

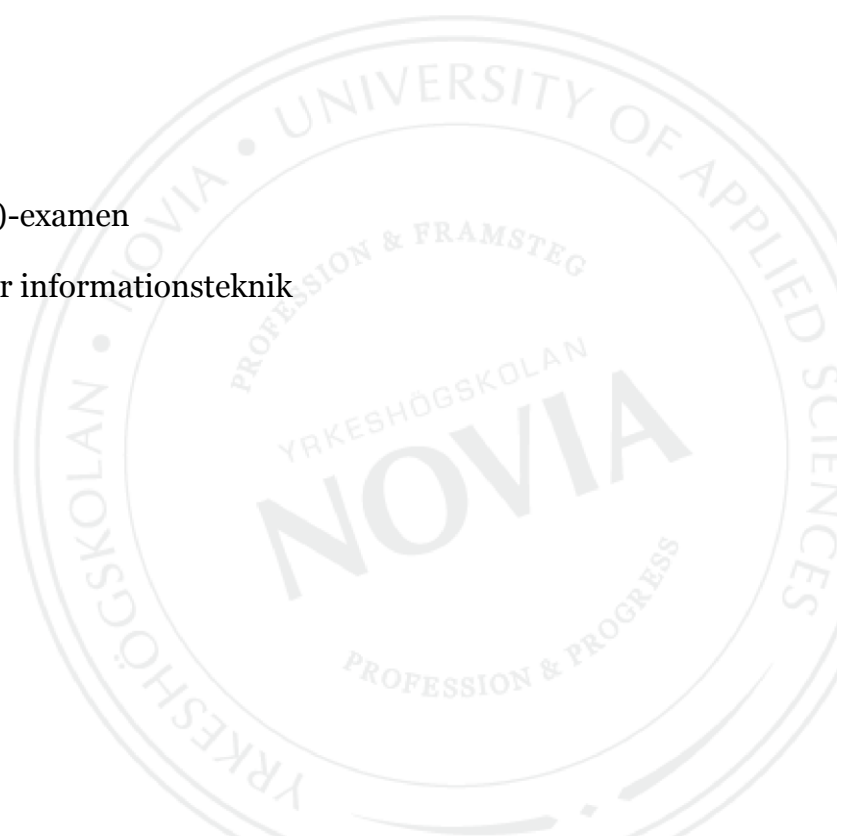
# Webbapplikation för registrering av försäljningsstatistik

Richard Sampert

Examens för ingenjör(YH)-examen

Utbildningsprogrammet för informationsteknik

Vasa 2016



## EXAMENSARBETE

Författare: Richard Sampert  
Utbildningsprogram och ort: Informationsteknik, Vasa  
Handledare: Susanne Österholm

Titel: *Webbapplikation för registrering av försäljningsstatistik*

---

Datum 16.04.2018

Sidantal 20

---

### Abstrakt

Detta examensarbete har utförts åt företaget Anvia i Vasa. Företaget är nu en del av Elisa. Målet med arbetet var att skapa ett nytt sätt att spåra vilka produkter eller tjänster som säljs. Efter en diskussion med användarna beslöts att en webbapplikation skulle skapas för att hantera vilka tjänster som såldes.

Arbetet utfördes genom att skapa en webbapplikation på Anvias interna portal Triton, genom att använda webbt teknikerna HTML, PHP och CSS.

Resultatet är en webbapplikation som skapades som ett tillägg till den interna portalen Triton.

---

Språk: svenska

Nyckelord: webbapplikation, PHP

---

## BACHELOR´S THESIS

Author: Richard Sampert  
Degree programme: Information Technology, Vasa  
Supervisor: Susanne Österholm

Title: Web Application for Registration of Sales Statistics

---

Date April 16, 2018                      20 pages

---

### Abstract

This thesis work has been carried out for the company Anvia in Vasa, which is now part of Elisa. The goal of this work was to create a new way of tracking which products or services are being sold. After a discussion with the users had been made, a web application had to be created to manage the services that were sold.

The work was done by creating a web application on Anvia's internal portal Triton, using the Web techniques HTML, PHP and CSS.

The result of this Bachelor's thesis is a web application that was created as an add-on to the internal portal Triton.

---

Language: Swedish

Key words: web application, PHP

---

## OPINNÄYTETYÖ

Tekijä: Richard Sampert  
Koulutusohjelma ja paikkakunta: Tietotekniikka, Vaasa  
Ohjaaja: Susanne Österholm

Nimike: *Verkkosovellus myyntitilastojen rekisteröintiä varten*

---

Päivämäärä 16.04.2018

Sivumäärä 20

---

### Tiivistelmä

Tämä opinnäytetyö on tehty yhtiölle Anvia Vaasassa. Anvia on nykyään osa Elisaa. Työn tavoitteena oli luoda uusi tapa seurata tuotteiden tai palveluiden myyntiä. Käyttäjien kanssa tehdyn keskustelun jälkeen, verkkosovellus perustettiin hallinnoimaan palveluita.

Työ tehtiin luomalla verkkosovellus Anvian sisäisellä portaalilla Tritonilla käyttäen HTML-, PHP- ja CSS-kieliä.

Opinnäytetyön tulos on toimiva verkkosovellus, joka on luotu laajenuksena Anvian sisäiseen Triton-portaaliin.

---

Kieli: ruotsi

Avainsanat: verkkosovellus, PHP

---

## Innehållsförteckning

1	Inledning .....	1
1.1	Uppdragsgivare .....	1
1.2	Bakgrund.....	2
1.3	Uppdrag .....	2
2	Tekniker .....	3
2.1	Webb- och skriptspråk .....	3
2.1.1	HTML.....	3
2.1.2	CSS .....	6
2.1.3	PHP.....	9
3	Utförande.....	12
3.1	Planering .....	12
3.2	Webbsidan "Seuranta" .....	12
3.2.1	Installation av Webbsidan "Seuranta" .....	13
3.2.2	Enkäten .....	14
3.2.3	Spara uppgifterna.....	16
4	Resultat.....	18
5	Diskussion .....	18
5.1	Vidareutveckling .....	19
5	Källförteckning .....	20

## Ordförklaringar

PHP	Rekursiv akronym för PHP: Hypertext Preprocessor är ett allmänt använt Scriptspråk med öppen källkod som är särskilt lämpat för webbutveckling och kan integreras i HTML-dokument.
CSS	Cascading Style Sheets, är stilmallar som tillåter både den som skapar dokument och den som läser dem att använda sin egen stilmall.
HTML	Hyper Text Markup Language, det huvudsakliga sidbeskrivningsspråket för webbsidor samt annan information som kan visas i en webbläsare.
SOAP	Ett protokoll för utbyte av information i decentraliserade och distribuerade miljöer.
REST	Representational State Transfer är ett IT-arkitekturbegrepp som beskriver ett möjlig kommunikationssätt mellan maskiner
HTTP	Hyper Text Transfer Protocol, används för att överföra information och är den underliggande protokollet som används av World Wide Web och det här protokollet definierar hur meddelanden formateras och överförs och vilka åtgärder som webbservrar och webbläsare ska ta som svar på olika kommandon.
URL	Uniform Resource Locator, är den teckensträng som identifierar en viss resurs på nätet, t.ex. en webbsida
Tagg	Tagg i detta examensarbete syftar på engelska ordet tag.
JSON	JavaScript Objekt Notation, är ett lättformat som används för datautbyte och är ett sätt att serialisera på ett sådant sätt att det blir JavaScript-kod.
Tigers	Tigers är ett program skapat av Anvia för att hålla koll på tillgängligheten av bredband till olika adresser.

Microsoft Dynamics NAV Microsoft Dynamics NAV är ett enterprise resource planning (ERP) app från Microsoft.

Triton Triton är en webbapplikation som utvecklats inom Anvia för att kunna upprätthålla kundernas tjänster.

# 1 Inledning

Uppdraget gick ut på att skapa ett nytt sätt att lagra data om försäljningen av produkter eller tjänster. Försäljningsprogrammet som nu används håller bara koll på hur mycket av en tjänst eller produkt som säljs totalt. Webbapplikationen skulle göra det enklare för anställda att kunna lägga in vilken tjänst eller produkt som säljs som förmannen skulle kunna hålla koll på via en Excel-fil som finns på nätverksenheten. Webbapplikationen skulle vara enkel att använda och flera användare skulle kunna mata in data samtidigt.

## 1.1 Uppdragsgivare

Anvia koncernen bestod av Anvia Telecom och dess dotterbolag. Anvia ingår sedan 2017 i Elisa Abp. Anvia Telecom som tidigare gick under namnet Vasa Läns Telefon Ab förnyade sin målsättning år 2007 och koncernen bytte namn år 2008 från Vasa Läns Telefon Ab till Anvia Telecom. Målsättningen som förnyades år 2007 var för att Anvia ville sträva till en ekonomisk tillväxt samt skapa en bättre kundrelation med sina kunder. Anvia var den dominerade telefonoperatören i Västra Finlands Län med 110 000 konsumentkunder och 15 000 företagskunder. Koncernen bestod av tre olika affärsområden: Anvia ICT och Anvia TV som är nu en del av Elisa och Anvia Securi som nu är en del av Viria Oyj.

Anvia Securi (Viria Securi) producerade lås- och säkerhetstjänster till företag och konsumenter. Affärsverksamheten gick mestadels ut på att leverera passersystem och larmsystem.

Anvia ICT affärsverksamheten erbjöd omfattande tjänster inom datakommunikation och IT till såväl konsumenter som företag. De viktigaste tjänsterna för konsumenterna var internet- och tv-tjänster och telefoni. Företagstjänsten hade nästan samma utbud som man hade till konsumenterna men man erbjöd även support och skolning till produkterna och tjänsterna.

Anvia TV affärsområde var utveckling, försäljning och leverans för mottagning av rörlig bild och ljud till allmänheten. Anvia TV utgörs av Anvia TV Oy och Hibox Systems Ab.

Anvia hade ca 750 anställda och dess omsättning år 2015 var 130,7 miljoner euro. Anvia ville förmedla en bra kundupplevelse samt att ständigt utveckla sina tjänster. Visionen var



att vara nationellt känd och prestigefull leverantör av lösningar och tjänster baserade på datanät. (Theseus, 2018).

## **1.2 Bakgrund**

Anvia tog i bruk ett nytt kassasystem Microsoft Dynamics Nav hösten 2015 som skulle ersätta det urgamla försäljningsprogrammet TTM som hade varit i användning redan under Vasa Läns Telefons tid. Då ändringen skedde så försvann även Anvias sätt att följa med vilken försäljare som säljer vilken produkt eller tjänst. Med Microsoft Dynamics Nav så syns all försäljning som en klumpsumma, så det är svårt att veta om det blir sålt via butiken eller via telefonen. Webbapplikationen skulle hjälpa Anvia att spåra vad som säljs och för att i fortsättningen ha möjligheten att hålla någon sorts tävling åt anställda och försöka motivera dem att jobba mer effektivt för att få bättre resultat åt firman.

## **1.3 Uppdrag**

Uppdraget gick ut på att hitta ett enkelt sätt att följa upp försäljningen av olika tjänster och produkter. Det var viktigt att flera skulle kunna lägga till uppgifter samtidigt, det skulle vara lätt att använda och att komma åt. Uppföljningstjänsten skulle tas i bruk och personalen skulle introduceras i hur den används.

Som krav ställdes att systemet skulle vara enkelt att komma åt, enkelt att använda och att flera ska ha möjlighet att använda den samtidigt. Ett annat krav var att man skulle kunna ta ut rapporter månadsvis från programmet.

## 2 Tekniker

Det här kapitlet beskriver de tekniker som användes i projektet. Man bestämde att applikationen skulle förverkligas som en webbenkät som skulle fyllas i via en webbläsare, informationen i enkäten skulle lagras i en Excel-fil på förmannens nätverksenhet. Man valde att enkäten skulle skapas med hjälp av HTML, PHP och CSS.

### 2.1 Webb- och skriptspråk

I underkapitlen presenteras de webb- och skriptspråken som har blivit använda i examensarbetet.

#### 2.1.1 HTML

HTML är en förkortning av Hyper Text Markup Language. HTML är inget programmeringsspråk utan det är ett märkspråk. Märkspråket HTML är en samling av HTML-element för att beskriva innehållet på webbsidor som webbläsare kan tolka. Syftet med en webbläsare som t.ex. Mozilla Firefox och Google Chrome är att läsa HTML-dokumentet och visa dem som webbsidor. Webbläsaren visar inte HTML-elementen utan använder HTML-elementen för att tolka innehållet på webbsidan. (Refsnes, Refsnes, & Refsnes, 2010, s. 1–2).

År 1989 föreslog Tim Berners-Lee som arbetade vid Europeiska partikelfysikinstitutet CERN (Centre European pour la Recherche Nucleaire) ett system för att tillåta forskare att dela dokument med andra med hjälp av elektroniska nätverksmetoder. Hans idé blev det som kallas World Wide Web. År 1992 släppte Berners-Lee och CERN gruppen första versionen av HTML 1.0 som slutfördes 1993. Denna specifikation var så enkel att den kan skrivas ut på ena sidan av ett papper, men även då innehöll den den grundläggande idén som har blivit en viktig del i utvecklingen av HTML. År 1994 utvecklades HTML 2.0 av Internet Engineering Task Forces' HTML Working Group . Denna grupp upplöstes senare till förmån för World Wide Web Consortium (W3C) som fortsatte att utveckla HTML. World Wide Web Consortium som hade tagit över HTML-utvecklingen försökte skapa någon standardisering i HTML 3.0 men det var så mycket argument över vad som bör inkluderas att det aldrig kom längre än diskussionfasen. År 1996 utfärdades en konsensusversion, HTML 3.2. medtillagda funktioner som tabeller och text som flyter runt bilder till den officiella specifikationen samtidigt som den bibehöll bakåtkompatibilitet med HTML 2.0. W3C släppte HTML 4.0 specifikationen i slutet av 1997

och därefter HTML 4.01 1999 som för det mesta skulle korrigera några fel i 4.0 specifikationen. År 2014 blev HTML 5 släppt som en stabil W3C rekommendation som betyder att specificeringsprocessen är klar.

HTML-dokument är byggt upp med hjälp av HTML-element som består av starttagg och sluttagg som bildar HTML-element. Mellan start- och sluttaggen finns elementets innehåll. HTML-element kan nästlas så att ett HTML-element kan vara innehåll i ett annat HTML-element. Det element som alltid har flest HTML-element i sig är HTML-taggen som definierar att det är ett HTML-dokument. Det finns element som kommer att sakna innehåll och dessa tomma element ska bara ha starttagg, ett exempel är ett linjebyte som skrivs som `<br>`. (Refsnes, Refsnes, & Refsnes, 2010, s. 12–14).

Man kan ge HTML-elementen ytterligare information genom att använda attribut. Ett attribut används för att definiera egenskaperna hos ett HTML-element och placeras inuti elementets starttagg, t.ex. `class="name"`. De attributen som de flesta HTML-elementen stöder är:

- Id, anger ett unikt id för ett HTML-element, id-värdet kan användas av CSS och Javascript för att utföra vissa uppgifter för ett unikt HTML-element med det angivna id-värdet.
- Class är till för att namnge flera HTML-element på hemsidan, vilket passar bra om du har flera områden på hemsidan som ska tilldelas samma stiler.
- Style, används för att ange den grafiska utformningen av ett HTML-element som färg, teckensnitt och storlek.

Då man anger värdet på ett attribut ska man använda små bokstäver både HTML4 samt HTML5 stöder användningen av stora bokstäver, men det är inte rekommenderat. (Refsnes, Refsnes, & Refsnes, 2010, s. 16–17).

För att skicka information från elementen till en server så använder man sig av HTML-formulär, t.ex. från ett textfält. Den typen av formulär-element som används mest är input-elementet. Input-elementets typ bestäms med hjälp av attributet Type, de vanligaste input-typerna är:

- Text, skapar ett område som användaren kan skriva text i.
- Checkbox, skapar valmöjligheter som kan markeras, checkbox används när du vill att användaren skall välja ett eller flera alternativ. Man kan välja flera alternativ åt

gången, ifall bara ett alternativ ska väljas åt gången borde man istället använda Radiobox.

- Radiobox, skapar valmöjligheter som kan markeras. Används när man vill att användaren skall välja endast ett alternativ.

För att skicka informationen i ett formulär till servern för behandling används typen submit på elementet input. I form-elementet så används oftast metoden post eller metoden get.

- Metoden GET visar innehållet i formuläret som en URL (webbadress) i webbläsarens adressfält.
- Metoden POST sänder datat i head-informationen dvs. skickar innehållet utan att det visas för användaren.

I elementet form kan man ange attributet action tillsammans med adressen till en webbsida. Action anger till vilken sida på servern formulärets data sänds. Attributet action anger till vilken sida på servern som formulärets datas sänds. (Refsnes, Refsnes, & Refsnes, 2010, s. 98–105).

*Kodexempel 1. Ett exempel på ett HTML-dokument.*

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Text Input</h2>
    <form>
      First name:<br>
      <input type="text" name="firstname">
    <br>
      Last name:<br>
      <input type="text" name="lastname">
    </form>
  </body>
</html>
```

# Text Input

First name:

Last name:

Figur 2. Hur en webbläsare tolkar HTML-dokumentet i kodexempel 1.

## 2.1.2 CSS

Cascading Style Sheets som oftast beskrivs med förkortningen CSS är ett kodningsspråk för att man ska kunna definiera utseendet och kontrollera layouten på webbsidor. Genom att använda CSS så kan man t.ex. skapa skuggor, kontrollera utseendet på kanter eller var bilder placeras. Man kan jämföra CSS med Microsoft Word där man kan skapa stil-regler för sitt innehåll. CSS är inte alternativ till HTML, de är två olika kodningsspråk som har helt olika uppgifter, CSS är en stilmall för att man skall kunna ange hur ett HTML-dokument ska visas. (McFarland, 2009, s. 1).

Det finns tre olika versioner av CSS. CSS-nivå 1 i W3C-rekommendationen skapades i december 1996. Utgåvan av CSS 1 stöder: teckensnittegenskaper, textattribut, anpassning av text, tabeller, bilder och mer, färger på text och bakgrunder, avstånd mellan ord, bokstäver och linjer, marginaler, gränser och positionering, och unik identifiering och klassificering av grupper av attribut. Det var efter att denna utgåva som arbetsgruppen Cascading Style Sheets och Formatting Properties bildades av W3C för att fokusera enbart på CSS.

I maj 1998 släpptes CSS 2 som lade till nya möjligheter som fast positionering. Det var inte förrän i juni 2011 som W3C släppte CSS 2.1, vilket fixade fel och anpassade möjligheterna bättre med webbläsarfunktioner.

CSS 3 som släpptes 2012 är den senaste utvecklingen av Cascading Style Sheets-språket och syftar till att utöka CSS 2.1. CSS 3 ger många efterlängtade nyheter, som avrundade hörn,

skuggor, gradienter, övergångar eller animationer, samt nya layouts som flera kolumner, flexibla rutor eller rutnätslayouts.

Det finns tre olika sätt att använda sig utav CSS:

- Inline CSS, man definierar CSS i ett enskilt HTML-element med attributet style.
- Internal, CSS koden skall vara inbäddad i HTML-dokumentets början innanför HTML-elementet <head>.
- External, CSS koden finns definierat i en skild fil med filändelsen .css.

External betraktas som det bästa sättet att använda sig utav CSS p.g.a. man kan ändra utseendet på en hel webbsida genom att ändra CSS koden i en enda fil istället för att gå igenom varje fil som webbsidan använder sig utav av. (McFarland, 2009, s. 34–36).

Varje CSS-stil har två grundläggande delar den första delen pekar på vad som skall formateras och den andra innehåller formateringsegenskaperna t.ex. första delen pekar på HTML-elementet <h1> och andra delen anger vilka egenskaper HTML-elementet skall ha t.ex. om man vill ändra textstorleken. Det finns tre olika sätt att ange regler till vilket HTML-element som skall formateras. Man kan välja att formatera alla HTML-element av samma typ, formatera ett HTML-element baserat på värdet på dess id-värde eller så kan man välja alla HTML-element som har samma angivna class. För att formatera ett unikt HTML-element så använder man sig utav attributet id om man vill formatera flera HTML-element använder man sig utav attributet class. (McFarland, 2009, s. 49–53).

Kodexempel 2. Ett exempel på hur CSS kan användas för att ändra på formatering av element.

```

<!DOCTYPE html>
  <html>
  <head>
    <style>
      p {
        font-family: Georgia, serif;
        font-size: 30pt;
        text-decoration: underline;
      }

      #p1 {
        font-family: cursive;
        font-size: 20pt;
        text-decoration: line-through;
      }

      p.p2 {
        font-family: serif;
        font-size: 25pt;
        text-decoration: underline overline;
      }
    </style>
  </head>
  <body>

    <p>Blir formaterad av p.</p>
    <p id="p1">Elementet är unikt då den definieras med id.</p>
    <p class="p2">väljer elementet med ett specifikt klassattribut.</p>

  </body>
</html>

```

Blir formaterad av p.

~~Elementet är unikt då den definieras med id.~~

väljer elementet med ett specifikt klassattribut.

Figur 3. Hur webbläsaren visar HTML-dokumentet i kodexempel 2.

### 2.1.3 PHP

PHP (rekursiv akronym för PHP: Hypertext Preprocessor) är ett allmänt använt öppet källkodsspecifikt skriptspråk som är särskilt lämpat för webbutveckling och kan integreras i HTML-dokument. Detta skriptspråk är serverbaserat, vilket betyder att till skillnad från t.ex. JavaScript så körs PHP-koden på webbservern i stället för att utföras på klienten. Man kan enkelt lägga in PHP-kod i ett HTML-dokument genom att använda sig av PHP:ns start- och sluttagg, `<?php` och `?>`. Zend Engine, en programtolk, är centrum för PHP och är den komponenten som analyserar och kör PHP-koden. (What is PHP?, 2013).

Första versionen av vad som blev känd som PHP skapades 1995 av en man som vid namnet Rasmus Lerdorf. Rasmus Lerdorf behövde någonting föra att göra det enklare att skapa innehåll på hans hemsida, något som skulle fungera bra med HTML men ändå ge honom mera flexibilitet än vad HTML kunde erbjuda honom. Vad han behövde var ett enkelt sätt att skriva skript som skulle köras på sin webbserver både för att skapa innehåll och hantera data som skickas tillbaka till servern från webbläsaren. Genom att använda Perl-språket skapade han en teknik som gav honom vad han behövde och bestämde sig för att kalla denna teknik "Personal Home Page/Forms Interpreter" tekniken gav ett bekvämt sätt att bearbeta webbformulär och skapa innehåll. Namnet "Personal Home Page / Forms Interpreter" förkortades senare till PHP/FI och har så småningom bytt namn för att representera "PHP: Hypertext Preprocessor". PHP/FI version 1.0 användes aldrig riktigt utanför Rasmus egen webbplats. Med introduktionen av PHP/FI version 2.0 började detta förändras. PHP/FI gick igenom två revisioner innan den blev fullständigt omskriven av Andi Gutmans och Zeev Suraki 1997.

PHP 3.0 var den första versionen som gick under namnet PHP Hypertext Preprocessor. Det var också den första versionen som stödde olika databaser, protokoll och API.

I slutet av 1998 såg Zeev Suraki och Andi Gutmans tillbaka på sitt arbete i PHP 3.0 och kände att de kunde ha skrivit skriptspråket ännu bättre, så de började med en ny omskrivning i Maj 2002 så publicerades PHP 4.0. PHP 4.0 är den första versionen som använder sig av en programtolk kallas "Zend Engine" som gör att PHP-skript laddas i minnet och sammanställs till Zend-opkoder, dessa opkoder utförs och HTML generas och resultatet skickas till klienten.

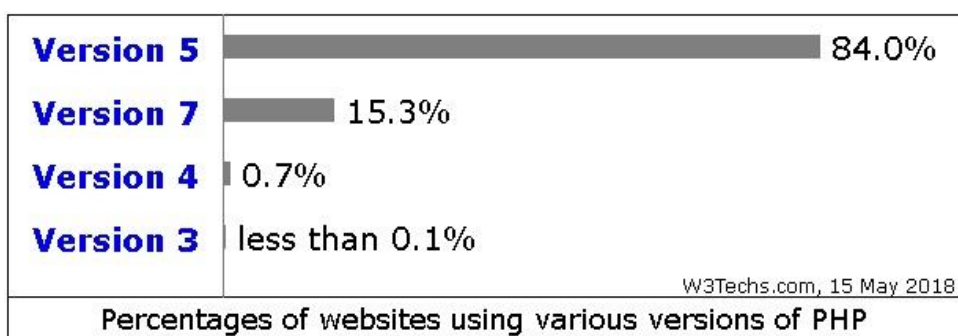
Den viktigaste förändringen kom i PHP 4.1.0 som introducerade superglobaler som `$_GET` och `$_POST`. Superglobaler är fördefinierade PHP variabler vilket innebär att de alltid är



tillgängliga och man kan komma åt dem från någon funktion, klass eller fil utan att behöva deklarerera dem.

I Juli 2004 så släpptes PHP 5.0 som använder sig av "Zend Engine 2". PHP 5.0 innehåller nya funktioner som förbättrad stöd för objektorienterad programmering.

Nuvarande version av PHP är version 7, som blev tillgänglig 2015. Den drivs av Zend Engine 3.0. I figur 4 visas den procentuella användningen av de olika versionerna av PHP i maj 2018.



Figur 4. Visar Statistik tagen på PHP användning på webbsidor tagen från w3techs.com.

I PHP för att kunna få en utskrift på en webbsida så används:

- Echo som är ett kommando som matar ut en eller flera strings man behöver inte använda sig av parenteser.
- Print som är ett kommando som matar ut en eller flera strings men använder inte heller parenteser.

Kodexempel 3. Användning av PHP för att skriva ut text med echo och print.

```
<!DOCTYPE html>
<html>
<body>

    <?php
        echo "<h1>Använder sig utav funktionen echo</h1>";
        echo "Echo this!";

        print "<h1>Använder sig utav funktionen print</h1>";
        print "Print this!<br>";
    ?>

</body>
</html>
```

## Använder sig utav funktionen echo

Echo this!

## Använder sig utav funktionen print

Print this!

Figur 5. Resultat av kodexempel 3.

### 3 Utförande

Denna del omfattar det praktiska utförandet av examensarbetet. Till det praktiska hörde att programmera en webbsida i Anvias interna portal Triton. Triton är en webbapplikation som utvecklats inom Anvia. Triton fungerar med alla HTML5-kompatibla webbläsarenheter. Det finns många olika verktyg i Triton som hjälper olika avdelningar inom Anvia. Triton hämtar kunduppgifter från Nav för att kunna upprätthålla de tjänsterna som Anvia har tillgängliga åt kunder. Triton har över 400 registrerade användare och ca 300 användare varje månad. Varje användare har egna inloggningsuppgifter som fungerar endast i Triton. De mest använda funktionerna i Triton är

- Söka och visa kundinformation och tjänster
- Sammanställning av anslutningarnas information t.ex. bredbandets information tas från Microsoft Dynamics Nav för att få kunduppgifterna.
- "Saatavuuskysely" kontrollerar tillgängligheten med hjälp av Tigers.

#### 3.1 Planering

Efter valet av tekniker var nästa skede att planera hur man skulle bygga upp funktionen. Eftersom att det inte fanns något specifikt sätt på hur webbsidan skulle byggas upp så var det helt upp till en själv hur man skulle gå till väga för att förverkliga arbetet. Tillsammans med uppdragsgivaren och användarna av denna funktion beslutades hur sidan skulle se ut och vilka produkter och tjänster som man skulle ha med i listorna. Vid planeringen bestämdes det även att man skulle kunna spara uppgifterna i en Excel-fil på förmannens nätverksenhet.

#### 3.2 Webbsidan "Seuranta"

För att kunna börja programmera webbsidan Seuranta så krävdes det att man skulle få rättigheter till Anvias server där alla filer lagras samt en kopia på Tritons källkod så att man inte i misstag skulle ladda upp någon fil som skulle förstöra systemen. Triton är uppbyggt på både SOAP och REST. Den äldre delen av Triton var uppbyggd på SOAP som under en längre tid var det protokollet som användes men efter några år så började man inom Anvia flytta över till REST, största delen är nu programmerat i REST.

SOAP (Simple Object Access Protocol) är ett XML-baserat meddelandeprotokoll för utbyte av information mellan datorer.

REST (Representational State Transfer) använder HTTP-protokoll för datakommunikation.

Största skillnaderna mellan SOAP och REST är:

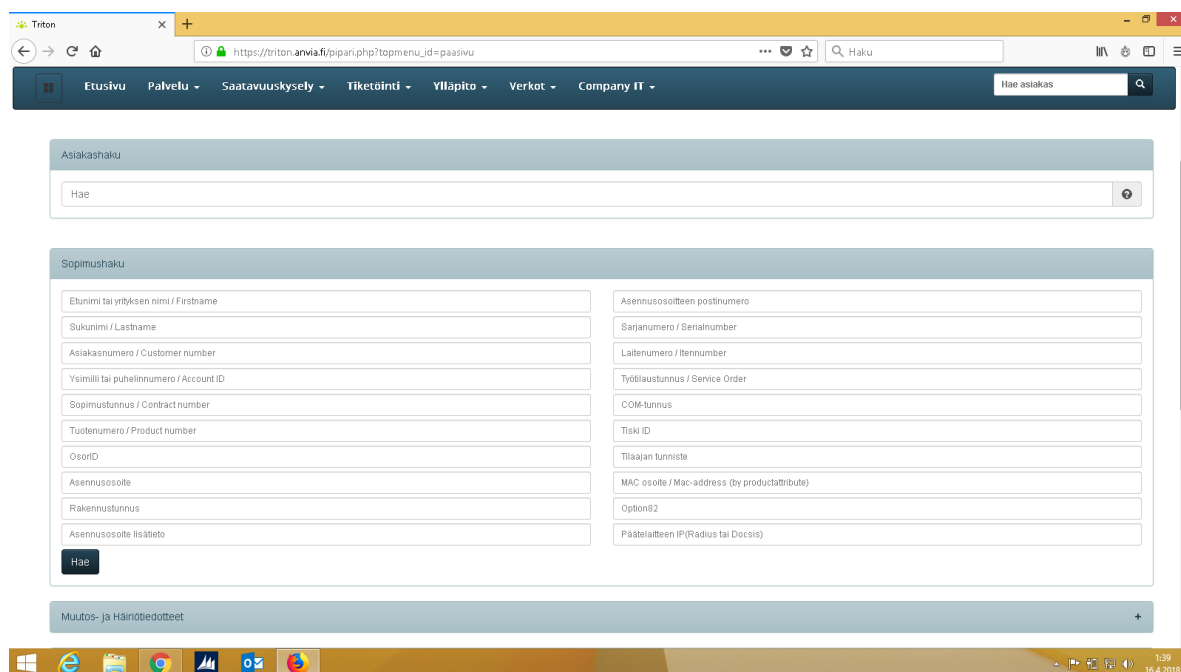
- REST API har ingen officiell standard alls eftersom det är en arkitektonisk stil. SOAP API har å andra sidan en officiell standard eftersom det är ett protokoll.
- REST API använder flera standarder som HTTP, JSON, URL och XML medan SOAP API bygger till stor del på HTTP och XML.
- REST API är mer lämpliga med JavaScript och kan även implementeras enkelt. SOAP API är också praktiskt med JavaScript men stöder inte för större implementering.

(SoapUI, 2018)

Webbsidan ”Seuranta” använder sig utav REST.

### 3.2.1 Installation av Webbsidan ”Seuranta”

För att börja programmera Webbsidan ”Seuranta” behövdes mera information på vilket sätt den interna portalen Triton var uppbyggd. Då man hade fått en förståelse på hur Triton var uppbyggd så valdes det att webbsidan skulle länkas in under menyn ”Ylläpito”



Figur 6. Bild på den interna portalen Triton.

För att webbsidan skulle bli tillgänglig från Triton, så behövde koden inkluderas i Triton. Kodexempel 4 visar anropen.

*Kodexempel 4. Kallar på funktionerna som finns på servern*

```
require_once(dirname(__FILE__).'/SEURANTO.class.php');  
require_once(dirname(__FILE__).'/../TritonBase.class.php');
```

Nu kunde användarna nå webbsidan via menyn ”Ylläpito” i Triton.

### **3.2.2 Enkäten**

För att göra det så enkelt som möjligt för användaren valdes att skapa ett formulär där man skall fylla i försäljaren, kundens namn, kundens kundnummer, produkter eller tjänsten och datumet. Formuläret skapades med elementet form som används för att skapa en HTML-blankett för användarinmatning. Textrutorna där data ska matas in är gjorda med elementet input som anger ett inmatningsfält där användaren kan mata in data. För att avskilja stycken från varandra användes elementet div som definierar en sektion i HTML-dokumentet. För att veta vad användaren skall fylla in i varje fält namngavsfälten med elementet label som definierar en etikett för ett input-element.

## Kodexempel 5. Enkätens källkod

```

<br />
<div class="container">
  <br />
  <div class="col-md-6" style="margin:0 auto; float:none;">
    <form method="post">
      <h3 align="center">Seuranto</h3>
      <br />
      <?php echo $error; ?>
      <div class="form-group">
        <label>Myyjä</label>
        <input type="text" name="myyja" class="form-control" />
      </div>
      <div class="form-group">
        <label>Nimi</label>
        <input type="text" name="nimi" class="form-control" />
      </div>
      <div class="form-group">
        <label>Asiakasnumero</label>
        <input name="asnr" class="form-control" />
      </div>
      <div class="form-group">
        <label>Tuote/Palvelu</label>
        <input type="text" name="tuote" class="form-control" />
      </div>
      <div class="form-group">
        <label>Päivämäärä</label>
        <input type="text" name="date" class="form-control" />
      </div>
      <div class="form-group" align="center">
        <input type="submit" name="submit" class="btn btn-info" value="Submit" />
      </div>
    </form>
  </div>
</div>

```

För att man ska kunna skicka in uppgifterna så krävs det att alla fält är ifyllda så att man ska kunna kontrollera så att alla uppgifter stämmer. Kontrollen är gjord med en if-else sats som kör den ena delen av koden om ett villkor är sant annars körs andra delen. Kodexempel 6 visar t.ex. att om fältet ”myyjä” är tomt så genereras ett felmeddelande ”Kirjoita nimesi”.

## Kodexempel 6. If-Else sats.

```

if(isset($_POST["submit"]))
{
    if(empty($_POST["myyja"]))
    {
        $error .= '<p><label class="text-danger">Kirjoita nimesi</label></p>';
    }
    else
    {
        $myyja = clean_text($_POST["myyja"]);
    }
}

```

## Kirjoita nimesi

### Myyjä

### Nimi

### Asiakasnumero

### Tuote/Palvelu

### Päivämäärä

Figur 7. Fel meddelandet "Kirjoita nimesi" som if-else satsen genererar kodexempel 6.

### 3.2.3 Spara uppgifterna

För att kunna spara uppgifterna måste alla kraven i enkäten uppfyllas, Dvs. alla fälten måste vara ifyllda. Då alla fälten är ifyllda så skickar man uppgifterna genom att trycka på knappen "submit". Programmet kontrollera och kör koden för if-else satserna om alla kraven möts kommer ett meddelande ovanför enkäten "Lähetetty" och textfälten töms och uppgifterna blir skickade till en Excel-fil som är sparad på förmannens nätverksenhet.

Kodexempel 7. Koden för att lägga till uppgifter i Excel Filen.

```

if($error == "")
{
    $file_open = fopen("myynti_kuukausi.csv", "a");
    $no_rows = count(file("myynti_kuukausi.csv"));
    if($no_rows > 1)
    {
        $no_rows = ($no_rows - 1) + 1;
    }
    $form_data = array(
        'sr_no'    =>    $no_rows,
        'myyja'   =>    $myyja,
        'nimi'    =>    $nimi,
        'asnr'    =>    $asnr,
        'tuote'   =>    $tuote,
        'date'    =>    $date
    );
    fputcsv($file_open, $form_data);
    $error = '<label class="text-success">Lähetetty</label>';
    $myyja = "";
    $nimi = "";
    $asnr = "";
    $tuote = "";
    $date = "";
}

```

Kodexempel öppnar filen myynti\_kuukausi.csv ifall alla kraven möts. Programmet öppnar filen och räknar raderna i Excel-tabellen och om datat är korrekt så läggs en ny rad till i Excel-filen.



**Lähetetty****Myyjä****Nimi****Asiakasnumero****Tuote/Palvelu****Päivämäärä**

Figur 8. Resultatet uppgifterna skickades till Excel-filen.

Då månaden har gått hamnar man att skapa en kopia av myynti\_kuukausi.csv filen och döpa om den gamla myynti\_kuukausi.csv filen till det namn som man vill ha t.ex. myynti\_huhtikuu18.csv. Då en kopia är skapad så döper man den bara till myynti\_kuukausi.csv så känner programmet igen att filen finns på rätt plats för att kunna lagra uppgifterna som blir skickade.

## 4 Resultat

Resultat av examensarbete är en fungerande webbsida som anställda i butiken fyller i då man har sålt en viss produkt eller tjänst för att kunna följa med vilka produkter som säljs under en viss tid. Förutom att den började användas i butiken så började kontaktcentret även använda en modifierad version av denna för att följa upp vilka kunder som hade begärt ringbud och om de hade diskuterat med någon skild person.

## 5 Diskussion

Detta har varit ett mångsidigt och intressant arbete, men utmanande. Det som kanske var mest utmanande var hur Webbsidan skulle integreras i Triton. Eftersom servern där all källkod finns har varit i användning i flera år och det fanns många olika filer som man hamnade att lägga till i sin kod för att få det hela att synkas med varandra. Mina kunskaper inom webbutveckling har också förbättrats mycket genom detta arbete.

## 5.1 Vidareutveckling

Planen var att i framtiden utveckla denna tjänst så att den skulle vara kopplad automatisk till anställdas egna inloggningsuppgifter på Triton. Den skulle uppdateras så att man inte behövde ha några skilda användarnamn eller lösenord utan man skulle logga in på datorn men sin e-post och så skulle Triton känna igen det. Eftersom Elisa köpte upp Anvia så har Elisa ingen användning av Triton längre eftersom Elisa har sina egna system som de använder men Viria, ett dotterbolag till Anvia, har dock användning av Triton. Viria använder fortfarande Triton för att ta emot felanmälningar för de tjänster som de erbjuder och för att skicka ut arbetsorderna till montörerna.

## 5 Källförteckning

Ahuka Communications [online]

[http://www.ahuka.com/?page\\_id=99](http://www.ahuka.com/?page_id=99) (hämtat 15.5.2018)

Java-samples [online]

<http://www.java-samples.com/showtutorial.php?tutorialid=889> (hämtat 15.5.2018)

McFarland, D. J. (2009) *CSS: The Missing Manual*. Sebastopol: O'Reilly Media

Refsnes, H., Refsnes, S., & Refsnes, K. J. (2010) *Learn HTML and CSS with w3schools* New Jersey: Wiley Publishing

SoapUI [online]

<https://www.soapui.org/learn/api/soap-vs-rest-api.HTML> (hämtat 15.5.2018)

Theseus [online]

[https://www.theseus.fi/bitstream/handle/10024/55970/Stolpe\\_Mats.pdf?sequence=1&isAllowed=y](https://www.theseus.fi/bitstream/handle/10024/55970/Stolpe_Mats.pdf?sequence=1&isAllowed=y) (hämtat 15.5.2018)

Tutorialspoint [online]

[https://www.tutorialspoint.com/mvc\\_framework/mvc\\_framework\\_introduction.htm](https://www.tutorialspoint.com/mvc_framework/mvc_framework_introduction.htm)  
(Hämtat 11.7.2017)

W3C. HTML & CSS. [online] <http://www.w3.org/standards/webdesign/HTMLcss> (Hämtat: 12.05.2017)

W3resources [online] <https://www.w3resource.com/> (Hämtat: 12.05.2017)

W3schools [online] <https://www.w3schools.com/default.asp> (Hämtat: 12.05.2017)

W3techs [online]

<https://w3techs.com/technologies/details/pl-php/all/all> (hämtat 15.5.2018)

What is PHP? [online] <http://php.net/manual/en/intro-what-is.php> (Hämtat: 12.05.2017)