

Teemu Mikkonen

## **PINTA-ALAN PIIRTOTYÖKALUN JATKOKEHITYS**

## **PINTA-ALAN PIIRTOTYÖKALUN JATKOKEHITYS**

Teemu Mikkonen  
Opinnäytetyö  
Kevät 2016  
Tietotekniikan koulutusohjelma  
Oulun ammattikorkeakoulu

## TIIVISTELMÄ

Oulun ammattikorkeakoulu

Tietotekniikan koulutusohjelma, ohjelmistokehityksen suuntautumisvaihtoehto

---

Tekijä(t): Teemu Mikkonen

Opinnäytetyön nimi: Pinta-alan piirtotyökalun jatkokehitys

Työn ohjaaja: Pertti Heikkilä

Työn valmistumislukukausi- ja vuosi: Kevät 2016

Sivumäärä: 25

---

Opinnäytetyö toteutettiin Piimega Oy:lle, joka on oululainen ohjelmistotalo. Piimega kehittää räätälöityjä toiminnanohjausjärjestelmiä yrityksille sekä metsä- ja sahateollisuuteen. Opinnäytetyö toteutettiin Piimega Total -toiminnanohjausjärjestelmään. Piimega Total -toiminnanohjausjärjestelmä on verkkosivuston kautta toimiva toiminnanohjausjärjestelmä, joka tarjoaa yritykselle kaikki tarvittavat komponentit toimintansa ohjaamiseen aina asiakkaiden hallinnasta itse tuotteen valmistukseen ja toimitukseen asiakkaalle. Piimega Total -järjestelmä käsittää myös tarjouslaskentatyökalun, jonka avulla käyttäjä voi laatia tarjouksen urakasta tai projektista. Opinnäytetyössä käsiteltävä piirtotyökalu on osa tätä tarjouslaskentatyökalua. Piirtotyökalun avulla käyttäjä kykenee piirtämään kuvaajan, josta työkalu laskee sitten tarvittavia määreitä tarjouslaskentaa varten.

Opinnäytetyön tavoitteena oli jatkokehittää pinta-alan piirtotyökalua asiakkaan tarpeen ja palautteen perusteella. Työn teoriaperusta koostuu Total-toiminnanohjausjärjestelmän ohjelmointiin liittyvistä .NET-ominaisuuksista ja ohjelmistokirjastoista sekä piirtotyökalun perustoimintaperiaatteesta.

Opinnäytetyön kirjoittamiseen on käytetty hyvin pitkälti Microsoftin virallisia ASP.NETiin liittyviä lähteitä niiden tarkkuuden ja monipuolisuuden ansiosta. Opinnäytetyön työosuudessa saatiin onnistuneesti luotua asiakasta hyödyttäviä ominaisuuksia, jotka nopeuttavat ja helpottavat asiakkaan toimintaa tulevaisuudessa. Asiakasyritys käyttää piirtotyökalun uusia ominaisuuksia jokapäiväisesti tarjouksen syötössään.

Piirtotyökalua on kehitetty opinnäytetyön työosuuden valmistumisen jälkeen monilla tavoin ja sitä tullaan mahdollisesti vielä jatkokehittämään asiakkaan tarpeiden mukaisesti.

---

Asiasanat: ASP.NET, piirtotyökalu, tarjouslaskenta, pinta-alan laskentatyökalu

## ABSTRACT

Oulu University of Applied Sciences  
Bachelor of computer sciences, software development

---

Author(s): Teemu Mikkonen

Title of thesis: Drawing tool development

Supervisor(s): Pertti Heikkilä

Term and year when the thesis was submitted: Spring 2016      Number of pages: 25

---

The thesis was done for Piimega Oy which is a software house from Oulu. Piimega develops tailored Enterprise Resource Planning systems for companies and also for lumber and timber industry. This thesis was made for Piimega Total product. Total is a ERP which functions trough web portal and gives the customer all the customer company needs for its operation, from CRM to production and delivery to customer. Total ERP also includes an offer calculation tool where you can create offer of a project or a contract and the tool calculates every needed value for the user. The drawing tool which further development is part of this thesis is part of this offer calculation tool. With the drawing tool user can draw a picture and the tool calculates the area of the drawn picture and other useful values.

The purpose of this thesis was to develop needed features for the drawing tool. Theory base consists of .NET features used in programming and general class libraries in .NET and also of the basic functionality of the drawing tool.

Microsoft's ASP.NET sources were mostly used writing this thesis because these sources consists of very reliable and accurate data of the classes and functionality of ASP.NET. In the practical part of this thesis new features were successfully developed for the customer company. The new features make customers offer calculation much more fast and easy. Customer company is using the new features of the drawing tool daily while calculating new offers.

The drawing tool will probably be further developed many ways by the customers' needs after this thesis practical part has finished.

---

Keywords: ASP.NET, drawing tool, offer calculation, area calculation tool

# SISÄLLYS

TIIVISTELMÄ.....	3
ABSTRACT.....	4
SISÄLLYS.....	5
1 JOHDANTO.....	6
2 .NET -KOMONENTTIKIRJASTO.....	7
2.1 Visual Basic .NET.....	8
2.2 ASP ja ASP.NET.....	8
2.2.1 Web-sivustot.....	9
2.2.2 MVC-arkkitehtuuri.....	10
2.2.3 WebForms.....	10
3 PINTA-ALAN PIIRTOTYÖKALU.....	11
3.1 Perustoiminnot.....	12
3.2 Toimintaperiaate.....	13
4 JATKOKEHITYS.....	16
4.1 Viivan valinta ja muokkaus.....	16
4.2 Kuvan muokkaus.....	19
4.3 Aukot ja poikkeavat seinäalat.....	20
5 POHDINTA.....	23
LÄHTEET.....	24

# 1 JOHDANTO

Opinnäytetyö tehtiin Piimega Oy:lle, joka on keskittynyt tuottamaan tuotteistettuja toiminnanohjausratkaisuja yrityksille sekä metsä- ja sahateollisuuteen. Suoritin opintoihini kuuluvan työharjoittelun ja kesäharjoittelun Piimega Oy:ssä, joten opinnäytetyön tekeminen samaan yritykseen oli luonnollinen jatkumo. Itse opinnäytetyö liittyi Piimega Oy:n Total-tuotteeseen, joka on yrityksen toiminnanohjaukseen keskittyvä ohjelmistokokonaisuus. Total-tuote tarjoaa toimivat moduulit tarjouslaskennasta valmistuksen kautta toimitukseen ja tuotantoon ja varastonhallintaan saakka.

Opinnäytetyön tavoitteena oli jatkokehittää asiakasyritykselle kehitettyä piirtotyökalua asiakkaan tarpeiden mukaan. Piirtotyökalu on osa Total-tuotteen tarjouslaskentakokonaisuutta, minkä avulla asiakasyritys kykenee laskemaan realistisen tarjouksen tuotteestaan suoraan asiakkaalleen. Tarjouslaskenta summaa jokaisen tarjouksessa käytetyn elementin ja sen rakenneosasen. Näin ollen tarjouslaskentatyökalulla saadaan laskettua tarkka määrä, kuinka paljon raaka-aineita ja elementtejä tarvitaan tuotteen tuotantoa varten. Näin tarjouslaskentatyökalulla saadaan laskettua realistinen lopputuotteen hinta, paljonko tuotteen valmistus maksaa yritykselle.

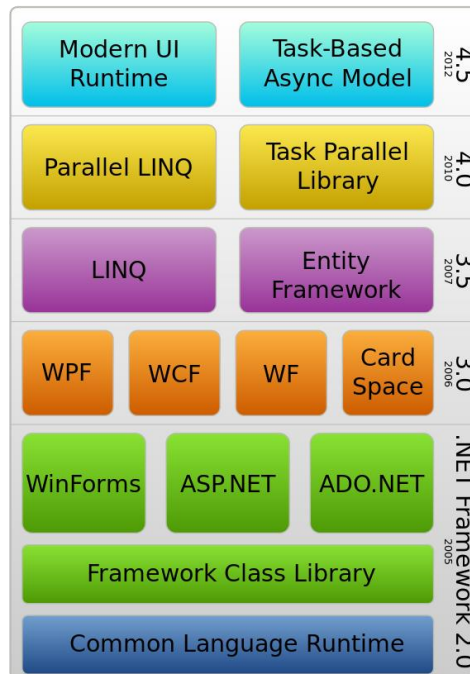
Piirtotyökalun avulla käyttäjä kykenee piirtämään kuvaajan ja työkalu laskee kuvaajalle pinta-alan ja muita tarjouksella tarvittavia arvoja. Työkalua oli jo alustavasti kehitetty harjoitteluprojektien aikana niin, että perusominaisuudet eli viivan piirtäminen ja pinta-alan laskeminen piirretyn kuvaajan perusteella onnistuivat. Opinnäytetyössäni keskityn enemmän lisäominaisuuksiin, jotka todettiin piirtotyökalun perusominaisuuksien kehittämisen jälkeen tarpeelliseksi tarjouslaskennan sujuvan ja tehokkaan käytön kannalta.

Piirtotyökalun jatkokehitys toteutettiin vanhan piirtotyökalun pohjalta ja siinä käytettiin ASP.NET-kehitysympäristöä, joka on Microsoftin kehittämä kehitysympäristö verkkosivustojen luontiin. Itse piirtotyökalu kehitettiin tarjouslaskentaan niin, että kun käyttäjä lisää komponentin, hänellä on mahdollisuus avata piirtotyökalu. Piirtotyökalu aukeaa ponnahdusikkunana tarjouslaskennan päälle. Piirtämisen jälkeen käyttäjä sulkee piirtotyökalun, jolloin piirtotyökalu palauttaa lasketut arvot tarjouslaskentatyökalulle.

## 2 .NET -KOMPONENTTIKIRJASTO

.NET-komponenttikirjasto on Microsoftin kehittämä ohjelmistokehys, joka on kohdennettu lähinnä Windows-alustalle. Laajan luokkakirjastonsa ja eri ohjelmistokielen välisen toimivuutensa ansiosta .NET on loistava alusta monenlaiseseen ohjelmistokehitykseen. NET-ohjelmistokirjasto koostuu kahdesta osasta: itse ohjelmistokirjasto FCL:stä (Framework Class Library) ja ohjelman kääntävästä virtuaaliohjelmistosta CLR:stä (Common Language Runtime) ( kuva 1). FCL:n laaja ohjelmistokirjasto sisältää laajat luokat mm. käyttöliittymiin, datan käsittelyyn, algoritmeihin, verkkosivustojen kehittämiseen sekä tietoliikenneyhteyksiin. Laajan luokkakirjaston lisäksi FCL tarjoaa yhteentoimivuuden useiden eri kielten välille, jotka on kirjoitettu käyttäen .NET-ohjelmistokehystä. (1.)

CLR on .NET-ohjelmistokehysten suorittava ja kääntävä virtuaalinen ohjelmisto, joka kääntää ja huolehtii käännettävän ohjelmiston ajamisesta ja yhteensopivuudesta järjestelmän kanssa. Ohjelman kirjoituskielestä riippumatta kehitettävä ohjelmisto suoritetaan aina CLR :n avulla. CLR sisältyy jokaiseen .NET-versioon. (2.)



KUVA 1. .NET ohjelmistokehys versioittain (3.)

## 2.1 Visual Basic .NET

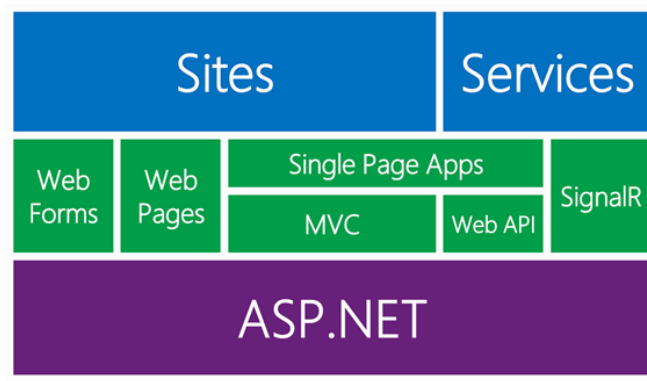
Visual Basic .NET on korkean tason ohjelmistokielen Visual Basicin seuraaja, joka julkaistiin 2002 yhdessä .NET ohjelmistokehityksen kanssa. Yhdessä Visual C#:n kanssa se on .NET-ohjelmistokehityksen käytetyimpiä kieliä. Visual Basic .NET eroaa monelta osalta edeltäjästään ja uusi versio onkin rakennettu kokonaan käyttämään CLR-ympäristöä. Tämän takia ohjelmat, jotka ovat kirjoitettu vanhalla Visual Basic -kielellä, eivät ole suoraan yhteensopivia .NET-alustalle, vaan ne on ensin muokattava käyttämään .NET-kirjastoja. (4.)

## 2.2 ASP ja ASP.NET

ASP.NET on Microsoftin avoimen lähdekoodin verkkosivujen ohjelmistokehitys verkkosivustojen ja ohjelmistojen kehitykseen ja se on seuraaja suosituille ASP (Active Server Pages) -teknologialle, joka oli Microsoftin ensimmäisiä palvelimella suoritettavia skriptejä dynaamisten verkko-sivustojen luontiin. Normaali verkkosivu on yleensä staattinen, sinne on kovakoodattu elementit ja ne pysyvät samana riippumatta käyttäjän tekemistä valinnoista ja klikkauksista. ASP.NET-ohjelmistokehityksen avulla käyttäjä voi luoda dynaamisia verkkosivustoja, jotka mukautuvat käyttäjän valintojen ja kommentojen mukaan. Esimerkiksi käyttäjä voi suodattaa taulukossa näkyvää tietoa ja ohjelmisto hakee uuden tiedon reaaliaikaisesti tietokannasta. ASP.NET julkaistiin jo vuonna 2002 osana .NET-ohjelmistokehitystä. ASP.NET on muiden .NET-komponenttien tapaan rakennettu CLR-komponentin päälle, mikä mahdollistaa sen, että myös ASP.NET-ohjelmat tukevat kaikkia .NET-tuettuja kieliä kuten C# ja Visual Basic .NETiä. (5.)

ASP.NETillä luodut web-ohjelmistot rakentuvat kolmesta eri ohjelmistokehityksestä: ASP.NET-verkkosivuista, WebForms-Windows-ohjelma ohjelmistokehityksestä ja MVC (Model-View-Controller) -ohjelmistokehityksestä (kuva 2). Jokainen näistä ohjelmistokehityksistä tarjoaa omat vahvuutensa ASP.NET-ohjelmistolle ja parasta onkin, että kaikkia näitä komponentteja ja niiden eri vahvuuksia on mahdollista käyttää sekaisin ASP.NET-ohjelmistossa. (6.)





KUVA 2. ASP.NET-kirjaston poikkileikkaus (7.)

### 2.2.1 Web-sivustot

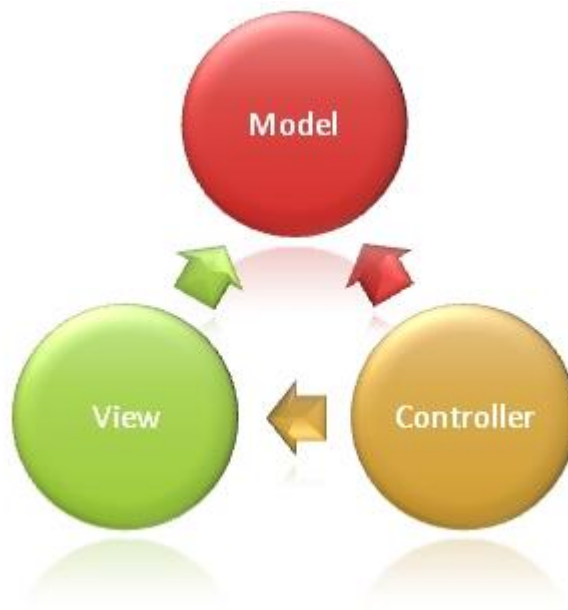
ASP.NET-verkkosivustot on yksinkertaisin ASP.NET-ohjelmistokehyksistä. Sen avulla voidaan luoda dynaamisia verkkosivustoja yhdistelemällä JavaScriptiä, palvelimella suoritettavaa ohjelmistokoodia, eri muotoilukieliä kuten CSS:ää sekä HTMLää ja PHP:ta yhdeksi kokonaisuudeksi. Yksinkertaisimmillaan sivusto voi sisältää vain HTML- ja ASP.-koodia (kuva 3). Esimerkin tapauksessa syntaksi on hyvin pitkälti HTML-koodia, <html>-komennolla aloitetaan HTML-koodin sisältö. <body>-komento aloittaa itse html-sisällön, <h1>-komento tarkoittaa suurinta otsikkoa ja tähän sijoitetaan tekstirivi "Hello Web Pages". <p>-komento tarkoittaa kappaletta, johon sijoitetaan tekstirivi "The time is" sekä ASP.NET-koodi komennolla "@DateTime.Now" haettava sen hetkinen aika, joka sijoitetaan heti tekstirivin perään. Jokainen komento suljetaan vastaavalla komennolla mutta lisäämällä kenoviiva komennon sisään, esimerkiksi "</p>". (8;9.)

```
<!DOCTYPE html>
<html>
<body>
  <h1>Hello Web Pages</h1>
  <p>The time is @DateTime.Now</p>
</body>
</html>
```

KUVA 3. Esimerkki yksinkertaisesta verkkosivuston syntaksista

## 2.2.2 MVC-arkkitehtuuri

MVC (Model-View-Controller) -arkkitehtuuri tulee sanoista malli-näkymä-ohjain ja se jakaantuu näihin kolmeen osaan (kuva 4). Ohjain-osio käsittelee kaikki käyttäjän syötteet ja lähettää ne eteenpäin mallille tai suoraan näkymälle. Malli-osio hakee datan käyttäjän syötteen mukaisesti ja käsittelee sitä näkymä-osiota varten. Sen jälkeen se lähettää tiedon näkymä-osiolle, joka käsittelee ja sijoittaa datan käyttäjälle näkyvään muotoon. Kuten web-sivutkin, MVC tarjoaa myös täyden kontrollin CSS:ään, JavaScriptiin sekä HTML:ään. (10; 11.)



KUVA 4. Havainnollistus MVC-mallin toiminnasta (12.)

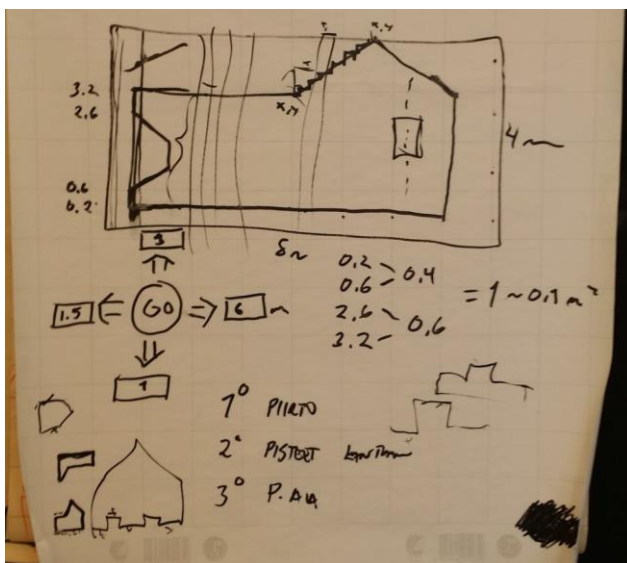
## 2.2.3 WebForms

WebForms on tärkeä osa ASP.NET-ohjelmistokehystä ja sen avulla on nopea ja helppo kehittää dynaamisia verkkosivustoja. WebForms sisältyy Microsoft Visual Studio IDE:hen (Integrated Development Environment), jota käyttämällä kontroleja on helppo siirtää hiiren avulla suunniteltavalle sivulle. WebFormsin avulla voit myös helposti säätää kontrollien asetuksia ja tapahtumakutsuja mitä sivustolle halutaan tulevan. WebForms on tapahtumapohjainen ohjelmointimalli. Kun käyttäjä pyytää sivua palvelimelta, sivu käännetään ja suoritetaan suoraan palvelimella. Tämän jälkeen generoidaan HTML-koodi, jota selain tulkitsee ja esittää käyttäjälle verkkosivuston muodossa. WebForms tukee kaikkia mahdollisia selaimia ja laitteita. Näin ollen se on hyvä ja nopea tapa kehittää verkkosivustoja mille tahansa laitteelle. (13; 14.)

### 3 PINTA-ALAN PIIRTOTYÖKALU

Pinta-alan piirtotyökalu rakennettiin osaksi PiiMega Total -tuotteen tarjouslaskentakomponenttia. Total-toiminnanohjausjärjestelmä on koodattu käyttämällä Visual Basic .NET -ohjelmistokieltä ja ASP.NET-ohjelmistokehystä. Piimega Total -toiminnanohjausjärjestelmän luomiseen on käytetty lähinnä ASP.NET-verkkosivukomponentteja ja sen käyttöliittymä onkin suunniteltu verkkoselaimella käytettäväksi. Total-toiminnanohjausjärjestelmään kuuluu myös Windows-pohjainen WebForms-ohjelma, joka on suunnattu enemmän asetusten määrittämiselle ja tuotteiden tuotannon hallintaan.

Pinta-alan piirtotyökalu suunniteltiin nopeuttamaan asiakasyrityksen tarjouslaskentaa (kuva 5). Piirtotyökalulla asiakas voi piirtää haluamansa seinäalueen piirtotyökalun kontrolleja käyttämällä. Tällöin työkalu laskee piirretyn alueen pinta-alan ja joitain muita laskennassa tarvittavia arvoja, kuten pohja- ja yläjuoksuja (kuva 6). Piirtäminen nopeuttaa ja helpottaa asiakasyrityksen tarjouslaskennan käyttöä huomattavasti. Käsien pinta-alojen ja laskennassa käytettävien juoksujen laskeminen on paljon hitaampaa ja työläämpää kuin piirtotyökalulla koko seinäalueen piirtäminen. Piirtämisen jälkeen käyttäjä painaa Valmis-painiketta, jolloin piirtotyökalu palauttaa lasketut arvot Total-tarjouslaskentaan, joka käyttää ja laskee palautettujen arvojen mukaisesti eri nimikkeitä ja tarvikkeita tarjoukselle. Myöhemmin sopimuksen hyväksymisvaiheessa näitä arvoja käytetään nimikkeiden määrien laskemiseen tuotantoa ja toimitusta varten.



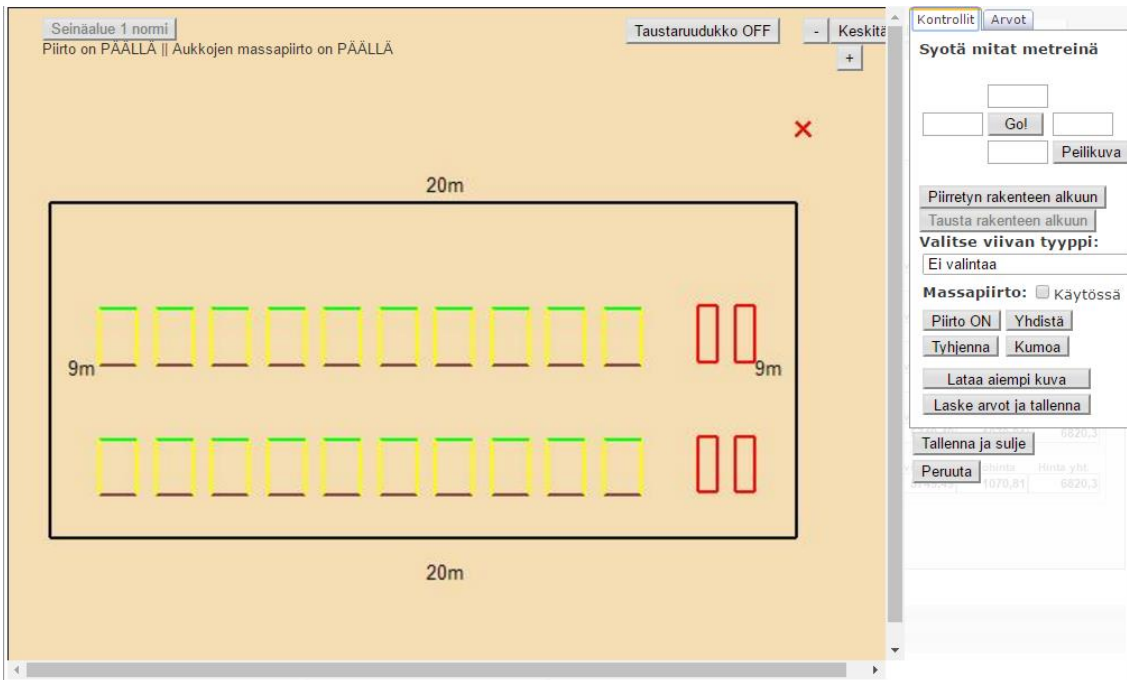
KUVA 5. Piirtotyökalun suunnittelupalaverin muistiinpanoja



*KUVA 6. Esimerkki punatiilisestä seinäalasta, jonka hinnan laskemiseen asiakasyritys käyttää piirtotyökalua*

### **3.1 Perustoiminnot**

Piirtotyökalun kaikki kontrollit keskittyvät piirtämiseen ja kursorin hallintaan (kuva 7). Ennen jatkokehitystä piirto tapahtui käyttämällä tekstilaatikoita Piirrä-painikkeen ympärillä sekä Yhdistä ja tyhjennä -painiketta käyttämällä. Myös Laske arvot ja tallenna -painike on hieman muuttunut jatkokehityksen yhteydessä, mutta perustoiminnaltaan sisältää hyvin pitkälti saman laskentalogiikan kuin ennen jatkokehitystä. (Kuva 7.) Tallenna ja sulje -painikkeesta työkalu palauttaa pinta-alan ja muut arvot tarjouslaskennalle ja tallentaa piirretyn kuvaajan tietokantaan. Kursorin piirto on mahdollista laittaa päälle ja pois Piirto ON/OFF -painikkeesta. Tällöin piirron nykyinen tila päivittyy näkyviin piirtoalueen vasempaan yläkulmaan. Yhdistä-painikkeella piirtotyökalu piirtää automaattisesti viivan rakenteen aloituspisteeseen.



KUVA 7. Piirtotyökalun jatkokehityksen jälkeinen käyttöliittymä. Kuvan 6 julkisivu piirrettyä piirtotyökalulla.

### 3.2 Toimintaperiaate

Piirtotyökalun tarkoituksena on pystyä laskemaan pinta-ala minkä tahansa muotoiselta ja tyypiseltä piirretyltä alueelta. Helpoiten tämä onnistuu käyttämällä matematiikkaa ja integraalia. Piirretty kuvaaja pilkotaan suikaleisiin, jolloin lasketaan kahden viivan välistä matkaa alhaalta ylös. Suikaleen alueella etsitään alueella olevat piirretyjen viivojen pisteet ja lasketaan niiden välinen korkeus ja kerrotaan se suikaleen leveydellä. Näin saadaan suikaleen pinta-ala. Piirretty kuvaaja käydään läpi suikale suikaleelta, vasemmalta oikealle, kunnes koko kuvaaja on käyty läpi ja saatu koko kuvaajan alue laskettua. Virhemarginaalia kyetään vähentämään kaventamalla mitattavan suikaleen leveyttä sivuttaissuunnassa, mutta tämä toisaalta hidastaa laskentaa, sillä läpikäytäviä alueita on useampia.

Piirtotyökalussa jokainen viiva on tallennettu datatable-tyyppiseen taulukkomuuttujaan. Taulukossa on tallessa viivan alku- ja päätepiste sekä tieto viivan tyypistä eli siitä, onko kyseessä päarakenteen sisällä oleva aukko vai onko viiva rakenteen ulkoreuna eli osa päarakennetta. Pinta-alan lasku aloitetaan luomalla pääsilmukka, joka käy läpi kuvan vasemmalta oikealle tiedetyin välein (kuva 8). Tämän pääsilmukan sisällä tarkastellaan piirretyt viivat määritetyltä alueelta. Mikäli jokin viivan piste on alueella, otetaan se talteen toiseen datatable-tyyppiseen muuttujaan. Tämän jälkeen

kerätyt viivat järjestetään omassa silmukassaan niin, että ne ovat korkeusjärjestyksessä. Kun viivat on saatu korkeusjärjestykseen, on niitä helppo vertailla myöhemmässä vaiheessa pareittain ja laskea niiden välinen etäisyys.

```
'Tsekataan kuva leveysuunnassa palkeittan, palkin leveys = Dim tarkkuus
For iPx1 As Double = 0.0 To CDBl(iKuvanLeveys) Step tarkkuus
  For Each dr As DataRow In dtViivat.Rows
    'jos viiva palkin alueella lisätään väliaikaiseen tauluun
    If (dr.Item("x") < iPx1 And iPx1 < dr.Item("x2")) Or (dr.Item("x") < iPx1 + tarkkuus And iPx1 + tarkkuus < dr.Item("x2")) _
      Or (dr.Item("x2") < iPx1 And iPx1 < dr.Item("x")) Or (dr.Item("x2") < iPx1 + tarkkuus And iPx1 + tarkkuus < dr.Item("x")) Then
      dtOsuneetViivat.ImportRow(dr)
    End If
  Next

  'Yritetään saaha oikeaan järjestykseen osuneet viivat että saadaan pareittain päällekkäiset viivat
  For Each dr As DataRow In dtOsuneetViivat.Rows
    'jos vinoviiva ni laskeskellaan y:n paikka
    If dr("y") <> dr("y2") And dr("x") <> dr("x2") Then
      Dim k1 As Double = -((dr("y2") - dr("y")) / (dr("x2") - dr("x")))

      If dr("x") > dr("x2") Then
        dr("Jarjestys") = (k1 * (dr("x") - dr("x") + (dr("x") - iPx1))) + dr("y")
      Else : dr("Jarjestys") = (k1 * (dr("x2") - dr("x2") + (dr("x2") - iPx1))) + dr.Item("y2")
      End If
    Else
      dr("Jarjestys") = dr("y")
    End If
  Next
```

KUVA 8. Pinta-ala laskenta silmukan alku

Kun viivat on datataulussa saatu järjestykseen, täytyy kahden viivan välinen välimatka korkeussuunnassa saada laskettua. Tätä varten viivat lisätään väliaikaiseen tauluun. Kun väliaikaisessa taulussa on kaksi jäsentä, aloitetaan viivojen kahden pisteen välisen etäisyyden laskeminen. Viivojen välinen välimatka lasketaan laskemalla viivan koordinaatti kyseiseltä kohdalta, missä ollaan kuvaajassa sivuttaissuunnassa menossa. Kun viivan aloituspiste y ja päätepiste y2 ovat samoja, on viiva vaakasuora ja sen korkeus suhteessa origoon on sama koko viivan matkalta. Kun viivan aloituspiste x ja päätepiste x2 ovat samoja, on viiva pystysuora ja sen korkeus on suoraan laskennassa tarvittava korkeus (kuva 9). Vinon viivan koordinaatin selvittämiseksi tarvitaankin hieman enemmän matematiikkaa. Kulmakertoimen kaavalla  $k = (y2 - y1) / (x2 - x1)$  saadaan laskettua vinon viivan kulmakerroin, missä k on kulmakerroin ja x ja y merkkäavat viivan pisteitä. Kulmakertoimen avulla tiedettäessä viivan aloitus- ja päätepiste voidaan laskea viivan y-koordinaatti halutussa kohtaa viivaa käyttämällä matemaattista kaavaa  $y2 = k \cdot (x2 - x1) + y1$  (kuva 10). Näin saadaan vinon viivan y-koordinaatti. Tätä y-koordinaattia verrataan toisen viivan y-koordinaattiin, mistä saadaan korkeus laskettua. Korkeus kerrottuna laskettavan alueen leveydellä saadaan aikaiseksi sen kohdan pinta-ala (kuva 11 ja kuva 12), joka lisätään kokonaispinta-alaan. Näin koko kuva läpikäymällä saadaan laskettua piirretyn kuvaajan pinta-ala.

```

'järjestetään viivat
tmpJarjesta = New DataView(dtOsuneetViivat)
tmpJarjesta.Sort = "Jarjestys ASC" ' "y ASC"
'kahtellaan pystysuunnassa viivojen väliin jäävä alue osuneet taulusta
For Each dr2 As DataRowView In tmpJarjesta
    'Kahtellaan viivat aina pareittain, viivat korkeusjärjestyksessä joten aina kahden viivan välein laskeskellaan pinta-alaan lisää
    If drTmpYViiva.Rows.Count = 0 Then
        drTmpYViiva.LoadDataRow(dr2.Row.ItemArray, True)
    Else
        'Lisää pinta-alaan saatu rajattu alue
        Dim alueKorkeus As Double = 0
        Dim k1, k2 As Double
        Dim yTmp As Double = 0
        Dim yTmp2 As Double = 0
        drTmpYViiva.LoadDataRow(dr2.Row.ItemArray, True)

        'Ensimmäinen viiva
        'jos viiva vaakatasossa otetaan sen y-koordinaatti, jos pystytasossa otetaan alku ja päätepiste, jos vinottain lasketaan kulmakerroin
        If drTmpYViiva.Rows(0).Item("y2") = drTmpYViiva.Rows(0).Item("y") Then
            yTmp = drTmpYViiva.Rows(0).Item("y")
            bPalkit = True 'tällä hetkellä palkit vain vaakasuorilta pinnoilta
            'pystyviiva
        ElseIf drTmpYViiva.Rows(0).Item("x2") = drTmpYViiva.Rows(0).Item("x") Then
            yTmp = drTmpYViiva.Rows(0).Item("y")
            yTmp2 = drTmpYViiva.Rows(0).Item("y2")
            If yTmp > yTmp2 Then
                alueKorkeus = yTmp - yTmp2
            Else : alueKorkeus = yTmp2 - yTmp
            End If
            'kulmakerroin vinolle viivalle kaavalla k = (y2-y1)/(x2-x1)
        ElseIf drTmpYViiva.Rows(0).Item("y2") <> drTmpYViiva.Rows(0).Item("y") And drTmpYViiva.Rows(0).Item("x2") <> drTmpYViiva.Rows(0).Item("x") Then
            k1 = -((drTmpYViiva.Rows(0).Item("y2") - drTmpYViiva.Rows(0).Item("y")) / (drTmpYViiva.Rows(0).Item("x2") - drTmpYViiva.Rows(0).Item("x")))
        End If
    End If

```

KUVA 9. Pinta-ala laskentasilmutta viivojen välisen korkeuden laskemiseen

```

' Jos vino viiva Saahaan laskettua alueen kohdalla oleva y kulmakertoimen kaavasta k = y2-y1 / x2-x1, tiedetään k,y1,x2,x1 :laske y2 | * x2-x1 -> y2 = k(x2-x1)+y1
If yTmp = 0 Then
    If drTmpYViivaAukot.Rows(0).Item("x") > drTmpYViivaAukot.Rows(0).Item("x2") Then
        yTmp = (k1 * (drTmpYViivaAukot.Rows(0).Item("x") - drTmpYViivaAukot.Rows(0).Item("x2")) + (drTmpYViivaAukot.Rows(0).Item("x") - iPx1)) + drTmpYViivaAukot.Rows(0).Item("y")
    Else : yTmp = (k1 * (drTmpYViivaAukot.Rows(0).Item("x2") - drTmpYViivaAukot.Rows(0).Item("x2")) + (drTmpYViivaAukot.Rows(0).Item("x2") - iPx1)) + drTmpYViivaAukot.Rows(0).Item("y2")
    End If
End If
If yTmp2 = 0 Then
    If drTmpYViivaAukot.Rows(1).Item("x") > drTmpYViivaAukot.Rows(1).Item("x2") Then
        yTmp2 = (k2 * (drTmpYViivaAukot.Rows(1).Item("x") - drTmpYViivaAukot.Rows(1).Item("x2")) + (drTmpYViivaAukot.Rows(1).Item("x") - iPx1)) + drTmpYViivaAukot.Rows(1).Item("y")
    Else : yTmp2 = (k2 * (drTmpYViivaAukot.Rows(1).Item("x2") - drTmpYViivaAukot.Rows(1).Item("x2")) + (drTmpYViivaAukot.Rows(1).Item("x2") - iPx1)) + drTmpYViivaAukot.Rows(1).Item("y2")
    End If
End If

```

KUVA 10. Viivan y-koordinaatin laskeminen kulmakertoimen ja aloituspisteen avulla

```

'laskeskellaan kahden viivan välinen korkeus
If yTmp > yTmp2 Then
    alueKorkeus = yTmp - yTmp2
Else : alueKorkeus = yTmp2 - yTmp
End If

```

KUVA 11. Korkeuden laskeminen

```

'PintaAla A= kanta*korkeus | korkeus = korkeuden itseisarvo
dPintaAla += Abs(alueKorkeus) * tarkkuus
drTmpYViiva.Rows.Clear()

```

KUVA 12. Lasketun alueen lisääminen kokonaispinta-alaan

## 4 JATKOKEHITYS

Pinta-ala piirtotyökalulle havaittiin jatkokehitystarve hyvin nopeasti perustoimintojen valmistumisen jälkeen. Tavoitteena oli saada piirtotyökalusta monipuolisempi ja helppokäyttöisempi. Tätä varten piti suunnitella lukuisia uusia ominaisuuksia. Tarpeena oli muun muassa saada määritellä viivan tyyppiä, valita viiva klikkaamalla sitä, piirtää ja muokata aukkoja monipuolisemmin, piirtää poikkeavia seinäalueita, joiden avulla tarjoukselle pystyttäisiin laskemaan tiilialueita eri tiilillä ja hinnoilla, sekä saada kuvasta paremmin sopiva piirtoalueelle kuvan suurentamisen ja pienentämisen muodossa.

### 4.1 Viivan valinta ja muokkaus

Ensimmäinen askel yksinkertaisen kuvaajan piirtämisen ja sen pinta-alan laskemisen jälkeen oli kehittää mahdollisuus valita piirrettävä viiva kuvasta klikkaamalla. Viivojen valinnan helppokäyttöisyys oli tärkeä ominaisuus jatkon kannalta. Tärkeää oli, että kuvan valintaan tulisi myös mahdollisuus muokata kuvaa siirtämällä jo piirrettyä viivaa sekä muuttaa jo piirretyn viivan tyyppiä. Itse viivan valintafunktion kehitin JavaScript-skriptauskielellä, joka suoritetaan käyttäjän verkkoselaimessa. JavaScriptin avulla pystytään helposti hallitsemaan käyttäjän antamat komennot suoraan selaimelta nopeasti ja joustavasti. Itse palvelimella suoritettava koodi ei pysty käsittelemään käyttäjän komentoja, kuten hiiren klikkausta tai napin painallusta joustavasti, vaan kaikki tiedot tulisi aina lähettää palvelimelle ja palvelin joutuisi vastaamaan takaisin. Tämä aiheuttaisi turhaa käytön aikaista viivettä, joten JavaScript kielen käyttäminen oli jo näiltä osin helppo valinta ominaisuuden luomiselle.

Kun käyttäjä klikkaa piirtoaluetta, joka on normaali kuva rajatun paneelin sisässä, kutsuu selainsivu JavaScriptin mousedown-funktiota ja suorittaa sen sisältämän koodin. Koodissa JavaScript laskee klikatun kohdan suhteessa paneelin reunaan, tallentaa sen piilotettuun ASP.NET-Hiddenfield-kontrolliin ja kutsuu piilotetun painikkeen ButtonClick (klikkaus) -funktiota, joka käynnistää palvelimella suoritettavan ASP.NET-funktion, jossa itse laskenta tapahtuu. Hiddenfieldejä on helppo käyttää muuttujien tallentamiseen, koska palvelimella suoritettava koodi pystyy suoraan käsittelemään ASP.NET-kontrolleja ja niiden sisältämiä arvoja, mutta arvot eivät silti näy suoraan käyttäjälle millään tavalla. ButtonClick-funktiossa haetaan piilotetuista Hiddenfield-kontrolleista

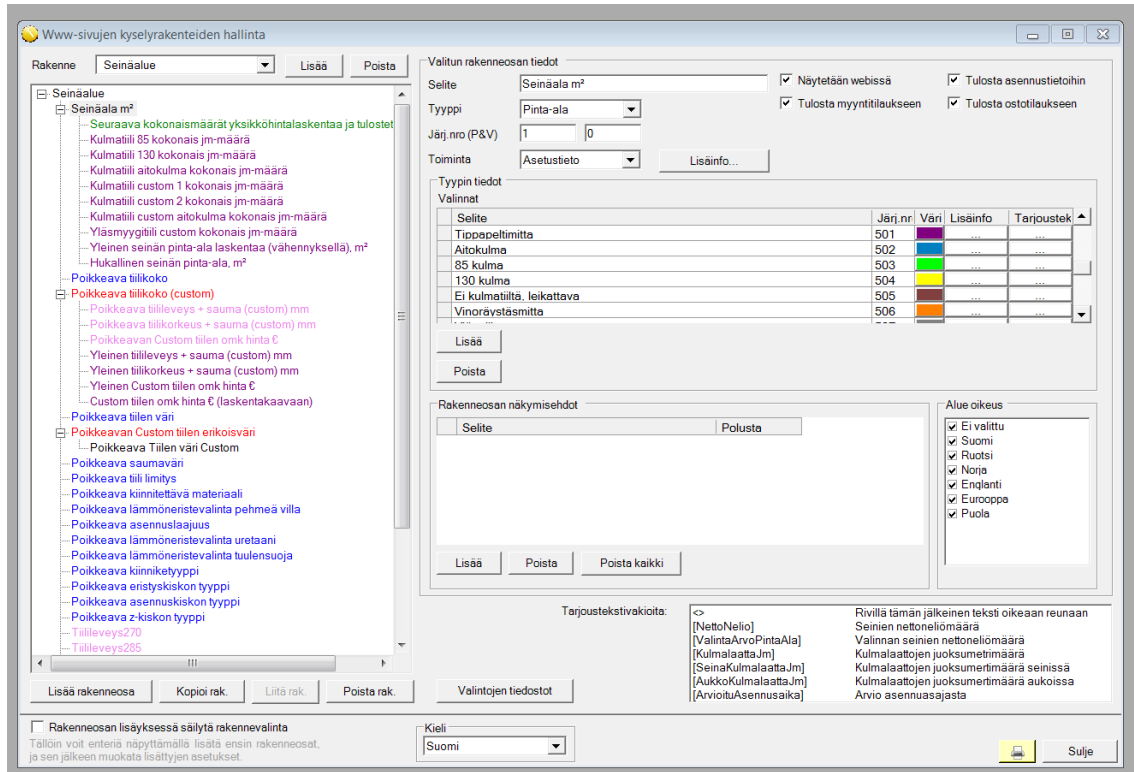


niihin tallennetut klikkauskohtien arvot, jotka tallennetaan toiseen muuttujaan. Tämän jälkeen klikattuja koordinaatteja vertaillaan jo piirrettyjen viivojen koordinaatteihin. Vaaka- ja pystyviivojen koordinaattien vertailu klikkauskohtien koordinaatteihin on hyvin helppoa ja suoraviivaista, tulee tarkistaa ehtolauseilla, sijaitseeko klikattu koordinaatti minkään viivan aloitus- ja lopetuspisteiden välissä määritetyn marginaalin turvin. Marginaaliksi päätyi testaamisen jälkeen 10 kuvapistettä, mikä tuntui sopivalta tarkkuudelta viivan paksuus huomioon ottaen. Mikäli marginaalia ei olisi ollut ollenkaan, olisi käyttäjä joutunut klikkaamaan todella tarkasti yhden kuvapisteen tarkkuudella oikein, mikä taas olisi hidastanut ja vaikeuttanut sujuvaa käyttöä huomattavasti. Vinojen viivojen kohdalla tuli ensin tarkistaa, onko viiva pystysuorassa klikatulla alueella, ja tämän jälkeen laskea, mikä olisi vinon viivan koordinaatti klikatun kohdan x-akselilla. Tämän jälkeen tätä tulosta verrataan klikattuihin koordinaatteihin, ja mikäli ne ovat asetetun marginaalin sisällä, merkataan viiva klikatuksi. Viivan valintaan kehitettiin myös mahdollisuus valita useampi viiva kerralla pitämällä Control-näppäintä pohjassa sekä mahdollisuus valita samantyyppiset viivat pitämällä ALT-näppäintä pohjassa.

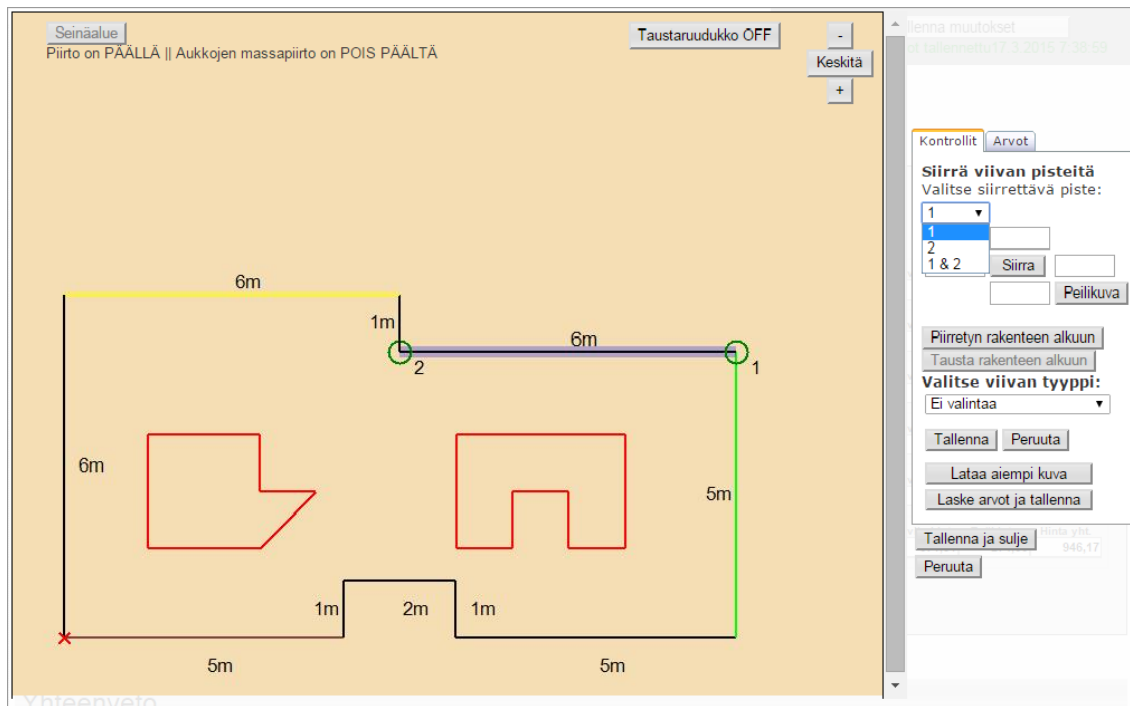
Viivan valinnan jälkeen oli luonnollista kehitellä jotain sitä hyödyntävää toiminallisuutta piirtotyökaluun. Viivan tyyppin muokkaus on tärkeä osa uutta piirtotyökalua. Sen avulla käyttäjä pystyy valitsemaan jälkikäteen piirretyn seinäalueen viivan ja valitsemaan sille erityisen tyyppin, joka vaikuttaa tarjouksen kokonaishintaan ja laskentaan. Viivojen väri-indeksi pystytään määrittämään Totalin Windows WebForms -ohjelmassa, jossa hallitaan tarjouksen syötön asetuksia (kuva 12). Viiva, jonka tyyppi on jokin muu kuin oletus, piirtyy Windows-ohjelmassa määritetyllä värillä piirtoalueelle.

Klikkaamalla piirtoalueella jonkin tyyppin omaavaa viivaa tulee pudotusvalikkoon näkyviin klikatun viivan tyyppi. Pudotusvalikko on myös piirrettäessä koko ajan näkyvillä. Tällöin sen valintaa muuttamalla uusi piirretty viiva piirtyy oletuksena valitun tyyppiseksi ja se näkyy heti valitun tyyppin värisenä. (Kuva 13.) Viivan tyyppi tallentuu tietokannassa jokaisen piirretyn viivan kohdalle omana arvokenttänään ja on oletuksena nolla. Kun viivatyyppiä muutetaan, tallentuvat viivat uudestaan kantaan. Viivan väri haetaan saman taulun eri tietueesta, johon viitataan tallennetulla yksilöintitietueella eli ID:llä. Tähän tietueeseen viivan väri on tallennettu ARGB-muodossa. ARGB-muoto sisältää väritiedot heksamuodossa, ja se tulee englanninkielen sanoista alpha, red, green, ja blue. ARGB siis sisältää tiedon, paljonko väri sisältää kutakin pääväriä (vihreää, punaista ja sinistä), sekä värin läpinäkyvyysarvon. Viivatyyppit ja niiden pituudet ja värit tallentuvat tietokantaan,

josta ne vaikuttavat tarjouksen laskentaan, tarjouslaskennassa muodostuvaan hintaan sekä raaka-ainelaskelmiin.



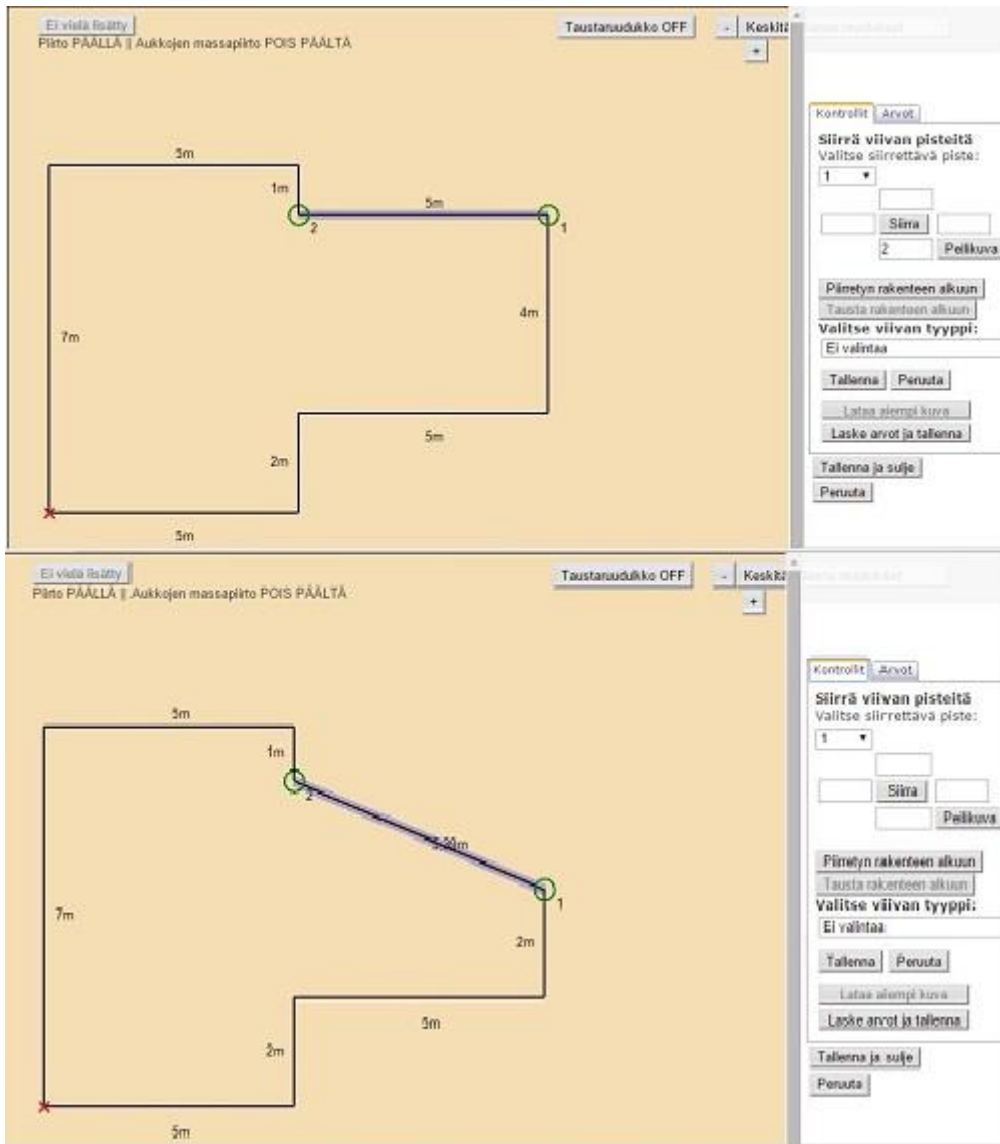
KUVA 12. Seinän asetukset



KUVA 13. Viivojen värit ja viiva valittuna, viivalla ei ole tyyppiä valittuna

## 4.2 Kuvan muokkaus

Kuvan muokkausmahdollisuus tehtiin nopeasti viivan tyyppin muokkaamisen jälkeen. Samoin kuin viivan tyyppin valinta, hyödyntää kuvan muokkauskin kuvan klikkausominaisuutta. Kuvaa klikkaamalla käyttäjä valitsee siirrettävän viivan. Pudotusvalikosta käyttäjä voi valita, haluaako hän siirtää jompaakumpaa viivan päätepistettä vaiko molempia yhtä aikaa, jolloin koko viiva liikkuu haluttuun suuntaan. Siirtotyökalu siirtää viivaa ja siihen liitetyjä muita viivoja valitun suunnan mukaisesti niin, että koko yhtenäinen rakenne säilyy. Näin käyttäjä voi halutessaan esimerkiksi kasvattaa jo piirretyn seinän leveyttä ja korkeutta. Kuvassa 14 on demonstroitu viivan muokkauksen toimintaa. Ylemmässä kuvassa eräs kuvaajan viivoista on valittuna ja oikeasta yläkulmasta on valittuna se viivan piste, jota halutaan siirtää. Tämän jälkeen tekstikenttään on syötetty arvo 2, joka vastaa ohjelmassa 2:ta metriä. Tämän jälkeen ohjelma siirtää valittua viivaa 2 metriä alaspäin, jolloin pystysuoran viivan pituudeksi tulee 2 metriä ja vinon viivan pituudeksi 5,50 metriä. Siirrä-nappia painamalla funktiossa etsitään valittu viiva datatable-tyyppisestä taulukkomuuttujasta, mihin piirretyt viivat on tallennettu. Valittua viivaa sitten siirretään käyttäjän valitsemaan suuntaan muuttamalla viivan aloitus- tai lopetuskoordinaatteja syötteen mukaisesti. Tämän jälkeen samasta datataulukosta etsitään valittuun viivaan yhteydessä olleet viivat eli viivat, joilla on ollut sama aloitus- tai lopetuskoordinaatit kuin valitulla viivalla. Sen jälkeen viivoja venytetään ja muokataan niitä samassa suhteessa niin, että ne yhdistyvät siirrettävään viivaan ja rakenne säilyy yhtenäisenä

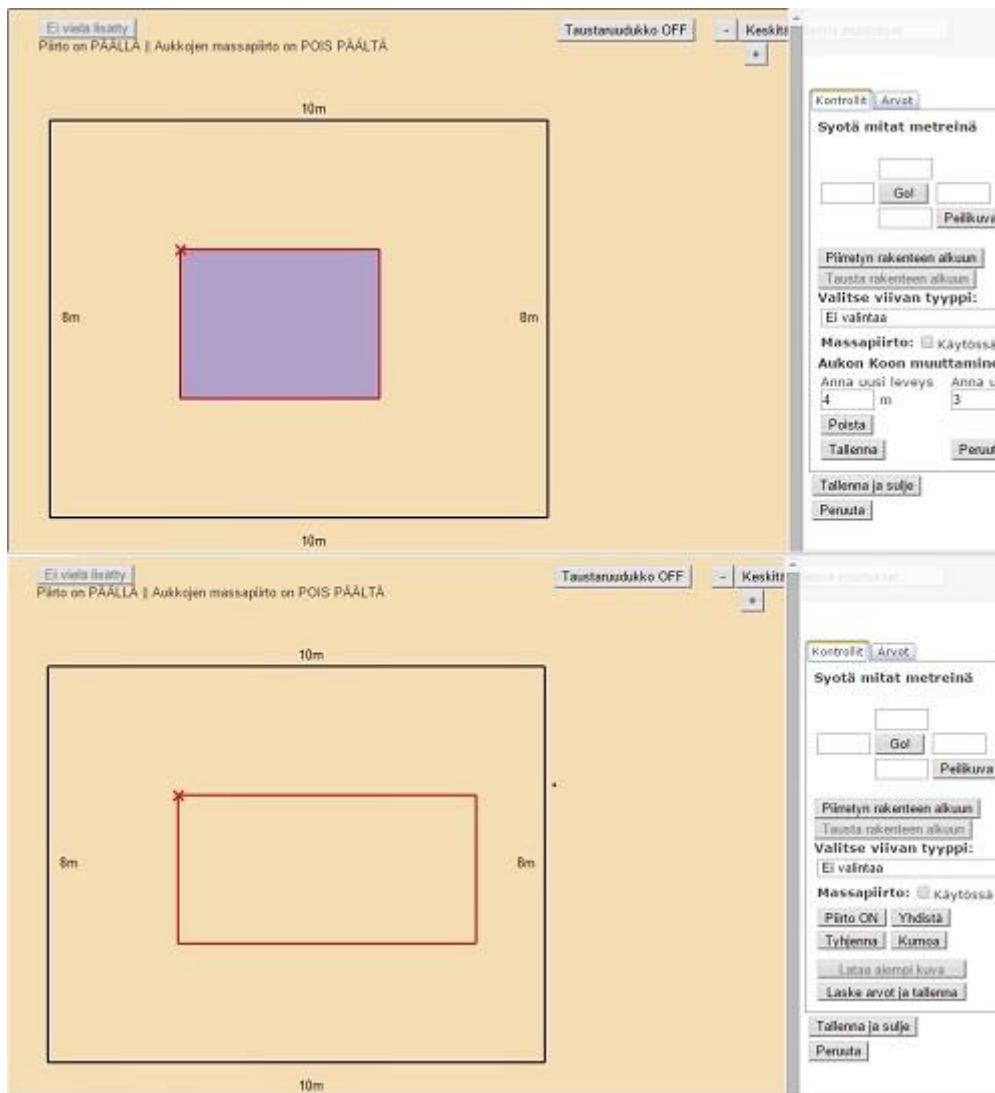


KUVA 14. Ennen ja jälkeen viivan muokkauksen

### 4.3 Aukot ja poikkeavat seinäalat

Piirtotyökalun alusta lähtien on kuvaan ollut mahdollista piirtää aukkotyyppisiä rakenteita piirrettävän alueen sisälle. Näiden pinta-alat vähentyvät kokonaispinta-alasta ja ne korostetaan punaisella värillä rakenteen sisällä. Viivan klikkauksen yhteydessä oli myös tarve kehittää ominaisuus, jonka avulla pystyisi valitsemaan ja muokkaamaan aukkoja. Tämä toteutettiin aiemmin jo mainitun viivan tunnistuksen yhteyteen niin, että mikäli funktio ei löytänyt, että klikkauskohta osuisi minkään viivan kohdalle, alkaa funktio käymään läpi datataulua, joka sisältää tiedot jokaisesta aukosta. Tauluun on aukkojen piirto vaiheessa tallennettu aukon pienin ja suurin x ja y koordinaatti sekä sen pinta-ala. Klikatun pisteen koordinaatin vertailu tapahtuu aukon minimejä ja maksimeita kohden. Mikäli klikattu koordinaatti osuu näiden väliin, on käyttäjä klikannut aukon

sisältä ja aukko merkitään valituksi ja korostetaan samalla sinisellä värillä kuin viivaa valitessakin. Kun oikea aukko on löydetty, avautuu käyttäjälle valikot, joiden avulla käyttäjällä on mahdollisuus muokata aukon kokoa antamalla tekstikenttiin uusi leveys ja korkeus. Oletuksena näihin tekstikenttiin sijoitetaan aukon mitat. Valitsemalla Tallenna käyttäjän valitsemit mitat astuvat voimaan ja valittu aukko muokkautuu syötettyjen arvojen mukaisesti. (Kuva 15.)



Kuva 15. Aukko ennen jälkeen muokkauksen

Poikkeavat seinäalueet kehiteltiin asiakkaan tarpeesta laskea samalle seinälle erillinen seinäalue eri laskenta-asetuksilla. Poikkeavat seinäalueet piirretään omana seinänään kuten normaalit seinätkin. Poikkeavaa on vain, että poikkeavan seinäalueen pinta-ala vähennetään pääseinän alasta. Poikkeava seinäalue näkyy päärakenteelta katsottaessa harmaana alueena kuvan taustalla ja se ei osallistu mitenkään pääseinän laskentaan muuten kuin vähentämällä sen sisältämä pinta-

ala pääseinän pinta-alasta tallennusvaiheessa. Laskennallisesti ja toiminnaltaan poikkeava seinäalue ei poikkea normaalista seinäalasta muuten kuin niin, ettei se voi sisältää aukkoja. Poikkeava seinäalue on jo itsessään aukko pääseinä alan kannalta. Niinpä laskennan helpottamiseksi tehtiin päätös, ettei poikkeaviin seinäaloihin ole tarvetta saada aukkoja. Kuten seinän päärakenteellekin, poikkeavan seinäalueen viivojen sisältämää tyyppiä pystyy muuttamaan normaalisti ja poikkeavan seinäalueen piirtäminen ja muokkaaminen tapahtuu kuin normaalia seinää piirtäessä.

## 5 POHDINTA

Opinnäytetyön aiheena oli piirtotyökalun jatkokehittäminen asiakkaan tarpeen mukaan. Työn tavoitteena oli jatkokehittää piirtotyökalua niin, että se kattaisi mahdollisimman hyvin kaikki eri tilanteet, joita asiakasyrityksellä tulee vastaan tarjousta syötettäessä. Työssä tehtyjen ominaisuuksien avulla tarpeet hyvin pitkälti täytettiin, vaikkakin jatkokehittämistarpeita tulee edelleen vastaan.

Koko piirtotyökalun luomiskaaren aikana opin, kuinka monipuolisia sivustoja ja toimintoja yhteen sivuun voi sisällyttää. ASP.NET-sivu taipuu yllättävän hyvin toimintoihin, joihin sitä ei suoranaisesti ole suunniteltu. Yhdistettynä JavaScript- ja jQuery-skriptauskomponentteihin mahdollisuudet ovat hyvin monipuoliset ja verkkosivustosta on helppo tehdä dynaaminen ja toimiva.

Aloitin opinnäytetyön kirjoittamisen ja ominaisuuksien kehittämisen keväällä 2015 ja opinnäytetyön kirjoittaminen jatkuikin aktiivisena kevään läpi. Kevään jälkeen kesälomien alkaessa opinnäytetyön kirjoittaminen jäi tauolle. Opinnäytetyö valmistui 2016 alkukevästä ja tähän mennessä piirtotyökalua on jouduttu jonkin verran vielä jatkokehittämään. Uusia ominaisuuksia ovat esimerkiksi rakenteen kehittyneempi ylä- ja alajuoksujen eli rakenteen ylä- ja alaviivan pituuden laskeminen, joka ottaa huomioon paremmin eri käyttötilanteet, sekä piirtoalueen tallentaminen tietokoneelle pdf-muodossa. Myös piirtotyökalua ja sen laskentaa on jouduttu useaan kertaan optimoimaan toiminnan hitauden vuoksi joissain tapauksissa. Esimerkiksi viivan valinta alkoi suuremmissa piirroksissa kestämään hyvinkin kauan ja sitä onnistuttiinkin nopeuttamaan moninkertaiseksi. Piirtotyökalu on tällä hetkellä aktiivisessa käytössä asiakasyrityksessä ja on hyvin mahdollista, että sitä tullaan vielä jatkokehittämään jossain määrin.

## LÄHTEET

- 1 .NET Framework. 2015. Wikipedia. Saatavissa: [http://en.wikipedia.org/wiki/.NET\\_Framework](http://en.wikipedia.org/wiki/.NET_Framework)  
Hakupäivä 17.3.2015.
2. Common Language Runtime. 2015. Wikipedia. Saatavissa: [http://en.wikipedia.org/wiki/Common\\_Language\\_Runtime](http://en.wikipedia.org/wiki/Common_Language_Runtime) .Hakupäivä 17.3.2015.
3. The .NET Framework stack. 2015. Wikipedia. Saatavissa: <https://upload.wikimedia.org/wikipedia/commons/thumb/d/d3/DotNet.svg/800px-DotNet.svg.png>  
Hakupäivä 25.2.2015.
4. Chappel D. 2002 . Understanding .NET. e-kirja. Saatavissa: [http://www.google.fi/books?hl=fi&lr=&id=gTSgBSn6vvoC&oi=fnd&pg=PR11&dq=chappel+D+Understanding+.NET&ots=f3x-WgkKQI&sig=Mn0M-evtKTSxiMd94QnQ0cT8HY4&redir\\_esc=y#v=onepage&q&f=false](http://www.google.fi/books?hl=fi&lr=&id=gTSgBSn6vvoC&oi=fnd&pg=PR11&dq=chappel+D+Understanding+.NET&ots=f3x-WgkKQI&sig=Mn0M-evtKTSxiMd94QnQ0cT8HY4&redir_esc=y#v=onepage&q&f=false) Hakupäivä 21.3.2015.
5. ASP.NET. 2015. Wikipedia. Saatavissa: <http://en.wikipedia.org/wiki/ASP.NET>  
Hakupäivä 21.3.2015.
6. Introducing ASP.NET Webpages. 2015. Microsoft. Saatavissa: <http://www.asp.net/web-pages/overview/getting-started/introducing-aspnet-web-pages-2/getting-started>  
Hakupäivä 17.3.2015.
7. ASP.NET Webpages intro. 2015. W3Schools. Saatavissa: [http://www.w3schools.com/aspnet/webpages\\_intro.asp](http://www.w3schools.com/aspnet/webpages_intro.asp) Hakupäivä 17.3.2015.
8. ASP.NET 4.5 new features. 2015. Techbubbles.com. Saatavissa: [http://www.techbubbles.com/wp-content/uploads/2013/03/image\\_thumb20.png](http://www.techbubbles.com/wp-content/uploads/2013/03/image_thumb20.png)  
Hakupäivä 2.1.2015.
9. ASP.NET Overview. 2015. Microsoft MSDN. Saatavissa: <https://msdn.microsoft.com/en-us/library/4w3ex9c2%28v=vs.140%29.aspx> Hakupäivä 17.3.2015.



10. MVC Intro. 2015. W3Schools. Saatavissa: [http://www.w3schools.com/aspnet/mvc\\_intro.asp](http://www.w3schools.com/aspnet/mvc_intro.asp)  
Hakupäivä 17.3.2015.

11. Introduction to ASP.NET MVC. 2015. Microsoft VirtualAcademy. Saatavissa:  
<http://www.microsoftvirtualacademy.com/training-courses/introduction-to-asp-net-mvc>  
Hakupäivä 17.3.2015.

12. MVC-model. 2015. W3Schools. Saatavissa: <http://www.w3schools.com/aspnet/27.jpg>  
Hakupäivä 2.3.2015.

13. Introduction to ASP.NET Web Forms. 2015. Microsoft. Saatavissa: <http://www.asp.net/web-forms/what-is-web-forms> Hakupäivä 17.3.2015.

14. Introduction to ASP.NET and WebForms. 2015. Microsoft MSDN. Saatavissa:  
<https://msdn.microsoft.com/en-us/library/ms973868.aspx> Hakupäivä 17.3.2015.