.2010

Kun halutut muutokset on tehty hyväksytään painamalla Accept -painiketta.

Tällä samalla tavalla voit myös muokata mitä tahansa aiemmin syötettyä tarjousta

## 2. Vanhojen tarjousten selaaminen

Aloitussivulla painamalla View and Edit All Quotations -painiketta, avautuu NJ\_AE\_Quotation -lomake (samaa aikaan ohjelma tarkistaa kaikkien tarjousten voimassa olon). Lomakkeella on lista kaikista tarjouksista, voit tarkastella ja muokata tietoja täällä.

Tässä näkyvässä käydään kuittaamassa tarjous tilatuksi tai hävityksi.

## 3. Raporttien tulostus

Päälomakkeelta valitsemalla Report selection, päästään lomakkeelle, jossa on Report type -valikko. Tästä valitsemalla voidaan tulostaa kuukausiraportti, tuntiraportti tai sitten osia kuukausiraportista (edellisen kuukauden voimassa olevat tarjoukset, hävityt tarjoukset, sekä tilatut tarjoukset).

### Monthly Report

Avaa esikatseluun edellisen kuukauden voimassa olevat tarjoukset, tilatut tarjoukset, sekä hävityt tarjoukset.

### Project hours report

Haku lomakkeella tulee esiin Quotation number niminen laatikko, josta valitsemalla tarjousnumero, saadaan kyseisen tarjouksen automaation ja instrumentoinnin tunnit, sähköosaston tunnit sekä projektin hallinnan tunnit.

### Valid Quotations

Tuo edellisen kuukauden voimassa olevat tarjoukset

### Ordered Quotations

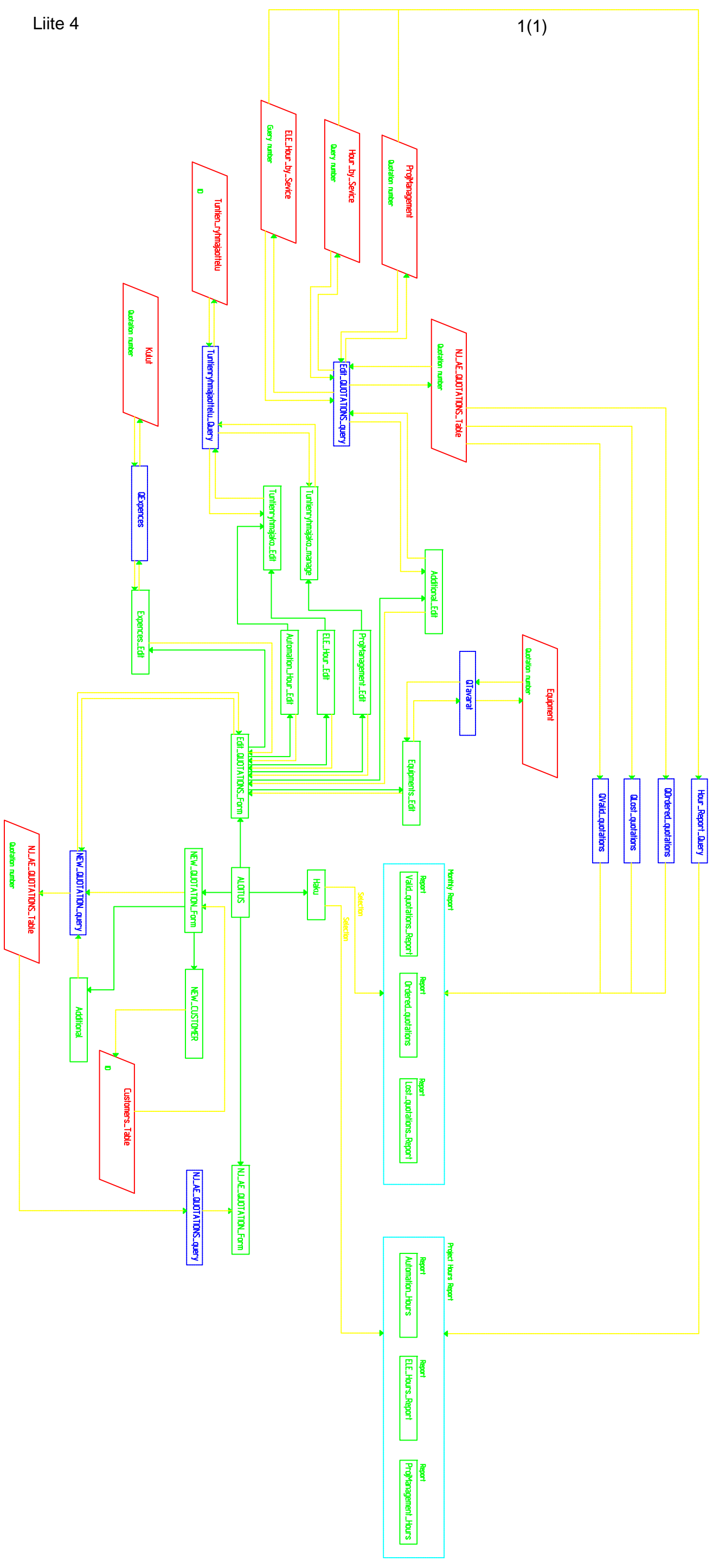
Tuo edellisen kuukauden tilatut tarjoukset

### Lost Quotations

Tuo edellisen kuukauden hävityt tarjoukset







Explanations of the symbols:

- ▭ Name of the Table  
Primary Key Field
- ▭ =Table
- ▭ Name of the Query
- ▭ =Query
- ▭ Name of the Report
- ▭ =Report
- ▭ Name of the Form
- ▭ =Form
- ↔ =Link between Forms
- ↔ =Data transfer



---

[www.savonia.fi](http://www.savonia.fi)

