



VAASAN AMMATTIKORKEAKOULU  
VASA YRKESHÖGSKOLA  
UNIVERSITY OF APPLIED SCIENCES

Anders Palmroos

SIMPLEX – EN ENKEL WEBBLÄSARE  
MED ETT SIMPELT  
ANVÄNDARGRÄNSSNITT

Företagsekonomi och turism

2013

## FÖRORD

Detta lärdomsprov har gjorts under tiden juni 2012 till januari 2013 i syftet att avlägga tradenomexamen inom utbildningsprogrammet för informationsbehandling vid Vasa yrkeshögskola.

Jag vill tacka Robert Metsäranta för den hjälp jag fick då jag stötte på problem med programmeringen i Qt Creator. Börje Harju har fungerat som handledare.

Vasa den 21 februari 2013

---

Anders Palmroos

## VASA YRKESHÖGSKOLA

Utbildningsprogrammet för informationsbehandling

**ABSTRAKT**

Författare	Anders Palmroos
Lärdomsprovets titel	Simplex – en enkel webbläsare med ett simpelt användargränssnitt
År	2013
Språk	svenska
Sidantal	77 + 15 bilagor
Handledare	Börje Harju

---

Syftet med detta lärdomsprov är att skapa en enkel webbläsare med ett simpelt användargränssnitt för äldre personer. Tanken bakom detta är att en del personer kanske inte behöver eller har nytta av en avancerad webbläsare.

Detta lärdomsprov innehåller funderingar kring hur en webbläsare skall göras enkel, lättanvänd och användbar. Det innehåller också uppbyggnaden av webbläsaren med hjälp av programmering. De vanligaste webbläsarnas för- och nackdelar tas upp. De vanligaste webbläsarna på marknaden samt målgruppens önskemål bidrog till riktlinjer för skapandet av webbläsaren.

Resultatet av arbetet är en enkel webbläsare som har tagit målgruppens riktlinjer i beaktande vid skapandet. Programspråket C++ (Qt) har använts för att programmera webbläsaren och som program har Qt Creator använts. Internet har använts som informationskälla för lärdomsprovet då ingen lämplig litteratur i bokform har hittats.

---

Ämnesord	Simpel, användargränssnitt, webbläsare, Qt, programmering
----------	---

VAASAN AMMATTIKORKEAKOULU

UNIVERSITY OF APPLIED SCIENCES

Utbildningsprogrammet för informationsbehandling

## ABSTRACT

Author	Anders Palmroos
Title	Simplex – an Easy Web Browser With a Simple User Interface
Year	2013
Language	Swedish
Pages	77 + 15 Appendices
Name of Supervisor	Börje Harju

---

The purpose of this thesis is to create an easy web browser with a simple user interface for the elderly. The idea behind this is that some people may not need or gain benefit from an advanced browser.

This thesis contains reflections on how a web browser should be made simple, easy to use and useful. It also contains the structure of a web browser with programming. The most common web browsers advantages and disadvantages are described. The most common web browsers on the market and targeted audience preferences contributed to guidelines for the creation of the browser.

The result of this thesis is a simple web browser that has taken the target group guidelines into consideration during the creation. The programming language C++ (Qt) have been used to program the web browser and Qt Creator have been used as the program to do it in. The Internet has been used as information source for the thesis as no suitable literature has been found in book form.

---

Keywords                      Simple, User interface, Web browser, Qt, Programming

## INNEHÅLL

FÖRORD .....	2
ABSTRAKT.....	3
ABSTRACT.....	4
1 BETECKNINGAR OCH FÖRKORTNINGAR .....	8
2 INLEDNING .....	15
2.1 Bakgrund.....	15
2.2 Problemområden.....	16
2.3 Syfte och mål .....	16
2.4 Avgränsningar.....	16
3 DE VANLIGASTE WEBBLÄSARNA OCH DERAS FÖR- OCH NACKDELAR .....	18
3.1 Mozilla Firefox .....	18
3.2 Internet Explorer .....	20
3.3 Opera.....	22
3.4 Safari .....	24
3.5 Google Chrome.....	26
4 WEBBLÄSARTESTER.....	29
4.1 Peacekeeper.....	29
4.2 Sunspider.....	30
4.3 V8.....	31
4.4 HTML 5 .....	32
4.5 Acid 3.....	33
4.6 RAM-minne test.....	34
5 VAD ÄR ENKELHET? .....	36
5.1 Enkelhet ur en webbläsares synvinkel .....	37
6 VAD ÄR ANVÄNDBARHET?.....	39
6.1 Varför är användbarhet viktigt?.....	41
6.2 Hur förbättra användbarheten? .....	42
6.3 Användbarhet ur en webbläsares synvinkel.....	42
7 MÅLGRUPPEN OCH DERAS ÖNSKEMÅL .....	44
7.1 Undersökning.....	44

7.2	Resultatet av undersökningen .....	44
7.3	Tillämpning av undersökningens resultat .....	53
8	QT.....	55
8.1	Vad är Qt?.....	55
8.2	Produkter.....	55
8.3	Vem ligger bakom Qt?.....	56
9	PROGRAMMERING AV WEBBLÄSAREN.....	57
9.1	Planering .....	57
9.2	Kodning.....	58
9.2.1	Huvudmeny och verktygsfältet .....	59
9.2.2	Navigeringsfältet .....	60
9.2.3	Huvudmenyns funktioner.....	63
9.2.4	Flikar och fönster .....	67
9.2.5	Sökfunktionen .....	68
9.2.6	Inställningar för webbläsaren.....	68
9.2.7	Bokmärken .....	69
9.2.8	Högerklicksmenyer .....	71
9.2.9	Sista finputsningen.....	71
10	RESULTAT.....	73
11	SLUTLEDNING .....	74
	KÄLLOR .....	75
	BILAGOR	

## **FÖRTECKNING ÖVER BILAGOR**

**BILAGA 1.** Instruktioner för installation av Simplex

**BILAGA 2.** Undersökningen i E-lomake

**BILAGA 3.** Logon för webbläsaren Simplex

## 1 BETECKNINGAR OCH FÖRKORTNINGAR

Inom ämnet finns det en hel del beteckningar och förkortningar som borde förklaras för att alla skall förstå vad de innebär. Här följer en lista över dessa i alfabetisk ordning.

Adressfält	På engelska adress field. Adressfältet visar adressen eller URL:en av webbsidan man besöker. Man kan skriva in webbadressen man vill gå till i adressfältet.
Användargränssnitt	Den del av ett datorprogram som användaren kommer i kontakt med. Exempelvis tangentbord, mus och allt som syns på skärmen.
Bakåtknapp	En bakåtknapp går bakåt ett steg i historiken, till webbsidan man befann sig på senast innan nuvarande.
Bokmärken	Med hjälp av bokmärken kan man spara och organisera webbadresser i olika mappar för att hitta dem snabbt och enkelt vid ett senare tillfälle. Man kan också importera/exportera bokmärken i de flesta webbläsare. I Internet Explorer kallas de för <i>favoriter</i> istället för bokmärken.
C++	C++ är ett objektorienterat programspråk som är baserat på programspråket C. Det är ett av de populäraste programspråken idag.
Flik(ar)	På engelska tab(s). Flikar ger möjlighet att ladda flera webbsidor utan att öppna ett nytt fönster. Man kan flytta och stänga flikar hur man vill. Snabbkommandot för att få en ny flik är oftast Ctrl+T.
Framåtknapp	En framåtknapp går framåt ett steg i historiken. D.v.s. om man nyss klickat på bakåtknappen kan man ångra



sig igen och gå framåt till webbsidan man befann sig på förrän man klickade på bakåt.

Fönster	På engelska window. Om man öppnar ett nytt fönster så syns det i Windows verktygsfält som ett till "program" av samma program. Det rekommenderas att man använder flera flikar istället för flera fönster. Flikar är endast användbara då det gäller visning av webbsidor och inte i program som t.ex. Microsoft Word, Microsoft Excel osv. då man i de programmen inte kan öppna en ny flik då man skapar ett nytt dokument utan det dyker upp i ett nytt fönster. Snabbkommandot för att få ett nytt fönster är oftast CTRL+N.
Gränssnitt	På engelska interface. Gränssnitt är en kontaktyta för datorkommunikation, t.ex. människa och dataprogram. Ett exempel på grafiskt användargränssnitt är Microsoft Windows, medan DOS är ett textbaserat gränssnitt.
Hemknapp	En knapp som tar dig tillbaka till startsidan om du klickar på den.
Historik	Om man vill ha reda på vilka webbsidor man besökt tidigare kan man kolla upp det från webbläsarens historik.
HTML	HTML står för Hypertext Markup Language och är ett standardiserat programkodsspråk för att visa webbsidor åt användaren.
HTML-versioner	Det har under åren kommit ut flera versioner av HTML. Först ut av HTML-versionerna var <b>HTML 2.0</b> som utvecklades av Internet Engineering Task Force HTML Working Group år 1996. Första versionen kom dock redan i slutet av år 1995. HTML 2.0 är en föråldrad

version av HTML. För en webbutvecklare finns det inget behov att studera HTML 2.0-standarden. Vissa webbsidor på nätet nämner första versionen som HTML 1.0 men då W3C inte existerade då och 1.0 hade massa begränsningar är det inte en officiell version. HTML 1.0 pågick mellan 1989–1994.

Nästa version som kom ut var **HTML 3.2** som blev en rekommendation av W3C den 14 januari 1997. HTML 3.2 lade till nya funktioner såsom teckensnitt, tabeller, applets, upphöjda tecken, index och mer till den befintliga HTML 2.0-standarden. Ett av de element som lades till i HTML 3.2-standarden var "<font>" taggen.

Senare samma år, den 18 december 1997 blev **HTML 4.0** en rekommendation av W3C. En andra utgåva offentliggjordes den 24 april 1998 med endast några redaktionella ändringar. Den viktigaste egenskapen hos HTML 4.0 var införandet av stilmallar (CSS).

Den 24 december 1999 blev **HTML 4.01** en W3C rekommendation. HTML 4.01 var en mindre uppdatering av korrigeringar och bug-fixar från HTML 4.0.

**XHTML 1.0** blev en W3C rekommendation den 20 januari 2000. XHTML 1.0 omformulerar HTML 4.01 i XML.

Sedan dröjde det ett bra tag innan nästa version kom. Den 22 januari 2008 publicerade W3C ett fungerande utkast för **HTML 5**. HTML 5 gör exakta regler för hur man ska hantera alla HTML-element och hur man

återhämtar sig från fel vilket förbättrar kompatibiliteten och minskar utvecklingskostnader. Några av de nya funktionerna i HTML 5 är funktioner för inbäddning av ljud, video, grafik, datalagring på klientsidan och interaktiva dokument. HTML 5 innehåller också nya element som "<nav>", "<header>", "<footer>" och "<figure>". Arbetsgruppen för HTML 5 inkluderar AOL, Apple, Google, IBM, Microsoft, Mozilla, Nokia, Opera och många hundra andra leverantörer. HTML 5 är långt ifrån färdig men förväntas bli riktigt bra.

HTTP	Det protokoll som ligger till grund för överföring av dokument i World Wide Web.
HTTPS	HTTPS är en fusion av HTTP och SSL (Secure Sockets Layer) vilket betyder att det är säkrare än HTTP. Det används av bland annat nätbanker för att skydda kundernas uppgifter och för att kryptera data som överförs.
Hyperlänk	Text eller bild som är klickbara och för en vidare till ett annat dokument.
Högerklicksmeny	En meny som öppnas då man klickar med höger musknapp på någonting.
IP-adress	Logisk adress som tilldelas alla datorer på Internet. Varje Internet-ansluten dator måste ha en unik IP-adress.
JavaScript	Ett objektbaserat scriptspråk med vars hjälp man kan ändra och visa objekt som man skapat med HTML.
Menyfält	På engelska menu bar. Ett menyfält brukar inkludera standardmenyer som t.ex. Arkiv, Redigera, Hjälp osv.

	<p>Innehåller de flesta åtgärderna man kan göra med programmet. Kan även kallas huvudmeny.</p>
Navigeringsfält	<p>På engelska navigation bar. Ett fält som inkluderar navigation över webbsidor, oftast bakåt/framåtknapp, adressfält, uppdatera knapp, stoppknapp, hemknapp och sökruta för t.ex. Google. Kan även kallas adressrad eller liknande.</p>
Plug-in (tillägg)	<p>Ett program som anropas av huvudprogrammet för att tillföra nya funktioner.</p>
Qt	<p>Qt är ett applikationsramverk för Windows, Mac OS X och Linux med vilket man inte behöver ändra om koden för distribution till olika plattformar. Qt använder sig av programspråket C++. Nokia ägde tidigare Qt men sålde det i september 2012 till Digia.</p>
RSS-läsare	<p>RSS är en förkortning för "Really Simple Syndication". Varje webbplats kan ge ut ett RSS-flöde. Genom att prenumerera på ett RSS-flöde meddelas du då det har hänt något nytt på webbsidan sedan du läste den sist. I huvudsak nyhetssidor och bloggar använder sig av RSS-flöden. För att läsa RSS-flöden måste man ha en RSS-läsare. Nuförtiden finns RSS-läsare i många webbläsare så man behöver inte ha något skilt program för det ändamålet.</p>
Startsida	<p>Webbsida som visas när webbläsaren startar. Kan ändras från inställningarna i webbläsaren.</p>
Statusfält	<p>På engelska status bar. Ett fält som finns längst ner i webbläsarfönstret.</p>

Stoppknapp	Knapp som man använder för att sluta ladda sidan, t.ex. om man ångrar sig och inte vill att sidan skall hinna ladda kan man klicka på denna knapp.
Sökfunktion	En funktion som söker efter texten man skrivit in för att hitta den på webbsidan man befinner sig på. Man kan gå till nästa eller föregående sökträff. Snabbkommandot är oftast CTRL+F.
Sökruta	På engelska search box. En sökruta där man kan välja mellan olika sökmotorer för att söka efter saker och ting på nätet.
Titelfält	På engelska title bar. Text som visas högst upp i webbläsarfönstret. Är oftast en beskrivande text över innehållet som finns på webbsidan.
Uppdatera knapp	En knapp som används för att uppdatera webbsidan man befinner sig på. Kan vara nödvändigt om webbsidan har varit öppen en längre tid och saker och ting har ändrat sedan sist. Exempel på detta är nyhetssidor.
URL	En standardiserad teckensträng som identifierar en webbplats. Exempelvis <a href="http://www.puv.fi">http://www.puv.fi</a> .
Verktygsfält	På engelska toolbar. En rad med klickbara verktyg för direkt åtkomst till dem (oftast funktioner som man brukar använda ofta). Kan även kallas verktygsrad eller liknande.
W3C	W3C står för World Wide Web Consortium och är ett internationellt community där medlemsorganisationer, personal och offentliga arbetar tillsammans för att utveckla webbstandarder.

Webbläsare	Program som visar webbsidor på skärmen, och som låter dig använda hypertext-funktionerna i WWW. Webbläsaren känner av musklick på länkade områden av webbsidor och hämtar det länkade dokumentet åt användaren.
Webbsida	På engelska web page. En sida på Internet som t.ex. kan innehålla text, bilder, filmer osv.
WWW	Hypertextbaserat system för länkning av dokument över Internet.
Öppen källkod	Datorprogram som vem som helst har åtkomst till källkoden på och kan använda, läsa, ändra och distribuera vidare. [7] [12] [26] [28] [29]

## 2 INLEDNING

Det finns en hel del webbläsare på marknaden. De mest vanliga är Mozilla Firefox, Internet Explorer, Opera, Safari och Google Chrome. Det man kan ifrågasätta är om de faktiskt är lätta att använda och enkla nog för att alla skall kunna känna sig nöjda. Eftersom marknaden är hård och webbläsarna tävlar om att få mest marknadsandelar bygger de in så mycket verktyg och egenskaper som möjligt. Hur är det då med personer som inte behöver alla dessa verktyg och vill ha en lite enklare webbläsare, med stora knappar till exempel. Finns det en sådan webbläsare? Svaret är tyvärr nej. Detta har marknaden inte tänkt på.

Med mitt lärdomsprov skulle jag programmera en webbläsare med tanke och fokus på målgruppens önskemål och riktlinjer. Webbläsaren har som syfte att vara enkel att använda och bara innehålla de mest nödvändiga funktionerna. Webbläsarens målgrupp skulle vara främst äldre personer. För att få önskemål från min målgrupp skapade jag en undersökning i E-lomake och resultaten av undersökningen beaktades vid skapandet av webbläsaren.

### 2.1 Bakgrund

Då jag funderade över ett ämne för lärdomsprovet i våras dök plötsligt en idé upp som jag gillade mycket. Min idé var att jämföra de populäraste webbläsarna på marknaden samt programmera en enkel webbläsare med ett simpelt användargränssnitt för äldre personer. De populäraste webbläsarna på marknaden är inte anpassade för äldre personer som förmodligen vill ha större knappar och fält. Jag funderade ändå vidare ett bra tag på ämne för lärdomsprovet för att försäkra mig om att inte välja första bästa idé.

Eftersom programmering inte är min starka sida tyckte jag det kunde vara bra att bättra på det genom lärdomsprovet. Jag beslöt mig helt enkelt för att göra en webbläsare som mitt lärdomsprov. Först funderade jag över att programmera webbläsaren i Visual Basic eller dylikt språk men då en klasskamrat nämnde Qt och argumenterade för varför jag skulle överväga det istället valde jag att programmera i Qt. Eftersom jag inte hade någon kunskap om programmering i Qt

från tidigare var jag tvungen att lära mig grunderna i det. Detta gjorde jag mest genom att kolla videor på YouTube om just Qt samt läsa hjälpmaterial om programmet. Robert Metsäranta har varit till stor hjälp under programmeringen då han hjälpt mig med problem och gett mig en hel del tips och instruktioner.

## **2.2 Problemområden**

Det finns idag många olika webbläsare med alltför avancerade funktioner och förbryllande gränssnitt. Dessa är möjligen inte ett bra val av webbläsare för alla människor. Därför borde det finnas ett annat alternativ för de individer som inte behöver så avancerade funktioner.

## **2.3 Syfte och mål**

Syftet med detta lärdomsprov är att programmera en enkel webbläsare med ett simpelt användargränssnitt för äldre personer. Jag skall undersöka de vanligaste webbläsarna på marknaden och ta reda på vilka fördelar och nackdelar de har. Med hjälp av en undersökning som görs i E-lomake får jag åsikter om hur webbläsaren skall se ut och dess funktioner. Jag sammanställer sedan undersökningens resultat och försöker tillämpa dem i webbläsaren.

Målet med lärdomsprovet är att läsarna skall få en bild av skillnaderna mellan de vanligaste webbläsarna på marknaden och en insyn om att det behövs en enklare webbläsare för äldre personer på marknaden. Det största målet är dock att lyckas skapa själva webbläsaren och få den att fungera på det vis jag vill (även på andra datorer än min egen dator) samtidigt som jag tagit målgruppens åsikter i beaktande när jag skapat den.

## **2.4 Avgränsningar**

Jag kommer inte att programmera alltför svåra funktioner till webbläsaren som t.ex. tillägg eller dylika avancerade funktioner. Jag kommer bara att ta med de viktigaste och nödvändigaste funktionerna tillsammans med målgruppens önskemål. Övriga webbläsare än de fem vanligaste webbläsarna kommer inte att



tas upp i detta lärdomsprov, förutom då jag kort nämner webbläsaren Arora i samband med en intressant funktion i slutet av lärdomsprovet.

### **3 DE VANLIGASTE WEBBLÄSARNA OCH DERAS FÖR- OCH NACKDELAR**

Jag kommer att redovisa för de vanligaste webbläsarna på marknaden samt utifrån vad andra har kommit fram till deras fördelar och nackdelar. Utifrån resultaten kommer jag att göra riktlinjer för hur min webbläsare skall utformas kombinerat med önskemålen från målgruppen.

En del människor använder bara första bästa webbläsaren som kommer med deras operativsystem. Internet Explorer kommer med Windows, Safari ingår i Mac OS X medan Mozilla Firefox ingår i Linux. Men det finns också två till stora webbläsare, nämligen Google Chrome och Opera. Om man använder Windows som operativsystem har man möjlighet att använda alla dessa fem webbläsare jag nämnt medan Google Chrome, Mozilla Firefox och Opera finns tillgängliga för både Windows, Mac OS X och Linux.

Det är upp till användaren själv att bedöma vilken webbläsare som de gillar mest för människor har olika åsikter om vad som är bra och det finns ingenting som passar alla människor. Det finns alltså inte någonting som heter "bästa webbläsare" utan det är ett personligt val vilken man gillar mest.

#### **3.1 Mozilla Firefox**

Mozilla Firefox är en webbläsare med öppen källkod tillverkad av Mozilla Foundation. Dess historia började 2002 då man gav ut Mozilla 1.0, den första stora versionen. Denna version kom med många förbättringar till webbläsaren, e-postklienten och andra applikationer. Trots det var det inte många som använde den. Vid denna tidpunkt hade Internet Explorer över 90 % av Internet-användarna. [11]

Två år senare släpptes Firefox 1.0 och det blev en stor framgång. På mindre än ett år hade Firefox laddats ner mera än 100 miljoner gånger. Nya versioner av Firefox har kommit ut regelbundet sedan dess och har fortsatt att sätta nya rekord. År

2008 nådde Firefox en marknadsandel på 20 % över hela världen och konkurrensen har påskyndat innovation och förbättrat Internet för alla. [11]

### **Fördelar och nackdelar**

De största fördelarna med Mozilla Firefox är enligt WordPressGuru att den använder sig av öppen källkod samt att det finns massor med tillägg att ladda ner till den. [32]

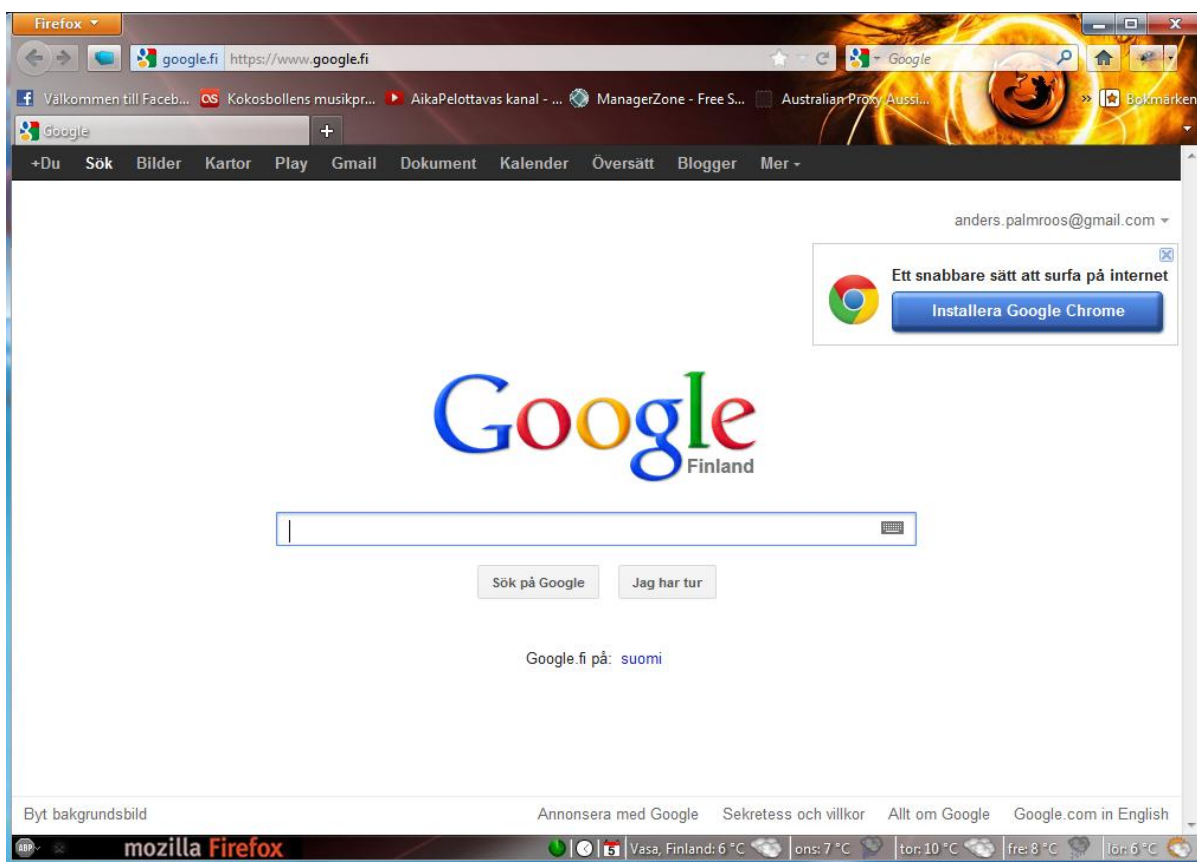
Som andra fördelar har Mozilla Firefox enligt Gizmo's Freeware ett enkelt användargränssnitt, smidig synkronisering, den är snabb, mycket säker, plattformsoberoende och har utmärkt kompatibilitet med webbsidor. Dessutom består Mozilla av en stor utvecklargemenskap. [8]

Pcmag.com nämner förutom det som Gizmo's Freeware nämnt det goda stödet för HTML 5, den innovativa flikimplementationen, grafisk hårdvaruacceleration och att den är väldigt anpassningsbar som fördelarna med Firefox. [18]

Enligt Gizmo's Freeware är nackdelarna med Mozilla Firefox att den inte är så snabb som den kunde vara (Opera och Google Chrome är snabbare) och att man måste starta om Firefox efter att man installerat tillägg. De anser också att Firefox drar mycket minne på äldre datorer, har teckensnitt som är mindre och otydligare och saknar stöd för populära ljud och video codecs. [8]

Pcmag.com nämner nackdelar såsom att det saknar inbyggd Flash-spelare och PDF-läsare och att Firefox ligger lite efter Google Chrome och Opera i HTML 5. [18]

I en jämförelse av webbläsare på Centennial Arts påstås det att en av nackdelarna med Firefox är att man inte kan se förhandsvisningar i miniatyr eller använda musgester. [2]



Figur 1. Mozilla Firefox 11 i Windows 7 miljö. [egen bild]

### 3.2 Internet Explorer

Internet Explorer är Microsoft's webbläsare. Första versionen av Internet Explorer kom ut år 1995 i samband med Windows 95. Sedan dess har många nya versioner kommit ut och nyaste versionen för tillfället är Internet Explorer 10 som kom ut år 2012. [3] [31]

Den är marknadens populäraste webbläsare p.g.a. att den ingår färdigt med operativsystemet Windows.

## **Fördelar och nackdelar**

Enligt WordPressGuru har Internet Explorer egentligen inga större fördelar men version 7 och uppåt är en stabil och relativt välgjord webbläsare. De anser att en häftig funktion i Internet Explorer är Quick Tabs där man kan se alla öppna flikar som miniatyrer. [32]

Gizmo's Freeware anser att fördelarna med Internet Explorer bl.a. är den otroliga hastigheten, minimalistiska gränssnittet, förbättring att följa webbstandarder (i de senaste versionerna), nedladdningshanteraren med skydd mot skadlig programvara, spårningsskydd, hårdvaruaccelerationen samt en bra integrering i operativsystemet (Windows). [8]

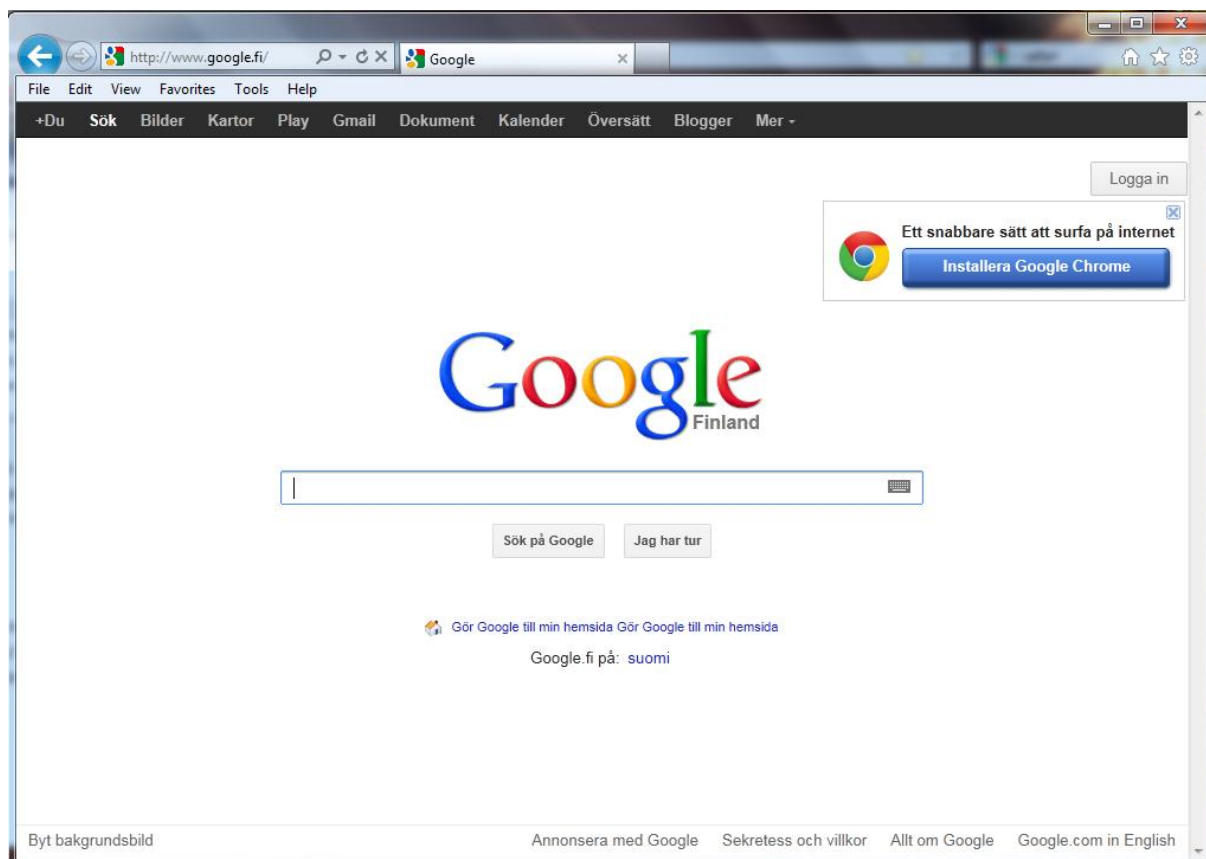
Pcmag.com nämner fördelar som förbättrad flikfunktion och att Internet Explorer har många säkerhetsfunktioner. [20]

Som nackdelar med Internet Explorer nämner Gizmo's Freeware att Internet Explorer 9.0 inte är kompatibel med Windows XP, sämre flikhantering än i Firefox, ingen automatisk återställning av flikar, omstart krävs vid installation och installationstiden är lång. Vidare nämns nackdelar som att den inte har någon synkronisering av bokmärken, vissa inkompatibiliteter med webbsidor och att det inte finns några teman. [8]

Pcmag.com nämner som nackdelar med Internet Explorer att de ligger efter andra webbläsare i HTML 5 stöd och att version 9.0 bara fungerar i Windows Vista och Windows 7. [20]

En jämförelse om webbläsare på Centennial Arts nämner nackdelar såsom brist på säkerhet i ActiveX-kontroller, förväxling mellan 32 och 64 bitars versioner och att det är ett stort mål för hackare och tjuvar på Internet. [2]

Nackdelar som WordPressGuru nämner är bl.a. att de använder en annorlunda tolkning av sidstilar. Istället för att följa standarden som W3C har gett ut använder sig Internet Explorer av sina egna namn och deklARATIONER vilket leder till att webbutvecklare måste ha speciell kod för just Internet Explorer. [32]



**Figur 2.** Internet Explorer 9 i Windows 7 miljö. [egen bild]

### 3.3 Opera

Opera började år 1994 som ett forskningsprojekt inom Telenor, Norges ledande telekom. Idag har Opera över 700 anställda på kontor runt om i världen. Opera är anpassad till ett stort antal plattformar, operativsystem och inbäddade Internet produkter - som inkluderar Mac, PC och Linux datorer, tabletter, mobiltelefoner, handdatorer samt olika spelkonsoler och andra anordningar. [14]

Opera Software har alltid strävat efter att utveckla den snabbaste och mest tekniskt sett avancerade webbläsaren. När den kraftfulla teknologin från Operas standardwebbläsare appliceras till Opera Mini browser gör att den sistnämnda fungerar även på de enklaste mobiltelefonerna med små skärmar och begränsat minne. Detta har gjort att Opera Mini browser är den populäraste webbläsaren för mobiltelefoni i dagsläget. [14]

## Fördelar och nackdelar

Fördelar med Opera är enligt PC för Alla att den har flera smarta extrafunktioner, bl.a. en finess som möjliggör till snabb surfning med hjälp av musrörelser. Vidare nämns fördelar som att den har turboläge och bittorrentklient. Dessutom är den riktigt snabb och var den första webbläsaren som gjorde att man kunde använda sig av flikar istället för fönster. Den var också först med en finess som visar besökta sidor när man startar webbläsaren. [15] [32]

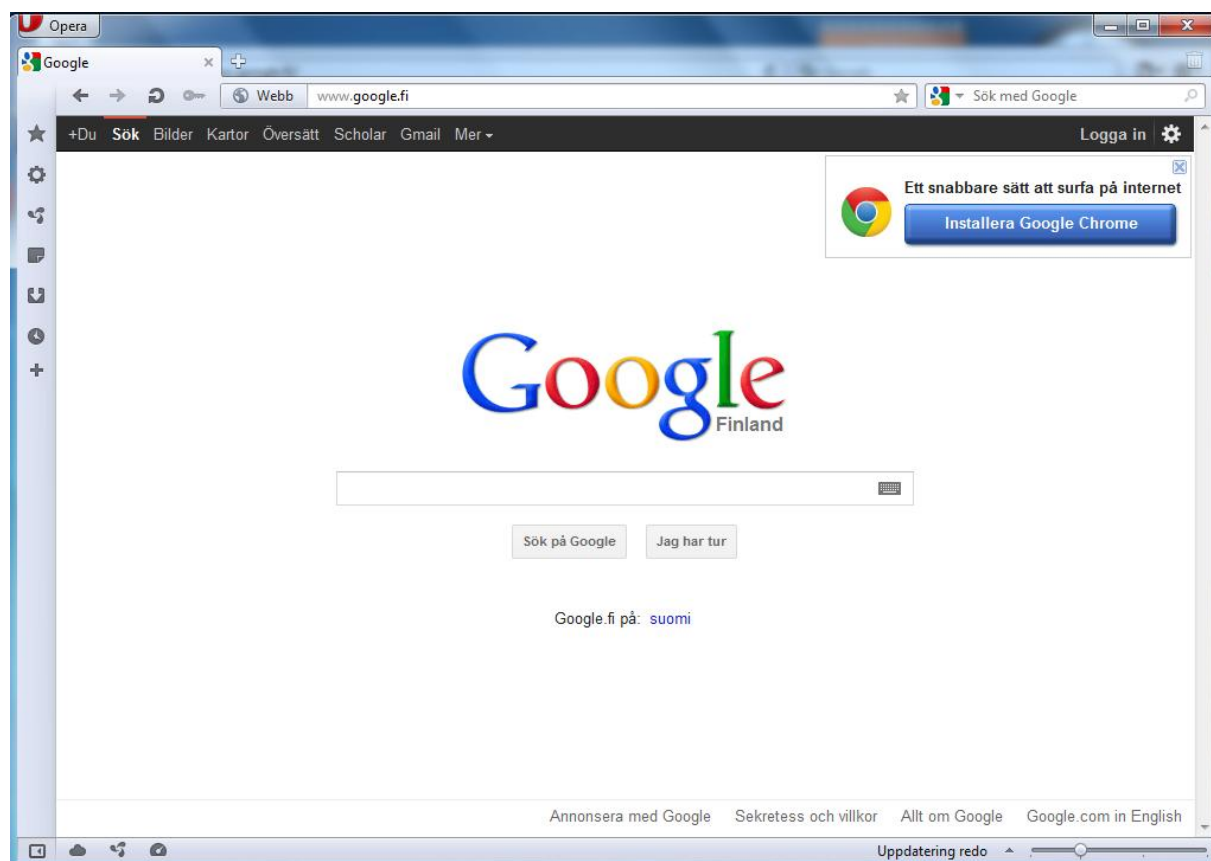
Gizmo's Freeware nämner som fördelar med Opera att den är snabb, funktionsrik, plattformsoberoende, följer webbstandarderna, har inbyggd e-post och torrentklient, stöd för tillägg och teman, stapling av flikar, visuella flikar och musgester, kontosynkronisering och massor av användbara funktioner. [8]

Pcmag.com nämner förutom det som Gizmo's Freeware nämnt det minimalistiska gränssnittet och det utmärkta stödet för HTML 5 som fördelar i Opera. [21]

Gizmo's Freeware nämner som nackdelar för Opera att tilläggs Galleriet inte är så omfattande, inga vanliga (standard) snabbkommandon och ingen möjlighet till säkerhetskopiering. [8]

Pcmag.com nämner som nackdelar med Opera att webbplatser ibland inte visas korrekt och att hårdvaruacceleration inte är aktiverat som standard. [21]

Nackdelar enligt PC för Alla är att den är lite av en outsider i konkurrensen av webbläsare. På mobiltelefonmarknaden är den störst men inom webbläsare för datorer så har den inte alls lika stor andel. Enligt PC för Alla har den inte lika många insticksprogram som Mozilla Firefox och erbjuder inte någonting som inte finns i Firefox. [16]



**Figur 3.** Opera 11.61 i Windows 7 miljö. [egen bild]

### 3.4 Safari

Safari är en webbläsare utvecklad av Apple. Efter årtal av Netscape Navigator och Internet Explorer för Macintosh datorer utvecklade Apple en webbläsare endast för Macintosh. Apple Safari har funnits sedan år 2003 för Macintosh och sedan år 2007 för Microsoft Windows. Den fungerar på iPad, iPhone, iPod Touch, Mac och PC. [1] [6]

Enligt Apple's hemsida är Safari den snabbaste webbläsaren i världen och första att utnyttja HTML 5 ljud och video, vilket tillåter tillverkaren att integrera ljud och video direkt genom traditionell HTML-formatering. [1]

#### **Fördelar och nackdelar**

Fördelarna med Safari är enligt Gizmo's Freeware att den har ett elegant användargränssnitt, snabb prestanda och hastighet då man surfar på nätet, följer webbstandarderna fullt ut, dess Safari-läsare med läslista (Reader), enkel åtkomst



till de mest besökta webbsidorna, inbyggd RSS-läsare samt att den är den bästa webbläsaren för operativsystemet Mac. [8]

Enligt Pcmag.com är fördelarna med Safari dess läsvy, cover flow och andra vackra gränssnittselement, bra flikimplementation och ganska bra urval av tillägg. [17]

Största fördelen med Safari är enligt PC för Alla dess snabbhet. [15]

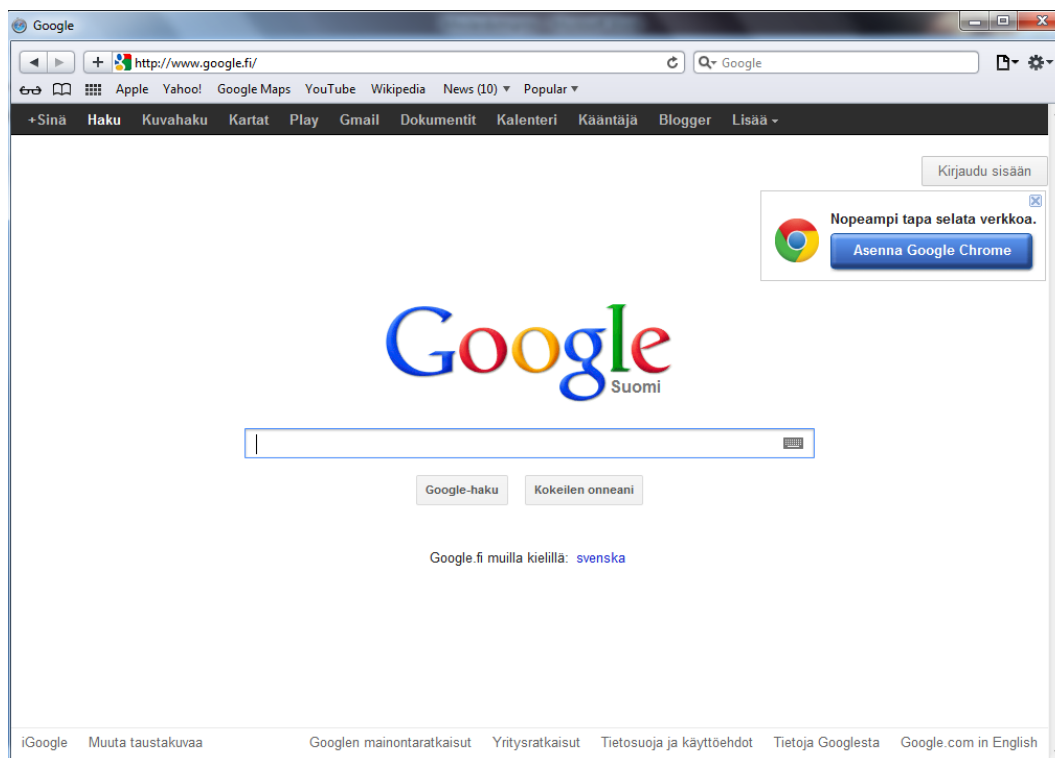
Nackdelarna med Safari är enligt Gizmo's Freeware att tilläggen är begränsade och inte användbara, höga minnes- och systemkrav och att storleksändring på teckensnitt i läsaren (Reader) inte är möjligt. [8]

Enligt Pcmag.com är nackdelarna med Safari dess långsamma uppstart, Windows-versionen av Safari uppdateras inte tillsammans med Mac-versionen och att det inte finns någon hårdvaruacceleration för Windows. [17]

En jämförelse om webbläsare på Centennial Arts nämner nackdelar såsom brist på alternativ att anpassa webbläsaren (något som de flesta konkurrenter erbjuder), problem med den inbyggda sökmotorn Google som inte kan ändras och svårigheter att ta bort cookies automatiskt när man stänger webbläsaren. [2]

WordPressGuru anser att det är lite si och så med användarvänligheten och funktionerna i Safari. Safari innehåller en del onödiga och märkliga funktioner som inte behövs och det kan vara lite svårt att hitta i röran. Som en till nackdel med Safari nämns även hastigheten som inte är värst bra. [32]

Nackdelar med Safari är enligt PC för Alla det gråa och lite tråkiga gränssnittet. [15]



**Figur 4.** Safari 5.1.5 i Windows 7 miljö. [egen bild]

### 3.5 Google Chrome

Google Chrome är en webbläsare utvecklad av Google. Webbläsaren använder sig av WebKit som renderingsmotor. En märkbar skillnad i jämförelse med etablerade webbläsare är att flikhanteringen sköts mer övergripande (och enskilt i förhållande till de andra flikarna), i jämförelse med exempelvis Mozilla Firefox och Internet Explorer. Google-relaterade produkter är integrerade i webbläsaren så att man lätt kan komma åt dem. Första stabila versionen av Google Chrome släpptes december 2008, vilket gör den till en ganska ny webbläsare på marknaden. [30]

#### Fördelar och nackdelar

Största fördelen med Google Chrome är enligt WordPressGuru att olika program från Google är integrerade från början vilket gör att det är lätt att komma åt dem. Andra fördelar är att varje öppen flik fungerar som en egen process vilket leder till säkrare och bättre surfning då en långsam hemsida inte påverkar de andra flikarna.

Adressfältet i Google Chrome fungerar som både adressfält och sökfält vilket är fiffigt uttänkt. En annan speciell sak med adressfältet är att det föreslår adresser som man inte ens besökt tidigare då man börjar skriva in något i adressfältet. [32]

Gizmo's Freeware nämner som fördelar med Google Chrome att den är extremt snabb, clean (ren), den har enkelt och intuitivt gränssnitt, stöder tillägg, man kan surfa i ett inkognito surfningsläge, kontosynkronisering, frekventa och "tysta" uppdateringar och att den har inbyggd Flash-spelare och läsare för PDF. [8]

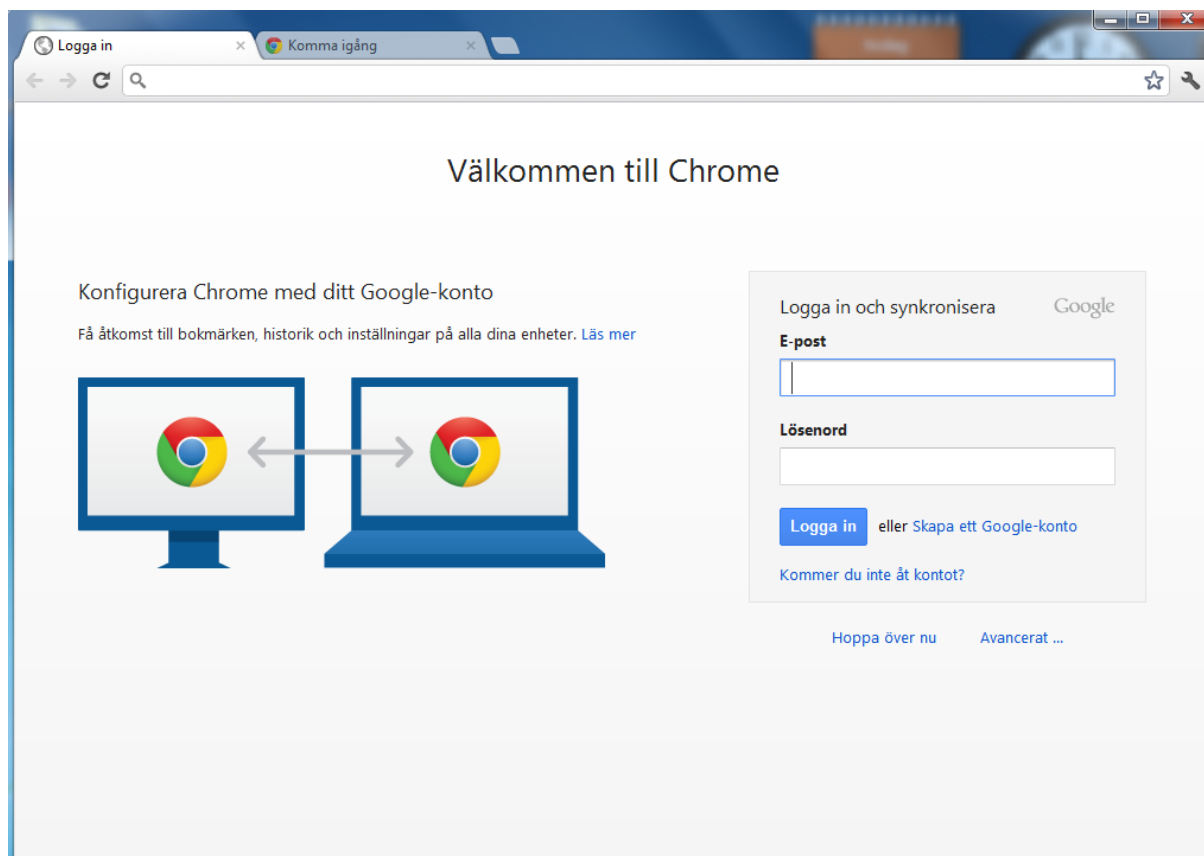
Fördelar med Google Chrome är enligt Pcmag.com förutom snabbheten att den har ledande stöd för HTML 5, hårdvaruacceleration och utmärkt säkerhet. [19]

Fördelar som Pc för Alla nämner är snabbheten och att den följer webbstandarderna. [15]

Som nackdelar med Google Chrome nämner Gizmo's Freeware att det krävs en internetuppkoppling för att installera webbläsaren, det inte så värst anpassningsbara gränssnittet och att det tar mer hårddiskutrymme än Mozilla Firefox och Internet Explorer. [8]

Pcmag.com nämner nackdelar som att sekretessfunktionen att inte spåras då man surfar på nätet är begrävd och avskräckt och att det fortfarande finns några enstaka inkompatibiliteter med webbsidor. [19]

En jämförelse av webbläsare på Centennial Arts nämner nackdelar såsom brist på föräldrakontroller och att hårdvaruaccelerationen inte fungerar på alla grafikkort. [2]



**Figur 5.** Google Chrome 18 i Windows 7 miljö. [egen bild]

## 4 WEBBLÄSARTESTER

Man kan enkelt testa webbläsare genom olika tester som finns på nätet. Självklart säger detta inte allt om hur bra webbläsaren är och kanske inte stämmer till hundra procent men det ger en bra blick i hur det stort sett ligger till. Jag kommer att testa de för tillfället nyaste versionerna av de fem vanligaste webbläsarna som jag tagit upp.

Testerna Peacekeeper, Sunspider, V8 och Acid 3 gjordes den 6 juni 2012 och då var dessa versioner de nyaste av respektive webbläsare: Mozilla Firefox 13.0, Internet Explorer 9.0, Opera 11.64, Safari 5.1.7 och Google Chrome 19.0.

RAM-minne testet gjordes den 22 juli 2012. HTML 5 testet gjordes den 19 januari 2013 och då var de nyaste versionerna av respektive webbläsare dessa: Mozilla Firefox 18.0, Internet Explorer 9.0, Opera 12.12, Safari 5.1.7 och Google Chrome 24.0.

Testerna har körts på en bärbar Packard Bell EasyNote TJ61 dator med 64-bitars operativsystem. Datorn har en AMD Athlon X2 Dual-Core processor på 2.10 GHz, 4 GB RAM-minne och ett ATI RADEON HD 4570 grafikkort på 512 MB. Inga andra flikar/fönster i webbläsaren eller processer i Windows var öppna under testerna.

De tester jag gjorde var Peacekeeper (testar grafiken), Sunspider (testar javascript), V8 (Googles test av javascript), HTML 5 (testar hur väl webbläsarna stöder den kommande HTML 5 standarden) och Acid 3 (testar hur väl webbläsarna följer webbstandarderna) som W3C har tagit fram. [10]

Till sist har jag även ett eget test där jag kontrollerar hur mycket RAM-minne webbläsarna tar under användning.

### 4.1 Peacekeeper

I testet Peacekeeper är det bäst att få så mycket poäng som möjligt.

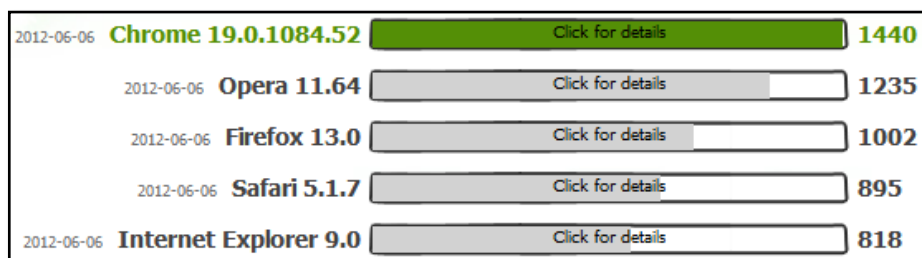
Google Chrome 19.0 fick **1440 poäng** i testet, HTML 5 förmåga **7/7**.

Opera 11.64 fick **1235 poäng** i testet, HTML 5 förmåga **5/7**.

Mozilla Firefox 13.0 fick **1002 poäng** i testet, HTML 5 förmåga **6/7**.

Safari 5.1.7 fick **895 poäng** i testet, HTML 5 förmåga **3/7**.

Internet Explorer 9.0 fick **818 poäng** i testet, HTML 5 förmåga **3/7**.



**Figur 6.** Testresultatet av Peacekeeper. [egen bild]

## 4.2 Sunspider

I testet Sunspider mäts resultaten i tid (ms), där lägre tid är bättre.

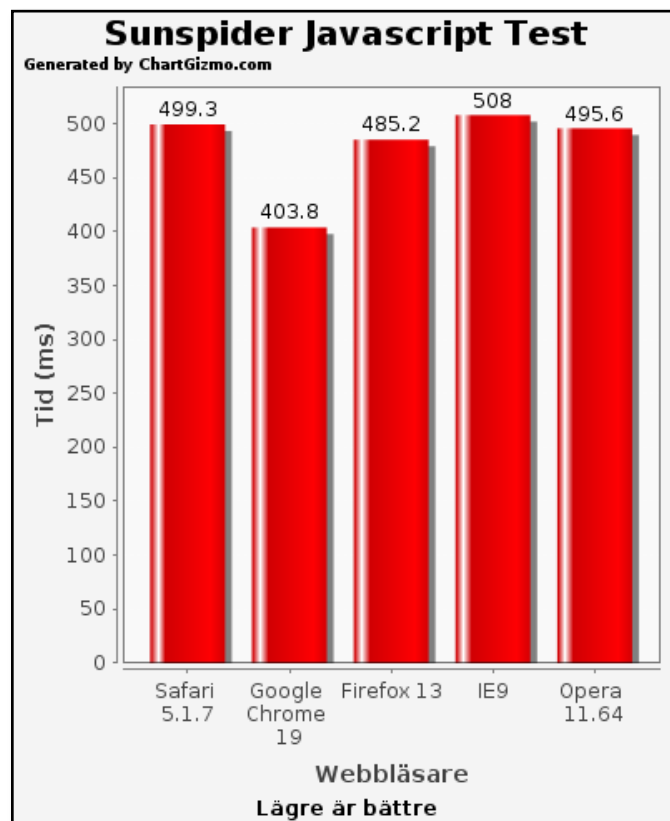
Google Chrome 19.0 fick **403.8 ms**.

Mozilla Firefox 13.0 fick **485.2 ms**.

Opera 11.64 fick **495.6 ms**.

Safari 5.1.7 fick **499.3 ms**.

Internet Explorer 9.0 fick **508.0 ms**.



**Figur 7.** Testresultatet av Sunspider. [egen bild]

### 4.3 V8

I testet V8 är det bäst att få så mycket poäng som möjligt.

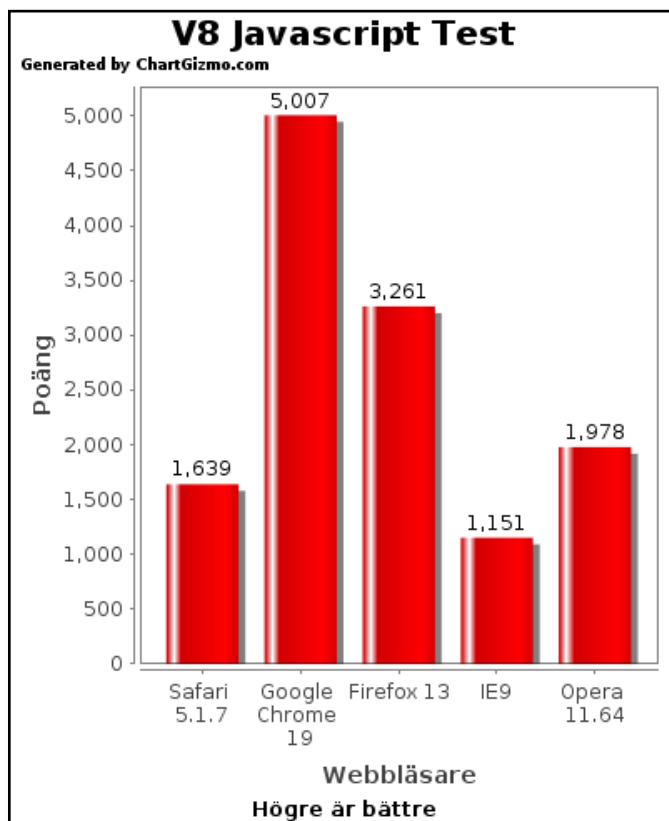
Google Chrome 19.0 fick **5007 poäng**.

Mozilla Firefox 13.0 fick **3261 poäng**.

Opera 11.64 fick **1978 poäng**.

Safari 5.1.7 fick **1639 poäng**.

Internet Explorer 9.0 fick **1151 poäng**.



**Figur 8.** Testresultatet av V8. [egen bild]

#### 4.4 HTML 5

I testet HTML 5 är det bäst att få så mycket poäng som möjligt av maximalt 500 poäng.

Google Chrome 24.0 fick **448 poäng + 13 bonuspoäng.**

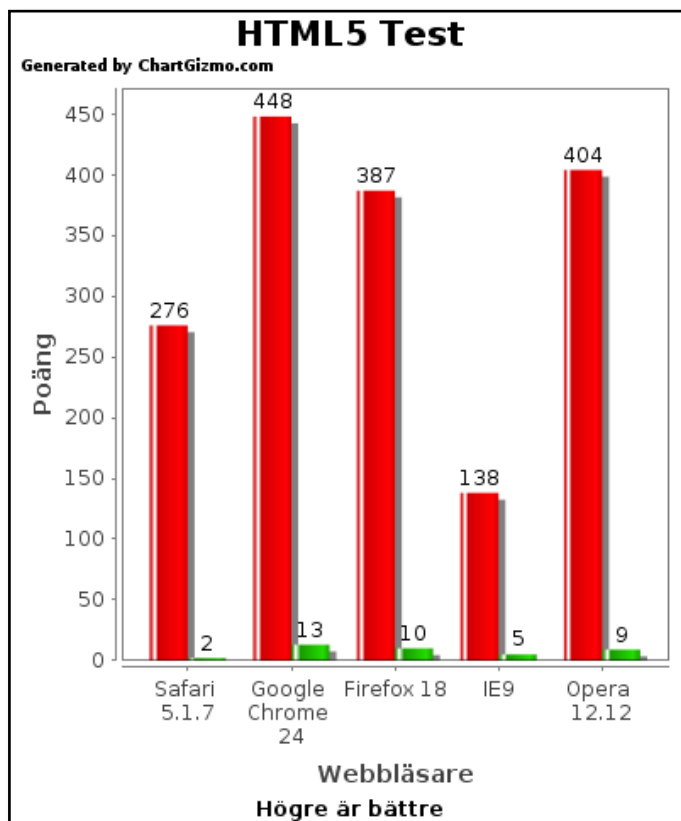
Opera 12.12 fick **404 poäng + 9 bonuspoäng.**

Mozilla Firefox 18.0 fick **387 poäng + 10 bonuspoäng.**

Safari 5.1.7 fick **276 poäng + 2 bonuspoäng.**

Internet Explorer 9.0 fick **138 poäng + 5 bonuspoäng.**

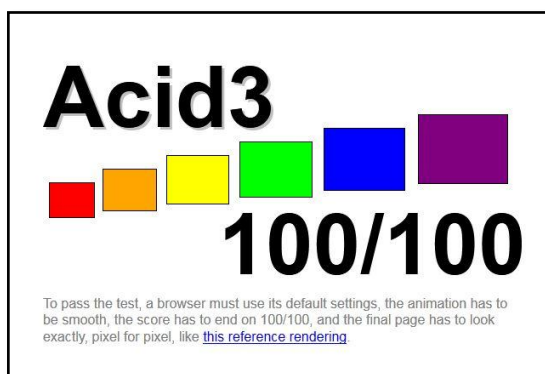




**Figur 9.** Testresultatet av HTML 5. [egen bild]

#### 4.5 Acid 3

Alla fem webbläsare fick **100 av 100** i testet för att de följer webbstandarderna. Lite förvånande att Internet Explorer fick 100 poäng med tanke på att de tidigare inte har följt standarderna fullt ut.



**Figur 10.** Alla fem webbläsare fick fullt godkänt i testet för att de följer webbstandarderna. [egen bild]

#### 4.6 RAM-minne test

Sista testet är ett RAM-minne test där jag testar hur mycket RAM-minne webbläsarna tar då en flik är uppe (bara Google) samt då fem flikar är uppe (Ot.fi, Facebook, YouTube, Last.fm och Verkkokauppa.com). Windows egna aktivitetshanterare används för att övervaka minnesanvändningen.

OBS! Jag räknar med att 100 000 kilobytes (kb) är 100 megabyte (MB). Egentligen är det 97,66 MB men för att göra det lättare att räkna så anger jag det så. Mindre RAM-användning är alltså bättre.

##### Bara en flik öppen

Då endast Google var öppet i samtliga webbläsare blev det ett litet oväntat resultat.

Mozilla Firefox använde allra mest RAM-minne, nästan **129 MB**.

Safari hade en sammanlagd användning på **97,08 MB**.

Google Chrome hade en sammanlagd användning på **75,27 MB**.

Opera hade en sammanlagd användning på **57,70 MB**.

Internet Explorer hade en sammanlagd användning på **30,23 MB**.

Resultatet var lite oväntat. Jag trodde nog att Mozilla Firefox skulle dra mindre än så och att Internet Explorer inte skulle dra minst av alla.

##### Fem flikar öppna

Då fem flikar var öppna i samtliga webbläsare blev resultatet annorlunda.

Safari tycks vara sämst av dessa webbläsare att hålla nere minnesanvändningen då man har flera flikar öppnade samtidigt. Safari använde hela **381,84 MB** sammanlagt.

Google Chrome hade en sammanlagd användning på **290,27 MB**.

Opera använde sammanlagt **272,81 MB**.

Internet Explorer använde sammanlagt **213,37 MB**.

Mozilla Firefox använde sammanlagt **196,26 MB**.

Mozilla Firefox visar sig vara optimerad att hålla låg minnesanvändning då en del flikar är öppna.

## 5 VAD ÄR ENKELHET?

Medan man funderar på att göra en webbläsare enkel kan man fundera vad enkelhet egentligen är för någonting. Enkelhet är någonting som kan variera från person till person väldigt avsevärt. Vissa tycker att det är enkelt på ett sätt och andra på ett annat sätt.

Enligt mig är enkelhet någonting som är lättgenomskådligt och simpelt utan onödiga detaljer. Utan att tidigare ha använt produkten skall de som använder den kunna göra de viktigaste sakerna utan att förvandlas till ett frågetecken. Jag anser också att tydlighet och enkelhet går hand i hand eftersom om man är tydlig i hur någonting fungerar är produkten oftast mer enkel att använda än om produkten inte har tydliga instruktioner.

Enligt Compound Thinking är enkelhet när man vet att en sten till skulle bli för många och en sten mindre skulle bli för få. Men det handlar inte bara om att veta rätta antalet stenar utan vilka stenar som är de rätta och hur man skall ordna dem. [4]



**Figur 11.** Med hjälp av denna bild på stenar förklarar Compound Thinking vad enkelhet är. [4]

Enkelhet uppnås inte bara genom att göra något enklare eller mindre komplex. Ta bort all komplexitet, all svårighet och alla detaljer från vad som helst och det som lämnar kvar är inte enkelt utan bara tråkigt. Å andra sidan omfattar enkelhet de rätta detaljerna, rätt svårighet och rätt komplexitet men eftersom allt hänger ihop på ett bra sätt lämnar det kvar en känsla av klarhet och att allting hör hemma exakt där det är. Enkelhet uppnås när allt betyder något. [4]

Med andra ord uppnås enkelhet med vad du lägger till – klarhet, syfte och avsiktlighet – inte av vad du tar bort. [4]

Ett citat av Albert Einstein om enkelhet:

“Everything must be made as simple as possible. But not simpler.” [9]

## **5.1 Enkelhet ur en webbläsares synvinkel**

Vad är enkelhet då för en webbläsare? Enligt vad Compound Thinking anser om enkelhet borde det alltså vara lagom mycket enkelhet med rätt svårighetsgrad som lämnar kvar en känsla av klarhet.

Om man tänker mer konkret på en webbläsare så tror jag att enkelhet är att knappar, fält och övriga kontroller är på rätt ställen. Om de är på ställen där var folk minst anar att de finns och de känns ologiskt placerade tycker kanske användarna att webbläsaren är raka motsatsen till enkel. Även texter och bilder på kontroller i webbläsaren tror jag att spelar en rätt så stor roll då man tänker på enkelhet. Det gäller att ha logiska namn och beskrivande bilder på kontroller för att man direkt skall förstå vad de gör, eller i alla fall få en inblick i vad man tror de gör.

Jag tror även att om knappar och fält i en webbläsare är lite större än vad de är i de vanligaste webbläsarna på marknaden blir webbläsaren automatiskt lite enklare. Personer med sämre syn ser kanske inte så tydligt vad det står i adressfältet om det är ganska liten fontstorlek på texten eller om en knapp är väldigt liten att klicka på och att då lyckas "träffa" den. Med större kontroller i en webbläsare borde det ju bli lite tydligare att se och förstå saker och ting men då kontrollerna är större blir förstås utrymmet där man surfar lite mindre. Det gäller alltså att hitta en bra balans.

Det gäller också att ha rätt antal funktioner i en webbläsare, d.v.s. att inte ha alltför många så att användarna drunknar i dem men det gäller också att ha de rätta funktionerna. Om nästan alla användare förväntar sig att programmet har en viss

funktion men den inte har den blir användarna högst antagligen besvikna och vill kanske inte använda programmet mera.

En av de viktigaste delarna är att det finns en klarhet. Användarna skall kunna känna att webbläsaren är logiskt strukturerad och de saker som borde finnas på ett visst ställe faktiskt finns där. Det skall kännas naturligt att var och en sak hör hemma exakt var den är placerad och att allt hänger ihop på ett bra sätt.

Ett bra sätt att få människor att veta vad de skall skriva i textrutor är att ha en text synlig då man inte har skrivit in någon text i textrutan som förklarar vad man skall skriva in i den. Exempel på detta är min sökfunktions sökruta där texten "Skriv sökordet i denna ruta" syns om man inte skrivit in någon text i sökrutan och den inte har fokus.

## 6 VAD ÄR ANVÄNDBARHET?

Användbarhet är en viktig punkt när man skapar en webbläsare, eller överhuvudtaget skapar något som skall användas av människor. Detta hänger lite ihop med kapitlet "Vad är enkelhet?" men jag anser ändå att det är lite olika saker.

Med användbarhet menas enligt Usability Net att göra produkter och system lättare att använda och anpassa dem närmare till användarnas behov och krav. Den internationella standarden ISO 9241-11 ger riktlinjer för användbarhet och definierar det som: [27]

*Den grad i vilken specifika användare kan använda en produkt för att uppnå ett specifikt mål på ett ändamålsenligt, effektivt och för användaren tillfredsställande sätt i ett givet sammanhang. [27]*

Användbarhet handlar om:

- Effektivitet - kan användare slutföra uppgifter, uppnå mål med produkten, d.v.s. vad de vill göra.
- Verkningsgrad - hur mycket resursåtgång behöver användarna för att åstadkomma detta? (ofta mätt i tid)
- Tillfredsställelse - vad tycker användarna om produkternas användarvänlighet? [27]

Användbarheten påverkas av:

- Användarna - vilka som använder produkten? T.ex. välutbildade och erfarna användare eller nybörjare?
- Deras mål - vad försöker användarna göra med produkten - stödjer den vad de vill göra med den?
- Användningsområdet - var och hur produkten används? [27]

Användbarhet skall inte förväxlas med funktionalitet eftersom det behandlar bara de funktioner och finesser som produkten har och har ingen inverkan på om

användarna kan använda dem eller inte. Ökad funktionalitet betyder inte förbättrad användbarhet! [27]

Nielsen Norman Group definierar att användbarhet är en kvalitetsegenskap som bedömer hur enkla användargränssnitt är att använda. Ordet användbarhet avser även metoder för att förbättra enkel användning under designprocessen. [13]

Medan Usability Net anser att användbarhet handlar om tre saker definierar

Nielsen Norman Group användbarhet med fem kvalitetskomponenter: [13]

- Lärbarhet (learnability)
- Effektivitet (efficiency)
- Memorerbarhet (memorability)
- Fel/misstag (errors)
- Tillfredsställelse (satisfaction) [13]

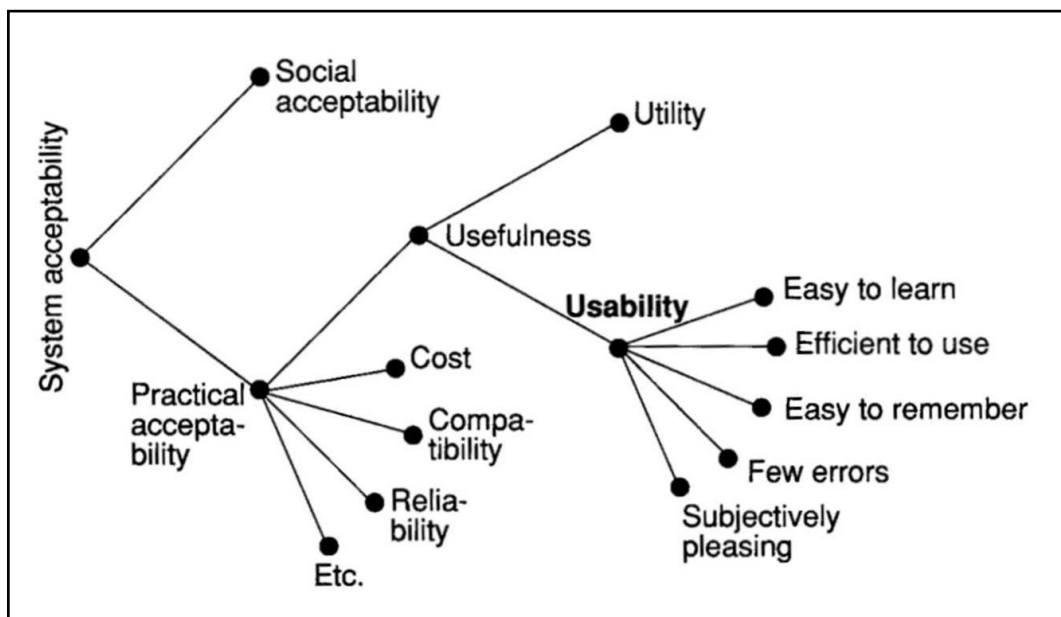
Det finns också andra viktiga kvalitetsegenskaper. En viktig sådan är funktionaliteten, som hänvisar till designens funktionalitet: gör den vad användarna behöver? [13]

Användbarhet och funktionalitet är lika viktiga och tillsammans avgör de huruvida något är nyttigt. Det har mindre betydelse att något är enkelt om det inte är vad du vill ha. Det är inte heller bra om systemet hypotetiskt kan göra vad du vill, men du kan inte få det att hända eftersom användargränssnittet är för svårt. För att studera en designs funktionalitet kan man använda samma metoder som förbättrar användbarheten. [13]

Definitioner över viktiga ord:

- Funktionalitet = om det ger de funktioner du behöver.
- Användbarhet = hur enkla och trevliga dessa funktioner är att använda.
- Nyttä = användbarhet + funktionalitet. [13]





**Figur 12.** Hur användbarhet (usability) relaterar till övriga begrepp enligt Nielsen. Nyttan (usefulness) omfattar funktionalitet (utility) och användbarhet (usability). [5]

### 6.1 Varför är användbarhet viktig?

På nätet är användbarhet en nödvändig förutsättning för att överleva. Om en webbplats är svår att använda lämnar man den. Om hemsidan misslyckas med att tydligt ange vad företaget erbjuder och vad användarna kan göra på webbplatsen lämnar användarna den. Om användarna går vilse på en webbplats lämnar de den. Om informationen på en webbplats är svår att läsa eller inte besvarar användarnas viktigaste frågor så kommer de också att lämna den. [13]

Det finns inget sådant som att en användare läser en manual för en webbplats eller annars spenderar mycket tid på att försöka lista ut ett gränssnitt. Det finns massor av andra webbplatser som är tillgängliga, att lämna webbsidan är det första användaren gör när den stöter på en svårighet. [13]

Den första lagen av elektronisk handel är att om användarna inte kan hitta produkten så kan de inte köpa den heller. [13]

## 6.2 Hur förbättra användbarheten?

Det finns många metoder för att studera användbarhet, men de mest grundläggande och nyttiga är användartester, vilka har tre delar: [13]

- Få tag på några representativa användare, t.ex. kunder för en webbutik eller anställda för ett intranät (i det senare fallet bör de arbeta utanför din avdelning).
- Be användare att utföra representativa uppgifter med designen.
- Observera vad användarna gör, var de lyckas och där de har problem med användargränssnittet. Håll tyst och låt användarna sköta snacket. [13]

Det är viktigt att testa användare individuellt och låta dem lösa eventuella problem på egen hand. Om du hjälper dem eller riktar deras uppmärksamhet till en viss del av skärmen har du påverkat testresultaten. [13]

För att identifiera en designs mest betydelsefulla användbarhetsproblem räcker det vanligtvis med att testa fem användare. I stället för att köra en stor och dyr undersökning är det bättre att köra många mindre tester och se över designen mellan var och en så att du kan fixa användbarhetens brister medan du identifierar dem. Iterativ design är det bästa sättet att höja kvaliteten på användarupplevelsen. Ju fler versioner och idéer på gränssnitt du testat med användarna, desto bättre. [13]

Användartester skiljer sig från fokusgrupper, som är ett dåligt sätt att utvärdera användbarhet på design. Fokusgrupper har en viss betydelse i marknadsundersökningar, men för att utvärdera samverkande design måste du noga iaktta då enskilda användare utför uppgifter med användargränssnittet. Att lyssna på vad folk säger är missvisande: man måste i stället titta på vad de verkligen gör. [13]

## 6.3 Användbarhet ur en webbläsares synvinkel

Jag hoppas att webbläsaren uppfyller målgruppens behov och mål. Webbläsarens främsta mål torde vara att kunna surfa på nätet med webbläsaren. Andra mål kan

vara att det finns funktioner som användarna kräver att det ska finnas och att dessa fungerar som de borde. Det ska vara lätt att hitta det de söker i webbläsaren och inte ta lång tid att söka efter var någonting finns. Webbläsaren borde vara användarvänlig för att användarna skall kunna känna sig tillfredsställda med den och det hoppas jag att den är. Jag har försökt att göra saker och ting tydliga i webbläsaren för att göra den lättanvändlig. Då webbläsaren främst är till för äldre personer har jag försökt att göra användbarheten därefter.

## **7 MÅLGRUPPEN OCH DERAS ÖNSKEMÅL**

Som målgrupp för min webbläsare har jag valt främst äldre personer som inte behöver alla onödiga funktioner som finns i webbläsarna på marknaden nuförtiden. Åldersgruppen är därmed tänkt att ligga på 50 år och uppåt. Som mål är att webbläsaren skall vara tydlig och enkel att använda, med endast de viktigaste funktionerna.

### **7.1 Undersökning**

För att fråga min målgrupp om webbläsaren valde jag att göra en undersökning i E-lomake och sedan publicera den på Facebook så att personer kunde svara på den och dela med sig vidare av undersökningen. Jag tycker det är ganska aktuellt att göra undersökningen på nätet istället för en undersökning på papper eller via intervju.

Det var svårt att få folk i åldersgruppen att svara på undersökningen. Kanske var det för att folk var för stolta för att göra det eller också var det för att jag inte känner tillräckligt många inom åldersgruppen. Men efter en sista satsning i form av ett massutskick via e-post så fick jag ändå ihop tillräckligt många svar.

Undersökningen bestod av fyra frågor men vissa av frågorna var egentligen som flera påståenden.

### **7.2 Resultatet av undersökningen**

Undersökningen var öppen att besvara under tiden 02.08.2012 – 14.10.2012. Sammanlagt svarade elva personer på undersökningen.

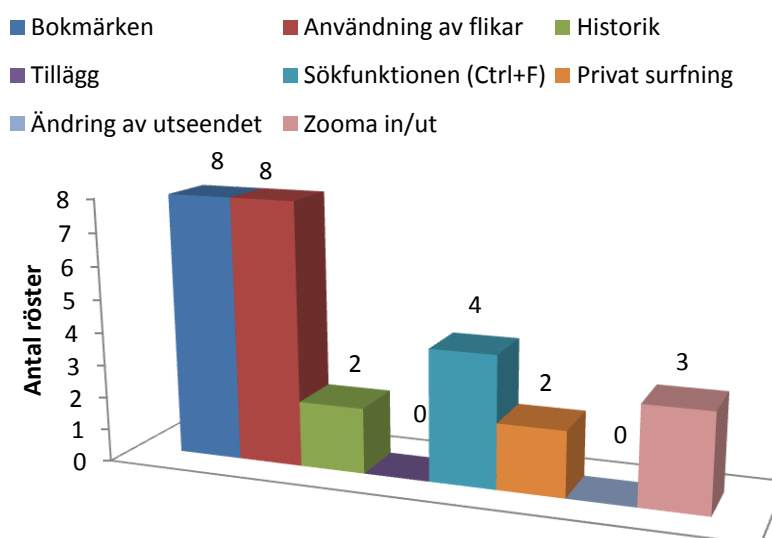
Den första frågan löd "vilka funktioner är för dig mest nödvändiga i en webbläsare?". Minst ett alternativ och högst tre alternativ fick man välja per person bland de åtta möjliga alternativ som fanns tillgängliga.

De funktioner som fick flest röster var bokmärken samt användning av flikar, som båda fick åtta röster vardera. Sökfunktionen och zooma in/ut funktionen fick fyra

respektive tre röster. Två röster fick historik och privat surfning medans tillägg och ändring av utseendet inte fick några röster alls.

Jag tyckte det var lite överraskande att bokmärken fick så många röster som det fick. Hade trott att zooma in/ut skulle fått mera röster än det fick.

### Vilka funktioner av dessa är för dig mest nödvändiga i en webbläsare?



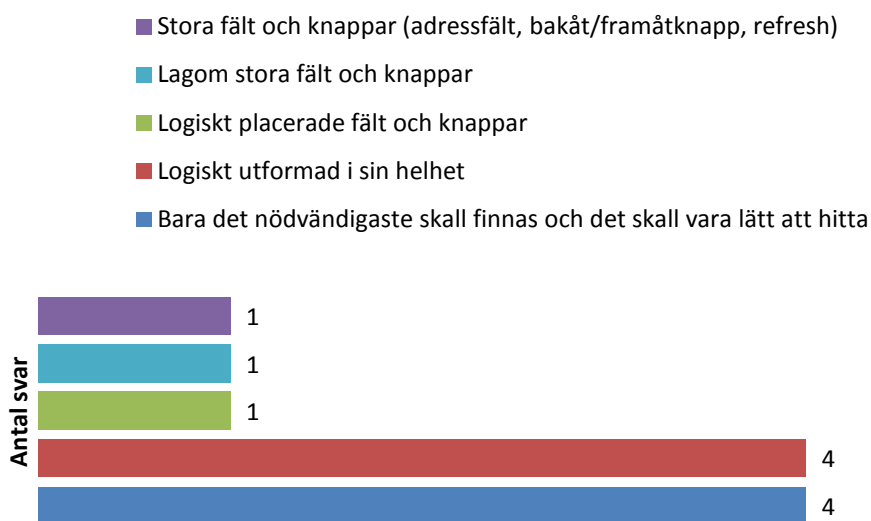
**Figur 13.** Diagram över första frågan. [egen bild]

Den andra frågan löd "hur tycker du att en enkel webbläsare skall se ut?". Man kunde endast välja ett alternativ av de fem alternativ som fanns att välja mellan.

Fyra personer tyckte att bara det nödvändigaste skall finnas och att det ska vara lätt att hitta. Likaså svarade fyra personer att den ska vara logiskt utformad i sin helhet. Stora fält och knappar, lagom stora fält och knappar samt logiskt placerade fält och knappar fick vardera ett svar.

Inga direkta överraskningar i svaren på frågan, trodde möjligtvis att flera skulle ha svarat stora fält och knappar.

## Hur tycker du att en enkel webbläsare skall se ut?



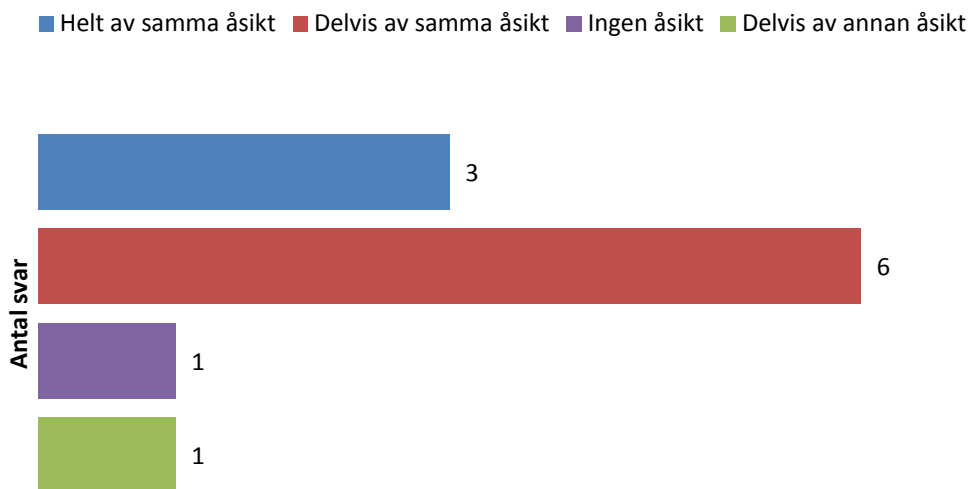
**Figur 14.** Diagram över andra frågan. [egen bild]

Tredje frågan bestod av sex påståenden där man kunde välja vad man tyckte från skalan 1-5, där 1 = helt av annan åsikt, 2 = delvis av annan åsikt, 3 = ingen åsikt, 4 = delvis av samma åsikt och 5 = helt av samma åsikt.

Första påståendet löd "en enkel webbläsare skall bara innehålla det mest nödvändiga". Sex av elva personer var delvis av samma åsikt medan tre personer var helt av samma åsikt. En person var delvis av annan åsikt medan en person inte hade någon åsikt alls.

Svaren var inte så värst överraskande på detta påstående.

## En enkel webbläsare skall bara innehålla det mest nödvändiga

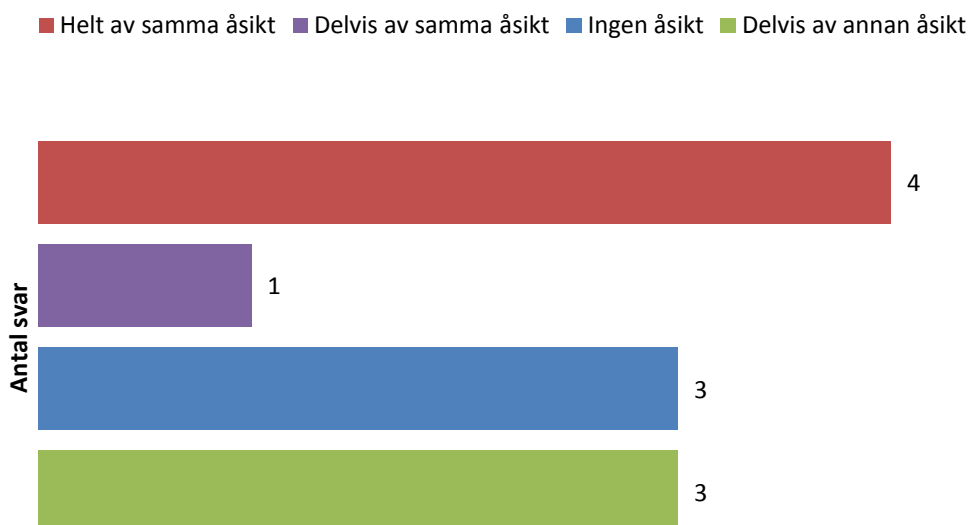


**Figur 15.** Diagram över tredje frågans första påstående. [egen bild]

Andra påståendet löd "sökruatan skall vara till höger om adressfältet". På detta påstående är de som röstat oeniga. Fyra personer var helt av samma åsikt medan tre personer var delvis av annan åsikt och tre personer hade ingen åsikt. En person var delvis av samma åsikt.

Hade väntat mig att flera skulle varit av samma åsikt och att tre personer valde att inte ha någon åsikt på frågan var lite intressant.

## Sökrutan skall vara till höger om adressfältet

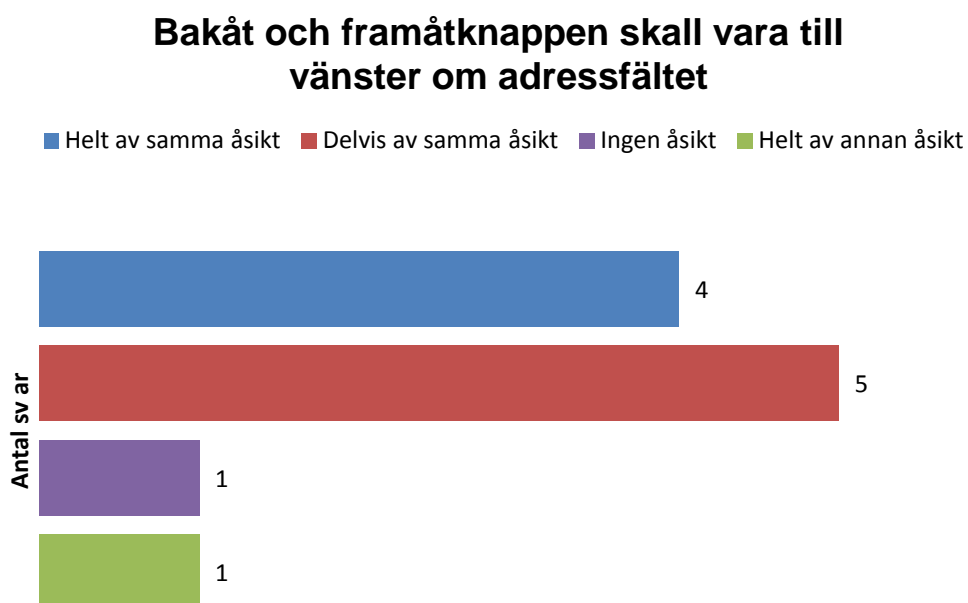


**Figur 16.** Diagram över tredje frågans andra påstående. [egen bild]

Tredje påståendet löd "bakåt och framåtknappen skall vara till vänster om adressfältet". Nästan alla tyckte att dessa knappar skall vara till vänster om adressfältet, kanske inte så konstigt för i alla webbläsare jag sett har det varit på vänster sida om adressfältet. Fem personer var delvis av samma åsikt och fyra personer var helt av samma åsikt. En person var av annan åsikt och tyckte kanske att de borde varit på höger sida om adressfältet medans en person inte hade någon åsikt alls.

Inga överraskningar i svaren, antog att de flesta ville ha dem på vänstra sidan om adressfältet.

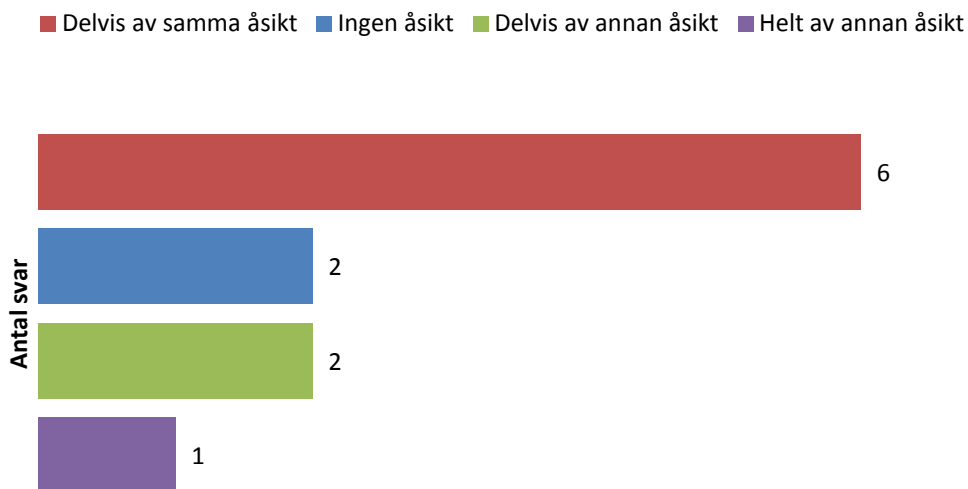




**Figur 17.** Diagram över tredje frågans tredje påstående. [egen bild]

Fjärde påståendet löd "Uppdatera sidan knappen skall vara till höger om adressfältet". Hela sex personer var delvis av samma åsikt medans två personer var delvis av annan åsikt. Två personer hade ingen åsikt i frågan och en person var helt av annan åsikt och tyckte att den kunde vara på något annat ställe.

### Uppdatera sidan knappen skall vara till höger om adressfältet

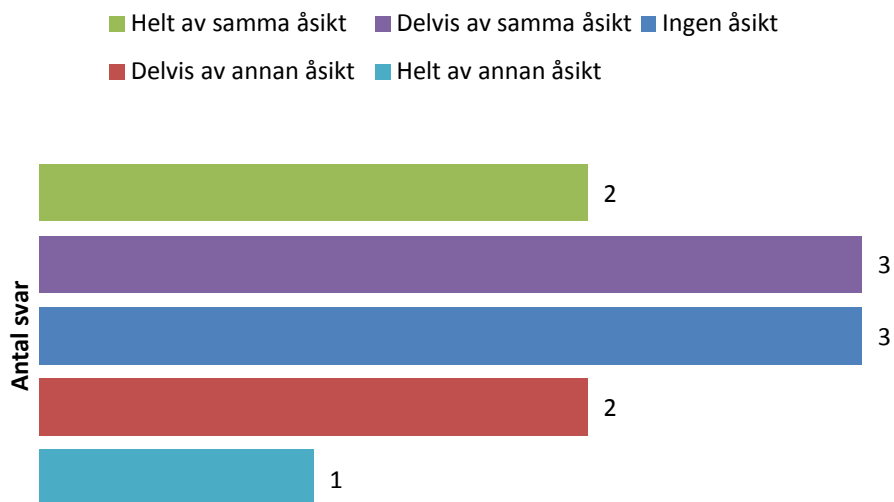


**Figur 18.** Diagram över tredje frågans fjärde påstående. [egen bild]

Femte påståendet löd "en meny för inställningar skall finnas längst upp till vänster". På detta påstående var åsikterna ganska utspridda. Tre personer var delvis av samma åsikt och lika många hade ingen åsikt. Två personer var helt av samma åsikt och lika många var delvis av annan åsikt. En person var helt av annan åsikt.

Frågan var dåligt formulerad då jag egentligen menade huvudmenyn. Tyckte inte resultatet var så överraskande då den skulle egentligen passa på båda sidorna.

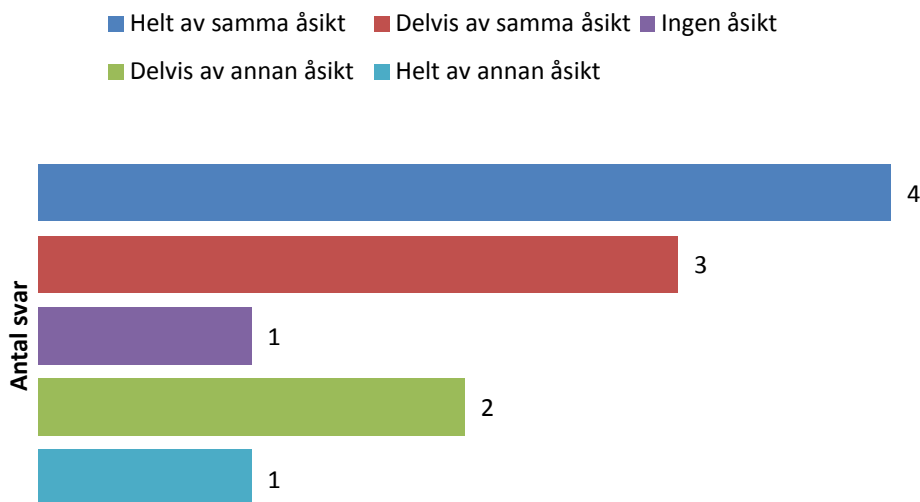
## En meny för inställningar skall finnas längst upp till vänster



**Figur 19.** Diagram över tredje frågans femte påstående. [egen bild]

Sjätte och sista påståendet löd "en knapp som tar dig tillbaka till startsidan skall vara till höger om sökfältet". Här var de som röstat eniga. Fyra var helt av samma åsikt och tre var delvis av samma åsikt. Två personer var delvis av annan åsikt, medans helt av annan åsikt och ingen åsikt fick en röst var.

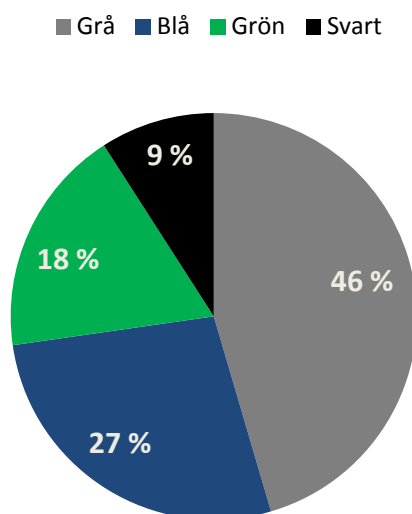
### En knapp som tar dig tillbaka till startsidan skall vara till höger om sökfältet



**Figur 20.** Diagram över tredje frågans sista påstående. [egen bild]

Sista frågan löd "vilken bakgrundsfärg tycker du webbläsaren skall ha?" Mest röster fick grå som fem av elva personer ansåg att skulle vara bäst som bakgrundsfärg. Av de övriga alternativen fick blå tre röster, grön två röster och svart en röst. Röd, gul samt orange fick inga röster alls.

## Vilken bakgrundsfärg tycker du webbläsaren skall ha?



**Figur 21.** Diagram över fjärde och sista frågan. [egen bild]

### 7.3 Tillämpning av undersökningens resultat

Efter att jag sammanställt resultaten av varje fråga i undersökningen har jag försökt tillämpa de svar som fick mest röster på varje fråga i min webbläsare.

På den första frågan blev det bokmärken och användning av flikar som fick mest röster. Tredje och fjärde mest röster fick sökfunktionen samt zooma in/ut. Alla dessa funktioner har jag därför tillämpat i min webbläsare.

På den andra frågan var de som svarat inte eniga. Lika många tyckte att bara det nödvändigaste skall finnas och att det skall vara lätt att hitta som att den skall vara logisk utformad i sin helhet. Jag har därför bara försökt att lägga in de mest nödvändiga funktionerna i webbläsaren och skapat ett verktygsfält där de flesta funktioner finns. Vertygsfältet bidrar till att det är lätt att hitta funktionerna och alla funktioner som finns där har både bild och text. Jag har även försökt göra webbläsaren logiskt uppbyggd med knappar och fält på logiskt placerade ställen.

Nästan alla tyckte mer eller mindre att en enkel webbläsare bara skall innehålla det mest nödvändiga på tredje frågans första påstående. Därmed behövde inga

vidare åtgärder göras utan jag tänkte bara inkludera det mest nödvändiga i denna webbläsare.

Det andra påståendet på tredje frågan var om sökrutan skulle vara till höger om adressfältet. Fyra personer var helt av samma åsikt och det var därmed det populäraste svaret vilket gjorde att jag placerade sökrutan till höger om adressfältet i min webbläsare.

Att bakåt- och framåtknappen skulle vara till vänster om adressfältet var det ingen tvekan om i svaren. Därför är dessa placerade allra längst till vänster i webbläsarens navigeringsfält.

Det fjärde påståendet på tredje frågan var om uppdatera knappen skulle vara till höger om adressfältet. De flesta tyckte att den skulle vara placerad där och därmed placerade jag den till höger om adressfältet, men till vänster om sökrutan.

På femte påståendet på tredje frågan var folket lite oense. Frågan var om en meny för inställningar skulle finnas längst upp till vänster. Denna fråga var dåligt formulerad för egentligen menade jag själva huvudmenyn. Det blev två alternativ som fick lika många svar och det var delvis av samma åsikt och ingen åsikt. Eftersom det lutade mera åt samma åsikt blev det att huvudmenyn placerades på vänster sida.

Sista påståendet på den tredje frågan var om en knapp som tar dig tillbaka till startsidan skulle finnas till höger om sökfältet. Här blev svaret jakande och jag placerade den allra längst till höger i navigeringsfältet.

Sista frågan på undersökningen handlade om vad webbläsarens bakgrundsfärg skulle vara. Här svarade nästan hälften av de svarande grå och därmed blev det grå som fick mest svar. Jag har därmed en ganska ljusgrå bakgrundsfärg på webbläsaren och jag tycker det passar ganska bra. Mörkgrå passade inte alls så jag höll mig till lite ljusare toner.

Jag hoppas att målgruppens önskemål blev uppfyllda så långt som det gick.

## 8 QT

### 8.1 Vad är Qt?

Qt är ett komplett utvecklingsramverk med verktyg som utformats för att effektivisera skapandet av applikationer och användargränssnitt för stationära, inbäddade och mobila plattformar. Med Qt kan kodare skapa innovativa applikationer och touch-baserade användargränssnitt en gång och sedan distribuera över alla plattformar utan att skriva om koden. [23]

### 8.2 Produkter

Qt SDK kombinerar ramverk med verktyg för att effektivisera skapandet av applikationer för mobila enheter samt dessutom till stationära plattformar (som Microsoft Windows, Mac OS X och Linux) och inbäddade enheter. Qt SDK finns i två olika varianter: Qt kommersiell SDK och Qt SDK för Nokia-enheter. [25]

Qt Creator är en plattformsoberoende integrerad utvecklingsmiljö (IDE). Den är grunden för Qt SDK. [24]

Den ger:

- Stöd för stationära och mobila anordningar
- C++ och JavaScript editor
- Inbyggd användargränssnitts-designer
- Projekt och bygghanteringsverktyg
- gdb och CDB felsökare
- Simulator för mobila användargränssnitt
- Stöd för versionshantering [24]

Förutom Qt Creator finns det även andra verktyg såsom Qt Quick Designer (en visuell redigerare för QML-filer), Qt Designer (används för att bygga widgets och dialoger), Qt Linguist (används för att översätta applikationer) och Qt Assistant (dokumentation över olika Qt-program). [24]

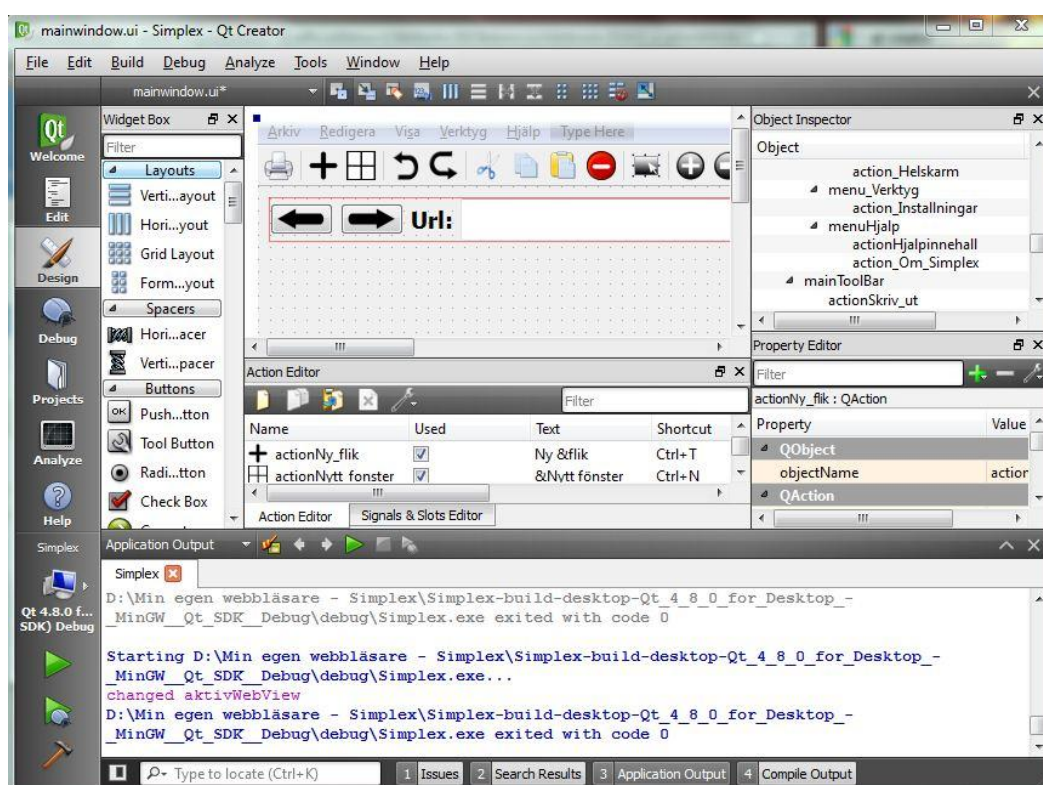
### **8.3 Vem ligger bakom Qt?**

Ramverket Qt skapades år 1994 av Trolltech men de sålde det vidare år 2008 till Nokia. I mars 2011 köpte Digia Plc. licens- och supporträttigheter av Qt från Nokia. I september 2012 köpte Digia hela Qt-verksamheten från Nokia. [22]



## 9 PROGRAMMERING AV WEBBLÄSAREN

För att programmera webbläsaren använde jag mig av Qt Creator samt mingw för att kompilera. Eftersom jag inte kunde någonting om Qt från tidigare kollade jag på videor på YouTube för att lära mig grunderna i programmet samt läste en del i hjälpmanualer och på forum om Qt. Jag gjorde givetvis några exempel som jag hittade på Internet för att vidare bekanta mig med programmets gränssnitt och funktioner.

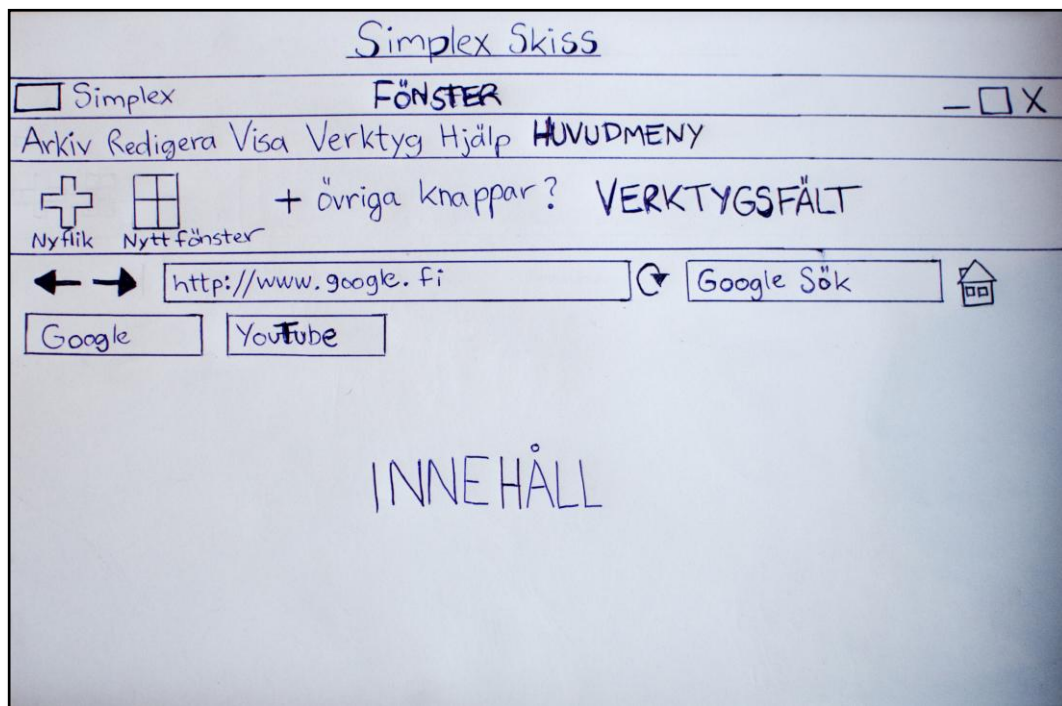


Figur 22. Designvyn i Qt Creator. [egen bild]

### 9.1 Planering

Förrän jag började med programmeringen ritade jag en skiss på ett papper på hur webbläsaren ungefär skulle se ut. Nu i efterhand känner man igen sig på vissa punkter men i verkligheten blev saker och ting lite annorlunda än på pappret. På pappret hade jag nämligen inte utfunderat vissa knappar och funktioner som jag senare kom på att skulle vara bra att ha med i själva programmet.

Mer planering än på pappret hade jag egentligen inte utan jag hade en bild i huvudet på hur webbläsaren skulle se ut och vilka funktioner den skulle ha. Jag hoppas att jag lyckades att (med hjälp av målgruppens önskemål och egna funderingar kring hur man skulle göra) göra den lättanvänd.



**Figur 23.** Min skiss över webbläsaren. [egen bild]

## 9.2 Kodning

Vad skall jag börja med? Hur skall jag gå till väga? Frågorna var många i början av programmeringsskedet. Jag bestämde mig helt enkelt för att först följa en kort video på YouTube om hur man gör en enkel webbläsare i Qt Creator. Notera att denna webbläsare inte innehöll mycket alls, utan bara grundpelarna för en webbläsare. Den innehöll nämligen endast själva webbläsarvyn, adressfältet samt en bakåt- och framåtknapp.

Det svåra men roliga och samtidigt utmanande var att fortsätta härifrån för att utveckla webbläsaren. Jag tycker det är viktigt att ha en bra struktur så att man i framtiden också vet vad man har gjort i koden. Därför har jag skrivit kommentarer

på de flesta ställen i koden och namngett kontrollerna med logiska namn istället för t.ex. "pushButton1" och "pushButton2". Jag vet att struktur hjälper mycket och sparar tid i slutändan trots att det tar ett tag att skriva kommentarer till all kod och att namnge alla kontroller.

markeraallaButton		QPushButton
sokEdit		QLineEdit
sokSpacer		Spacer
sokfellaabel		QLabel
sokforegaendeButton		QPushButton
soklabel		QLabel
soknastaButton		QPushButton

**Figur 24.** Det är bra att namnge kontrollerna med logiska namn så att man lätt vet skillnad på dem. [egen bild]

### 9.2.1 Huvudmeny och verktygsfältet

En av de första sakerna jag gjorde efter att ha haft grundpelarna var att skapa en huvudmeny. Jag gick noggrant igenom vad de vanligaste webbläsarna har i sina huvudmenyer för att skala av och välja ut de mest nödvändiga av dessa funktioner. Många funktioner som t.ex. privat surfning, tillägg och massor av inställningar är onödiga för denna webbläsares syfte.

Då jag utfunderat vilka funktioner jag ville ha i menyerna och beaktat vilka funktioner som var viktigaste för min målgrupp (fråga ett i min undersökning) var nästa steg att skapa ikoner för en del av alternativen/funktionerna från huvudmenyn. Uttrycket "en bild säger mer än 1000 ord" stämmer riktigt bra. Speciellt då ändamålen är att skapa en enkel och lättanvänd webbläsare hjälper bilder till en hel del.

För att skapa ikonerna använde jag mig av bildredigeringsprogrammet Adobe Photoshop CS6. I vissa av fallen använde jag mig av färdiga figurer och i vissa av fallen gjorde jag dem helt från början. Några ikoner som var svåra att göra på egen hand hämtade jag från nätet. Nästan genast märkte jag att Qt Creator har en egenskap som gör att då man laddat upp en bild till ett alternativ i huvudmenyn

kan man också dra detta alternativ till ett verktygsfält (som i detta fall finns nedanför huvudmenyn). Givetvis kan man också göra detta fast man inte laddat upp någon bild men i ett verktygsfält hör det ju till att man har bilder på i alla fall de flesta av knapparna. Verktygsfältet bidrar därför till att man kan göra de flesta av mina snabbkommandon/funktioner utan att gå in i huvudmenyn.



**Figur 25.** Huvudmenyn och verktygsfältet. [egen bild]

### 9.2.2 Navigeringsfältet

Efter att huvudmenyns ikoner och de alternativ som skall finnas i huvudmenyn var klara gjorde jag ikoner till bakåt och framåtknappen. Jag konstaterade att webbläsaren behövde en uppdateringsknapp, stoppknapp och hemknapp så dessa skapades omgående. I ett litet senare skede kom jag även på att det behövdes en gå-till-webbsida knapp. Ikoner gjordes till alla dessa knappar och de gjordes även denna gång i Adobe Photoshop CS6.

Bakåt- och framåtknappen har jag programmerat så att om det finns någonting i historiken för bakåt så kan man klicka på bakåtknappen och om det finns någonting i historik för framåt kan man klicka på framåtknappen.

```

void MainWindow::on_bakotButton_clicked()
{
    // Går bakåt ett steg till webbsidan man var på tidigare
    aktivWebView->back();
}

void MainWindow::on_framotButton_clicked()
{
    // Går framåt till sidan man var på förrän man klickade på bakåt
    aktivWebView->forward();
}

```

**Figur 26.** Koden för knapparna bakåt och framåt då man klickat på dem. [egen bild]

Uppdateringsknappens kod är rätt så simpel. Namnet på knappen är `uppdateraButton` och det högst upp säger att om man klickar på denna knapp händer detta. Inom `{ }` är koden där man gör någonting. Det som börjar med `"/` är en kommentar. `aktivWebView` är kontrollen som visar själva webbläsaren, den gör en åtgärd *Reload* när man klickar på knappen. Uppdateringsknappen är programmerad så att man kan klicka på den endast då webbsidan har laddats färdigt.

```

void MainWindow::on_uppdateraButton_clicked()
{
    // Uppdaterar/laddar om sidan i aktivWebView
    aktivWebView->triggerPageAction(QWebPage::Reload);
}

```

**Figur 27.** Koden för uppdateringsknappen. Då man klickar på knappen uppdaterar den webbsidan i webbläsaren (`aktivWebView`). [egen bild]

Stoppknappens kod är ungefär som uppdateringsknappens kod bara att istället för åtgärden *Reload* används åtgärden *Stop*. Stoppknappen kan man endast klicka på då en webbsida håller på att laddas.

Hemknappen laddar in webbsidan man har angett i inställningar för startsidan (under Verktyg->Inställningar i huvudmenyn).

Gå-till-webbsida knappen har samma funktion som då man klickar på Enter-tangenten i adressfältet då man skall gå till en webbsida. Den angivna webbsidan börjar att laddas och om det lyckas visas webbsidan. Gå-till-webbsida knappen kan man endast klicka på ifall man ändrar på texten i adressfältet. Annars skulle den ha i princip samma funktion som uppdateringsknappen och det är ju inte meningen.

```
aktivWebView->load(QUrl(ui->adressEdit->text()));
```

**Figur 28.** Koden som används då man klickar på gå-till-webbsida knappen eller genom Enter-tangenten. Texten man skrivit in i adressfältet (adressEdit) laddas in i webbläsaren (aktivWebView). [egen bild]

Som sista sak i navigeringsfältet skapades en sökruta och en sökknapp med vilken man kan söka på Google. Först tänkte jag göra så att föreslagna sökord skulle dyka upp då man började skriva in något i sökrutan och att man kunde välja mellan flera sökmotorer (precis som i exempelvis Mozilla Firefox). Det blev dock lite för avancerat för mig så istället gjorde jag en enkel sökruta där det görs en sökning på Google över texten man skrivit in och inga föreslagna sökord ges. Man kan antingen klicka på Enter-tangenten då man har muspekaren i sökrutan eller genom att klicka på sökknappen för att göra en sökning. Jag kom fram till att de flesta använder sig av Google som sökmotor och det räcker gott och väl till.

```
QString url = QString(GSEARCH_URL).arg(ui->googleEdit->text());
if (ui->googleEdit->text().isEmpty()) aktivWebView->load(QUrl("http://www.google.fi"));
else
aktivWebView->load(QUrl(url));
```

**Figur 29.** Koden som används för att göra en sökning på Google över det man skrivit in i sökrutan. Om det inte finns någon text i sökrutan då man försöker söka visas Google's startsida. [egen bild]

Adressfältet i webbläsaren kan användas både som sökfält och adressfält. Om man skriver in ett ord i adressfältet som inte innehåller en punkt (".") söker man efter ordet på Google. Om ordet innehåller en punkt räknas det som att ordet egentligen är en webbadress och därför försöker webbsidan nås. Exempel på detta är att "Google" räknas som ett sökord medans "Google.fi" räknas som en webbadress.

### 9.2.3 Huvudmenyns funktioner

Först ut av funktionerna i huvudmenyn som jag programmerade var *kopiera*, *klipp ut* och *klistra in*. Jag ville att kopiera och klipp ut endast skulle gå att använda sig av då någon text var markerad. Klistra in skulle aktiveras ifall man kopierat eller klippt ut någon text någonstans ifrån (med någonstans menar jag alla applikationer och inte bara från denna applikation). Kopiera använder sig av åtgärden *copy* för att kopiera texten.

```
ui->actionKopiera->setEnabled(ui->adressEdit->hasSelectedText());
```

**Figur 30.** Koden för att aktivera kopiera endast då text har markerats (i adressEdit). [egen bild]

```
ui->adressEdit->copy();
```

**Figur 31.** Koden för att kopiera texten (från adressEdit). [egen bild]

På liknande vis som kopiera gjordes klipp ut, bara att istället för åtgärden *copy* användes åtgärden *cut*. Klistra in använder sig av åtgärden *paste* för att klistra in text och för att aktivera klistra in endast då man kopierat/klippt ut någon text används *clipboard*. Clipboard sparar texten då man i vilket program som helst kopierar/klipper ut någon text. Jag aktiverar klistra in om det finns text i clipboarden.

```
QClipboard *clipboard = QApplication::clipboard();
ui->actionKlistra_in->setEnabled(clipboard->mimeTypeData()->hasText());
```

**Figur 32.** Koden för att deklarera clipboard och aktivera klistra in om det finns text i clipboarden. [egen bild]

De funktioner som jag programmerade näst var *ta bort* och *markera allt*. Ta bort programmerades på samma vis som kopiera och klipp ut på det sättet att den aktiverades bara ifall text var markerad. Då man klickar på ta bort körs åtgärden *del* som tar bort texten som man har markerat.

```
ui->addressEdit->del();
```

**Figur 33.** Koden för att ta bort texten med hjälp av åtgärden *del* (i *adressEdit*).

[egen bild]

*Markera allt* är en funktion som hela tiden är aktiverad. Då man klickar på markera allt kör den åtgärden *SelectAll* som markerar all text vid den textruta som för närvarande har fokus (alltså den textruta som muspekaren befinner sig på).

*Ångra* och *gör om* funktioner var något som jag märkte att jag ville ha med efter ett tag. Ångra kör *undo* som ångrar det som finns i en textruta. Om man har ändrat eller skrivit in text i en textruta kan man ångra det man skrivit in med hjälp av ångra.

```
ui->addressEdit->undo();
```

**Figur 34.** Koden för att ångra. [egen bild]

Gör om är motsatsen till ångra, om man t.ex. skrivit in en bokstav i en textruta och sedan klickar på ångra tas bokstaven bort från textrutan och gör om aktiveras. Då kan man klicka på gör om och bokstaven kommer tillbaka igen. Med hjälp av gör om kan man alltså göra om saker man ånkrat. Åtgärden som används för att göra om är *redo*.

En annan funktion jag kom på att jag ville ha med i webbläsaren var att kunna visa och gömma verktygsfältet och navigeringsfältet. Detta gjordes enkelt genom att ha en kryssbox där man kunde kryssa i om det skulle visas och kryssa ur om det skulle gömmas. Att gömma fälten sköttes med funktionen *hide* och för att visa fälten igen användes funktionen *show*.



En funktion för att visa webbläsaren i helskärm var naturligtvis också nödvändig att ha med eftersom den möjligheten finns i de vanligaste webbläsarna och den används rätt så flitigt av en del personer. För att åstadkomma att visa webbläsaren i helskärm gömde jag först alla kontroller och fält när man klickat på knappen helskärm och sedan visade jag fönstret i helskärm. Då man klickat ur knappen helskärm visades kontrollerna och fälten igen och fönstret var som vanligt.

En funktion som jag absolut ville ha med var *zooma in/ut* funktionen. Denna funktion funderade jag ett bra tag på hur jag skulle fullända. Alternativ som jag funderade på var bl.a. att man kunde dra zoomen med hjälp av ett reglage (slider) eller en enkel zoom med knappar. Till slut blev det en väldigt enkel funktion med zooma in- och ut knapp samt en återställningsknapp. Zooma in knappen zoomar in 10 % varje gång man klickar på den medan zooma ut knappen zoomar ut 10 % varje gång man klickar på den. Återställningsknappen zoomar tillbaka till det ursprungliga läget, nämligen 100 %.

```
aktivWebView->setZoomFactor(aktivWebView->zoomFactor()+0.1);
```

**Figur 35.** Kodan för att zooma in. [egen bild]

```
aktivWebView->setZoomFactor(1.0);
```

**Figur 36.** Kodan för att återställa zoomningen till 100 %. [egen bild]

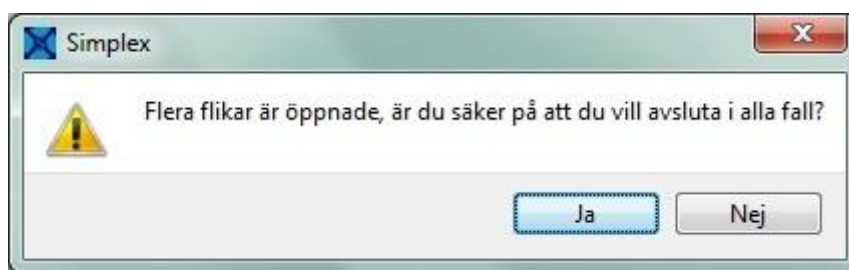
Vissa funktioner som jag trodde skulle vara svårare att göra än andra var i själva verket enklare. De funktioner jag tänker på i detta fall är *förhandsgranska* och *skriv ut*. Innan jag kollade upp hur man programmerar något sånt hade jag en känsla av att det skulle vara ganska svårt att göra det och att jag inte skulle hitta något bra exempel på detta. Mina förningar stämde dock inte alls för jag hittade ett riktigt bra exempel och fick det att fungera på webbläsaren i ett nafs.

Eftersom det är operativsystemets egna printdialog som körs var detta inte en svår funktion att skapa. Om man däremot skulle behövt skapa en egen printdialog hade

det genast varit mycket svårare. Jag har gjort så att om man klickar på förhandsgranska öppnas en dialog där man ser en förhandsgranskning på webbsidan man befinner sig på medan om man klickar på skriv ut öppnas en dialog där man kan välja printer och sedan skriva ut de sidor man vill skriva ut.

En liten fiffig funktion som jag blev inspirerad av från webbläsaren Arora var att då man försöker avsluta webbläsaren med flera flikar öppnade så dyker en meddelanderuta upp som varnar om detta. En sådan funktion gjorde jag i min webbläsare då man antingen klickade på avsluta från huvudmenyn eller på x-krysset längst upp till höger då man hade fler än en flik öppen och försökte stänga webbläsaren.

Meddelanderutan meddelar att fler än en flik är öppen och frågar om man vill avsluta i alla fall. Om man klickar på "Ja" stängs webbläsaren medan "Nej" bara stänger meddelanderutan. Anledningen till att jag tycker att funktionen är fiffig är därför att ibland kan man i misstag klicka på stäng och då lämnar inte flikarna kvar längre tills man startar webbläsaren igen. Många webbläsare har dock en sådan funktion att flikarna sparas till nästa gång men jag har ingen sådan funktion i min webbläsare.



**Figur 37.** Meddelanderutan som dyker upp ifall man försöker stänga webbläsaren och man har fler än en flik öppen. [egen bild]

```

if (ui->flikWidget->count() > 1)
{
    varning.exec();
    {
        if (varning.clickedButton() == jaButton) {
            close();
        }
        if (varning.clickedButton() == nejButton) {
            event->ignore();
        }
    }
}

else
close();

```

**Figur 38.** Koden som körs om man har fler än en flik öppen då man försöker stänga webbläsaren. [egen bild]

En sak som finns inkluderad i de flesta applikationer är information om applikationen. Detta brukar man ha under hjälp i huvudmenyn. Jag skapade en dialog som öppnas då man klickar på "Om Simplex" från hjälp i huvudmenyn. I dialogen finns kort lite information om vad programmet är för någonting och hur man kan ta kontakt med den som skapat programmet.

#### 9.2.4 Flikar och fönster

Eftersom flikar gör det väldigt smidigt att surfa ville jag givetvis implementera detta i min webbläsare. Flikar fick dessutom delat med bokmärken mest röster om de viktigaste funktionerna i min webbläsare. Istället för att använda sig av en massa fönster (som man gjorde tidigare då inte flikar var "uppfunna") kan man nuförtiden surfa i flikar vilket gör det hela mycket enklare och mer överskådligt.

Att programmera så man kunde använda sig av flikar då man surfade var en rätt så jobbig del av arbetet med denna webbläsare. Det var många nya saker som man inte förstod sig på men tack vare Robert Metsärantas hjälp fick jag goda instruktioner och tips angående förverkligandet av användning av flikar. Självklart har jag också gjort att man kan öppna nya fönster av webbläsaren, för de personer som föredrar att använda sig av flera fönster istället för flera flikar.

### 9.2.5 Sökfunktionen

En sökfunktion är ett måste i en webbläsare. Självt använder jag sökfunktionen dagligen då jag inte orkar leta efter ett visst ord på en viss plats utan låter sökfunktionen leta rätt var på webbsidan ordet jag söker finns eller om det alls finns. Sökfunktionen var riktigt rolig att göra. Min sökfunktion innehåller en textruta där man kan skriva in sökordet, knappar för att söka nästa och föregående träff av sökordet, en knapp för att markera alla träffar av sökordet samt en knapp för att stänga sökfältet. Vid ett senare tillfälle lade jag också till en knapp för att tömma sökrutan.

Om det inte finns någon text i sökrutan eller om sökordet inte finns på den webbsida man befinner sig på är knapparna för nästa och föregående sökträff samt knappen för att markera alla sökträffar avaktiverade. Om sökordet man angett i sökrutan inte finns på den aktuella webbsidan dyker ett meddelande upp som säger att sökordet inte finns och att man skall pröva med ett annat sökord istället. Om sökordet hittas markeras det med grå färg och man kan gå till nästa och föregående sökträff medans om man klickar på knappen markera alla markeras alla sökträffar med en gul markeringsfärg. Om man har skrivit in text i sökrutan och enkelt vill ta bort texten kan man klicka på töm sökrutan som tömmer det som finns i sökrutan.

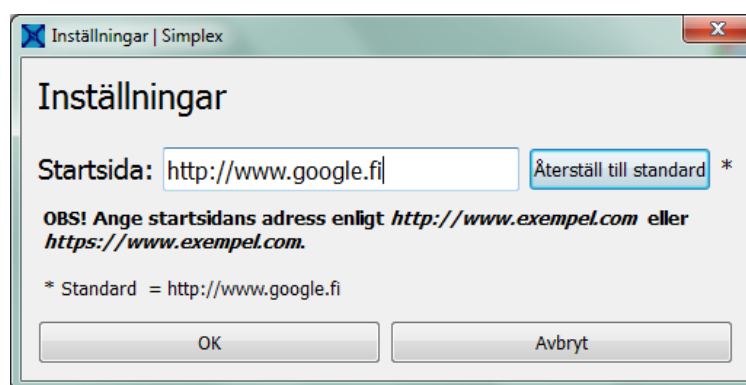


Figur 39. Sökfunktionen. [egen bild]

### 9.2.6 Inställningar för webbläsaren

De flesta större webbläsare har en hel del inställningar som användare kan ändra på efter behov. Dessa brukar vara att kunna ändra startsidan, ange vart filer skall sparas, säkerhetsinställningar, fontinställningar samt en massa annat. Då denna webbläsares syfte är att vara enkel och lättanvänd har jag bara en enda inställning och den inställningen är att ändra startsidan. Det är den allra viktigaste inställningen i mitt tycke och då många av inställningarna skulle varit för svåra att göra enligt kunskapen jag har valde jag att välja bort dem.

För inställningarna (eller inställningen i detta fall) har jag gjort att en dialog öppnas där man kan kopiera in eller skriva in helt själv webbadressen man vill använda som startside. För att spara startsidan klickar man på knappen "OK" i dialogen. Då man sedan startar webbläsaren på nytt visas startsidan man angav i inställningarna. Om man vill ändra startsidan till standard har jag en knapp för det också, vilket sätter Google.fi som startside. Man kan när som helst gå till den aktuella startsidan genom att klicka på hemknappen.



**Figur 40.** Webbläsarens inställningar. [egen bild]

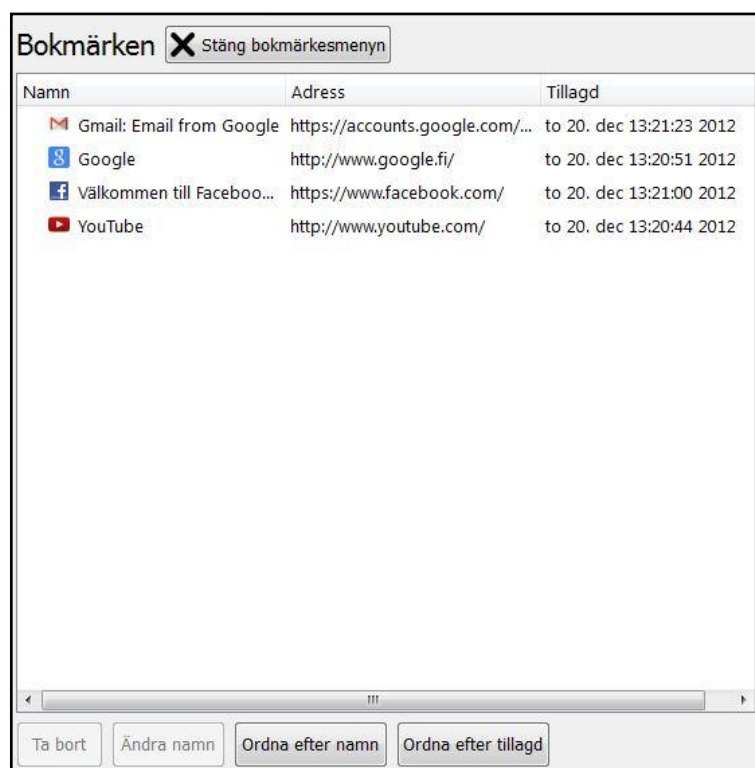
### 9.2.7 Bokmärken

Bokmärken blev delat med flikar vald till den viktigaste funktionen i min webbläsare. Med hjälp av bokmärken kan man spara viktiga webbsidor till ett senare tillfälle för att komma åt dem då. Bokmärken var den funktion som var allra svårast att tillämpa i min webbläsare. Det fanns många tänkbara sätt att göra bokmärken på. Först var det planerat att jag skulle ha en skild dialog var man kunde se alla bokmärken man lagt till. Först hade jag också planerat att man kunde skapa mappar som man kunde kategorisera bokmärkena i.

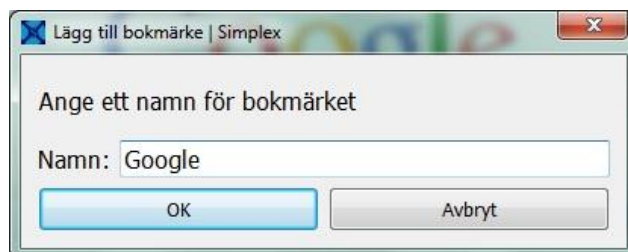
Allt detta blev lite för svårt för mig och jag beslöt att placera bokmärkena i samma dialog som själva webbläsaren istället. Då man klickar på knappen för att visa alla bokmärken dyker en lista med dem upp till vänster om webbläsarvyn. Under listan finns några knappar för att manipulera och visa bokmärkena enligt egna önskemål. Jag har en knapp med vilken man kan ta bort ett bokmärke, en för att ändra namn på bokmärke och så kan man ordna bokmärkena enligt namn eller enligt när de blev tillagda. Det finns naturligtvis även en knapp för att stänga

bokmärkesmenyn. Eftersom bokmärkena kan ordnas enligt namn eller när de blev tillagda borde man klara sig riktigt bra utan mappar.

För att lägga till ett bokmärke klickar man på "lägg till bokmärke" och då öppnas en dialog dit man skall ange namnet på bokmärket. Titelnamnet på den webbsida man befinner sig på finns färdigt angett för bokmärket men man kan också ange valfritt namn då man sparar bokmärket. Då man klickar på "OK" i dialogen läggs bokmärket till i listan över bokmärken. För att gå till ett bokmärke i listan över bokmärken kan man dubbelklicka på bokmärket.



**Figur 41.** Bokmärkesmenyn. [egen bild]



**Figur 42.** Dialog för att lägga till bokmärke. [egen bild]

### 9.2.8 Högerklicksmenyer

Högerklicksmenyer var något som jag inte hade tänkt så mycket på förrän i slutet av programmeringen. Högerklicksmenyer är väldigt viktiga att ha med eftersom många människor använder dem istället för snabbkommandon eller knapparna direkt från verktygsfältet. Jag har i min webbläsare skapat högerklicksmenyer på samtliga texttrutor men också i själva webbläsarvyn och i flikfältet. Det fanns färdigt gjorda högerklicksmenyer från början men de kunde jag inte använda då de var på engelska och jag ville ha dem på svenska. Jag försökte därför ta reda på om man kunde översätta dem till svenska men det kunde man tydligen inte så jag gjorde dem helt från början istället.

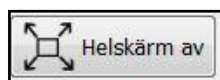
### 9.2.9 Sista finputsningen

Då jag egentligen hade programmerat klart allting fanns det ännu vissa saker att göra. Då man i lugn och ro går igenom och testar funktionerna i webbläsaren kan man märka att det inte riktigt fungerar som det borde eller att knappar är för små eller att det rent av saknas någonting. Under sista finputsningen lade jag till ikoner för sökfältets- och bokmärkesmenyns stängknappar. Jag ändrade storleken på knappar som borde vara något större och lade till en knapp för att tömma sökrutan i min sökfunktion.

Jag lade också till att en meddelanderuta dyker upp då man ändrat/sparat startsidan som bekräftar att startsidan har ändrats och skriver ut till vilken webbsida den har ändrats. Nästan på samma vis lade jag också till att en meddelanderuta dyker upp då man lagt till ett bokmärke som meddelar att bokmärket lades till.

Tidigare fanns det en stängknapp på varje flik i webbläsaren. Det ändrade jag på i slutskedet för användarvänlighetens skull. Jag ändrade till att den sista fliken inte kan stängas, d.v.s. om bara en flik är öppen så finns det ingen stängknapp på den fliken. Om däremot två eller flera flikar är öppna kan man stänga vilken som helst tills en flik är kvar.

Det viktigaste jag gjorde under sista finputsningen var ändå att när man visar webbläsaren i helskärm dyker en knapp upp som man kan klicka på för att stänga av helskrmsläget. Det går naturligtvis också att stänga av helskrmsläget via huvudmenyn, snabbkommandot eller högerklicksmenyn men det är mycket enklare att stänga av det med hjälp av en knapp som är fullt synlig.



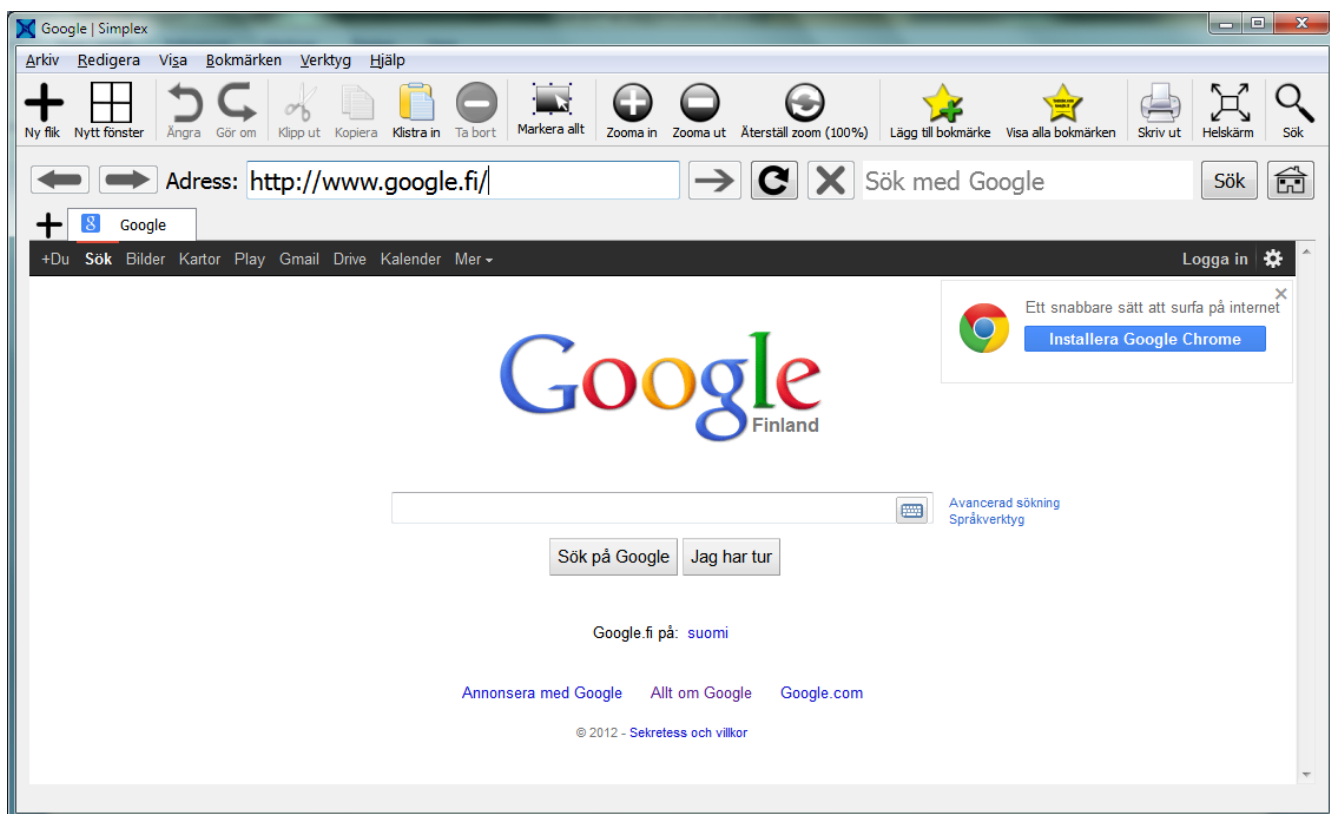
**Figur 43.** Knapp för att stänga av helskrmsläget. [egen bild]



## 10 RESULTAT

Resultatet av lärdomsprovet blev en webbläsare som förhoppningsvis skall vara enkel att använda och uppfylla målgruppens önskemål. Webbläsaren har testats på en annan dator än min egen för att kontrollera att webbläsaren fungerar som den borde även på andra datorer och enligt vad jag har testat fungerar allt som det ska.

Jag har efter detta lärdomsprov lärt mig en del om programmering i Qt och C++ men jag har också fått en hel del annan information då jag forskat angående webbläsare och det känns som jag förstår bättre nu hur program/applikationer byggs upp samt att jag har fått en bättre bild av hela processen.



Figur 44. Slutresultatet. [egen bild]

## 11 SLUTLEDNING

De vanligaste webbläsarna på marknaden har alla sina fördelar och nackdelar. Att de är enkla att använda för alla åldersgrupper tvekar jag dock på. Då jag gjorde olika tester på webbläsarna fanns det en ganska överlägsen vinnare bland dessa - Google Chrome. Google Chrome vann alla test förutom mitt eget test där minnesanvändningen testades.

Det har varit roligt och mycket lärorikt att undersöka webbläsarna samt att testa webbläsarna mot varandra.

Min undersökning fick elva stycken svar. På de flesta frågorna höll de svarande med mig medans på vissa frågor var det lite mer spridda svar, precis som det skall vara. Jag tyckte det var intressant att se hur de personer som svarat på undersökningen svarat och jag är nöjd över hur de svarat.

Då jag började på med själva programmeringen av webbläsaren var jag lite vilsen. I princip allt var nytt och det mesta kändes jobbigt och informationen om hur man skulle programmera en webbläsare kändes svår att få tag på. Då Robert Metsäranta förklarade en del begrepp åt mig och då jag efter en tid lärt mig programmet lite bättre såg jag en ljuspunkt i det hela. Nu då jag vet att jag har lyckats få ihop en webbläsare för äldre personer som blev ungefär som jag planerat känns det bra. Syftet med mitt lärdomsprov har uppnåtts samtidigt som jag lärt mig en del om C++ programmering i Qt. Det känns som man fått mera kunskap och faktiskt har åstadkommit någonting.

Om jag i något senare tillfälle i livet får mycket tid över kommer jag kanske att fortsätta utveckla webbläsaren. Detta är första versionen av webbläsaren och givetvis finns det mycket man kan förbättra och jobba vidare på.

## KÄLLOR

### Elektroniska publikationer

- [1] Apple. Vad är Safari? [online] [Uppdaterad 2012].[hänvisning 10.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.apple.com/se/safari/what-is.html>>
- [2] Centennial Arts. Comparison on all Major Web Browsers: Internet Explorer, Safari, Firefox, and Google Chrome [online] [Uppdaterad 08.06.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://articles.centennialarts.com/2012/06/08/comparison-on-all-major-web-browsers-internet-explorer-safari-firefox-and-google-chrome/>>
- [3] Code Project. History of Internet Explorer [Birth of #IE to #IE9] [online] [Uppdaterad 12.03.2011].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.codeproject.com/Articles/167829/History-of-Internet-Explorer-Birth-of-IE-to-IE9>>
- [4] Compound Thinking. What is Simplicity? [online] [Uppdaterad 31.05.2006].[hänvisning 08.06.2012]. Tillgänglig i form av www-dokument: <URL: <http://compoundthinking.com/blog/index.php/2006/05/31/what-is-simplicity/>>
- [5] ed-informatics.org. Usability, Learnability, Memorability [online] [Uppdaterad 28.12.2009].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://ed-informatics.org/2009/12/28/medical-computing-1/>>
- [6] eHow tech. The history of the Safari Browser [online] [Uppdaterad 16.04.2012].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: [http://www.ehow.com/about\\_5139105\\_history-safari-browser.html/](http://www.ehow.com/about_5139105_history-safari-browser.html/)>
- [7] Ett smultronställe för Internetblåbär. Den levande IT-ordlistan [online] [Uppdaterad 15.01.1997].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.acc.umu.se/help/smultron/ordlista.htm>>
- [8] Gizmo's Freeware. Best Free Web Browser [online] [Uppdaterad 07.09.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.techsupportalert.com/best-free-web-browser.htm>>
- [9] Goodreads.com. Albert Einstein, quotes [online] [Uppdaterad 2012].[hänvisning 08.06.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.goodreads.com/quotes/show/277>>
- [10] Lissy.se. Test: IE9 vs Google Chrome vs Safari vs Firefox vs Opera [online] [Uppdaterad 25.03.2011].[hänvisning 19.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.lissy.se/2011/03/25/test-ie9-vs-google-chrome-vs-safari-vs-firefox-vs-opera>>

- [11] Mozilla. History of the Mozilla Project [online] [Uppdaterad 01.02.2012].[hänvisning 03.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.mozilla.org/about/history.html>>
- [12] New Media Bytes. Anatomy of a Web Browser: Understanding Browser Terms [online] [Uppdaterad 28.12.2008].[hänvisning 14.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.newmediabytes.com/2008/12/28/understanding-web-browser-anatomy-terms/>>
- [13] Nielsen Norman Group. Usability 101: Introduction to Usability [online] [Uppdaterad 04.01.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.nngroup.com/articles/usability-101-introduction-to-usability/>>
- [14] Opera. About Opera [online] [Uppdaterad 2012].[hänvisning 03.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.opera.com/company/>>
- [15] PC för Alla. Här är vinnaren av webbläsartestet [online] [Uppdaterad 12.03.2010].[hänvisning 18.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://pcforalla.idg.se/2.1054/1.301554/>>
- [16] PC för Alla. Så väljer du rätt webbläsare [online] [Uppdaterad 23.03.2009].[hänvisning 18.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://pcforalla.idg.se/2.1054/1.219174/sa-valjer-du-ratt-webblasare>>
- [17] Pcmag.com. Apple Safari 5.1.7 for Windows [online] [Uppdaterad 07.08.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.pcmag.com/article2/0,2817,2364856,00.asp>>
- [18] Pcmag.com. Firefox 18 [online] [Uppdaterad 09.01.2013].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.pcmag.com/article2/0,2817,2349494,00.asp>>
- [19] Pcmag.com. Google Chrome 23 [online] [Uppdaterad 14.11.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.pcmag.com/article2/0,2817,2373853,00.asp>>
- [20] Pcmag.com. Internet Explorer 9 (IE9) [online] [Uppdaterad 31.07.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.pcmag.com/article2/0,2817,2369160,00.asp>>
- [21] Pcmag.com. Opera 12 [online] [Uppdaterad 06.08.2012].[hänvisning 19.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://www.pcmag.com/article2/0,2817,2375401,00.asp>>
- [22] Qt.digia.com. About us [online] [Uppdaterad 2013].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://qt.digia.com/About-us/>>

- [23] Qt.digia.com. Product [online] [Uppdaterad 2013].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://qt.digia.com/Product/>>
- [24] Qt.digia.com. Product, Developer Tools [online] [Uppdaterad 2013].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://qt.digia.com/Product/Developer-Tools/>>
- [25] Qt.digia.com. Product, Qt SDK [online] [Uppdaterad 2013].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: <http://qt.digia.com/Product/Qt-SDK/>>
- [26] Uc Berkeley Library. Glossary of Internet & Web Jargon [online] [Uppdaterad 2012].[hänvisning 14.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.lib.berkeley.edu/TeachingLib/Guides/Internet/Glossary.html>>
- [27] Usability Net. What is usability? [online] [Uppdaterad 2006].[hänvisning 08.06.2012]. Tillgänglig i form av www-dokument: <URL: [http://www.usabilitynet.org/management/b\\_what.htm](http://www.usabilitynet.org/management/b_what.htm)>
- [28] W3C. About W3C [online] [Uppdaterad 2012].[hänvisning 13.12.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.w3.org/Consortium/>>
- [29] W3Schools. W3C HTML Activities, HTML Versions [online] [Uppdaterad 2013].[hänvisning 21.02.2013]. Tillgänglig i form av www-dokument: <URL: [http://www.w3schools.com/w3c/w3c\\_html.asp](http://www.w3schools.com/w3c/w3c_html.asp)>
- [30] Wikipedia. Google Chrome [online] [Uppdaterad 29.12.2012].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: [http://sv.wikipedia.org/wiki/Google\\_Chrome](http://sv.wikipedia.org/wiki/Google_Chrome)>
- [31] Wikipedia. History of Internet Explorer [online] [Uppdaterad 01.01.2013].[hänvisning 09.01.2013]. Tillgänglig i form av www-dokument: <URL: [http://en.wikipedia.org/wiki/History\\_of\\_Internet\\_Explorer](http://en.wikipedia.org/wiki/History_of_Internet_Explorer)>
- [32] WordPressGuru. Hitta rätt i djungeln av webbläsare, test av de 5 största [online] [Uppdaterad 18.03.2010].[hänvisning 18.04.2012]. Tillgänglig i form av www-dokument: <URL: <http://www.wordpressguru.se/hitta-ratt-i-djungeln-av-webblasare-test-av-de-5-storsta/>>

**INSTRUKTIONER FÖR INSTALLATION AV SIMPLEX**

Instruktioner för installation av  
Simplex

## **INNEHÅLL**

1. SYSTEMKRAV .....	3
2. INSTALLERA SIMPLEX .....	3
3. AVINSTALLERA SIMPLEX.....	6

## 1. SYSTEMKRAV

Simplex kräver 47 MB (Megabyte) ledigt utrymme på hårddisken. Programmet har bara testats i Windows 7 men borde fungera i Windows Vista och förhoppningsvis i Windows XP också. Simplex kräver inte mycket processorkraft eller RAM-minne men det rekommenderas ändå att ha minimum 1 GHz processor och 512 MB RAM-minne.

## 2. INSTALLERA SIMPLEX

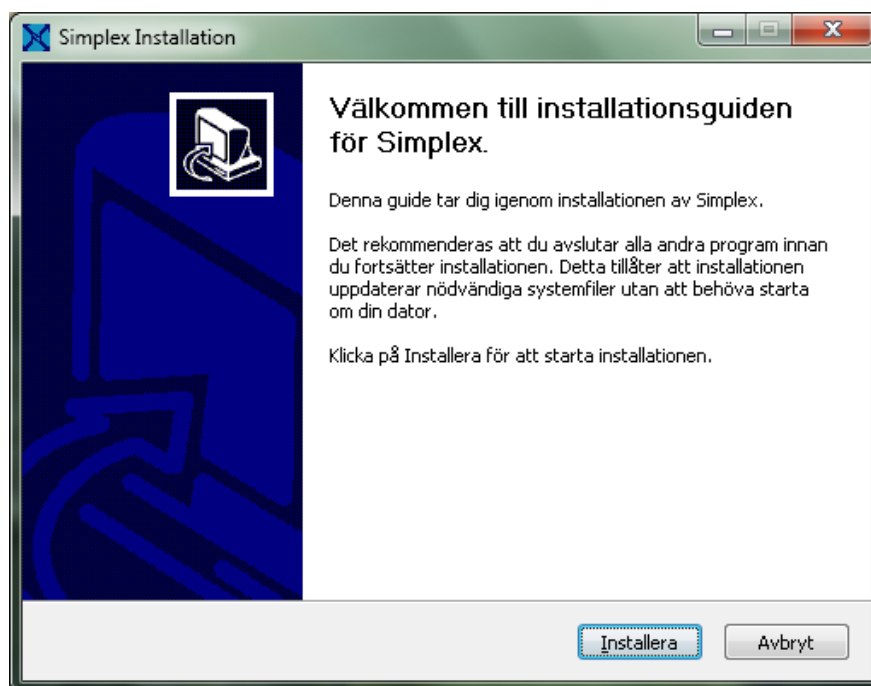
Denna guide går igenom hur du installerar Simplex på din dator.

1. Dubbelklicka på exefilen med namnet "Simplex installation".

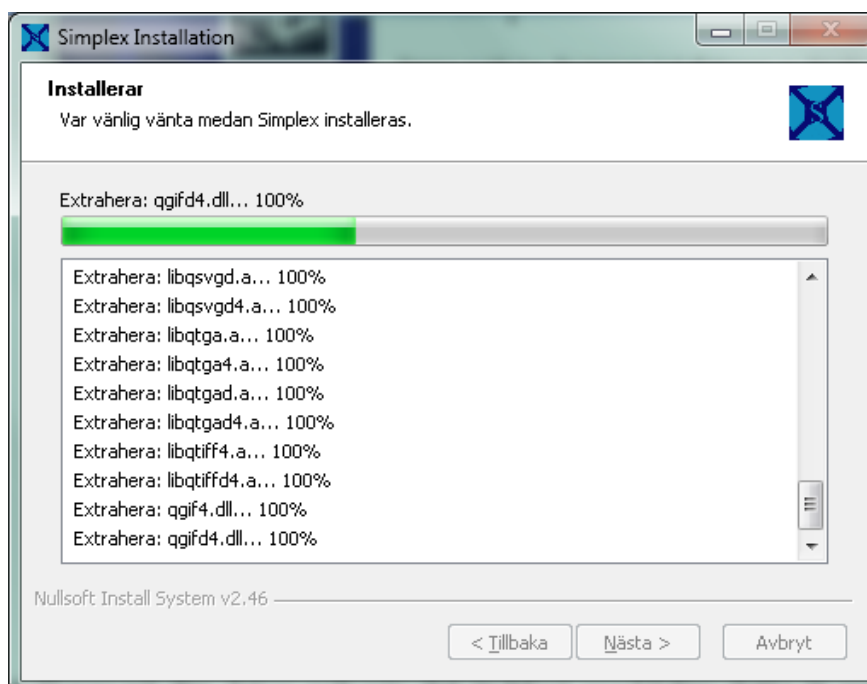


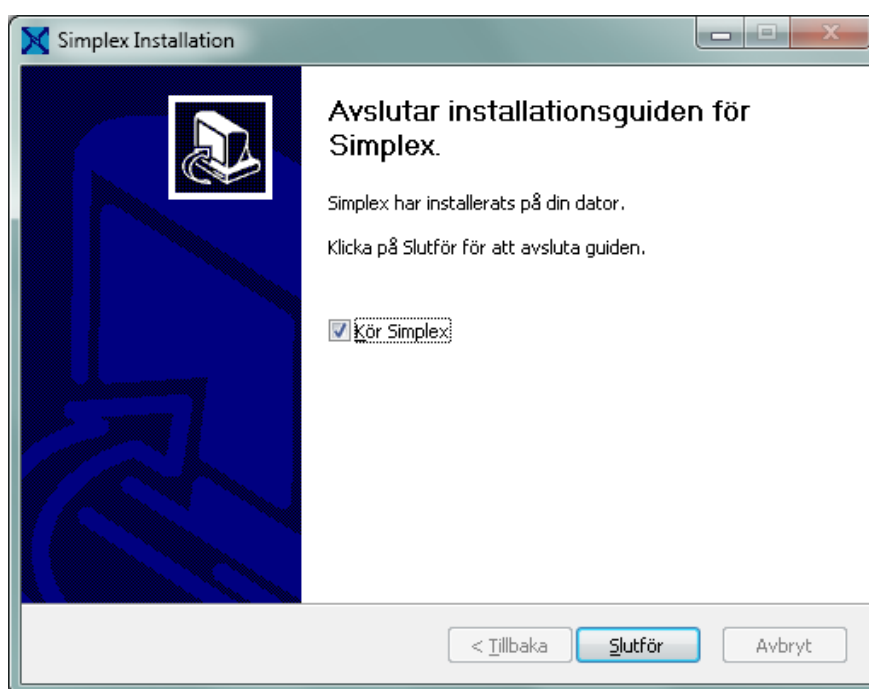
2. Det dyker upp en dialog där det står "Välkommen till installationsguiden för Simplex." Klicka på "Installera" för att installera Simplex.





3. Simplex installeras på din dator och när det är klart meddelas detta åt användaren. Om man vill att Simplex skall köras direkt efter man slutfört installationen kan man ha ikryssat rutan "Kör Simplex". Rutan är som standard ikryssad men om man inte vill att Simplex skall köras direkt efteråt kan man kryssa ur rutan. Klicka på "Slutför" för att slutföra installationen.





Efter installationen skapas en genväg av Simplex på skrivbordet och i startmenyn samt en mapp "Simplex" under "Alla program" (All programs) i startmenyn.

### 3. AVINSTALLERA SIMPLEX

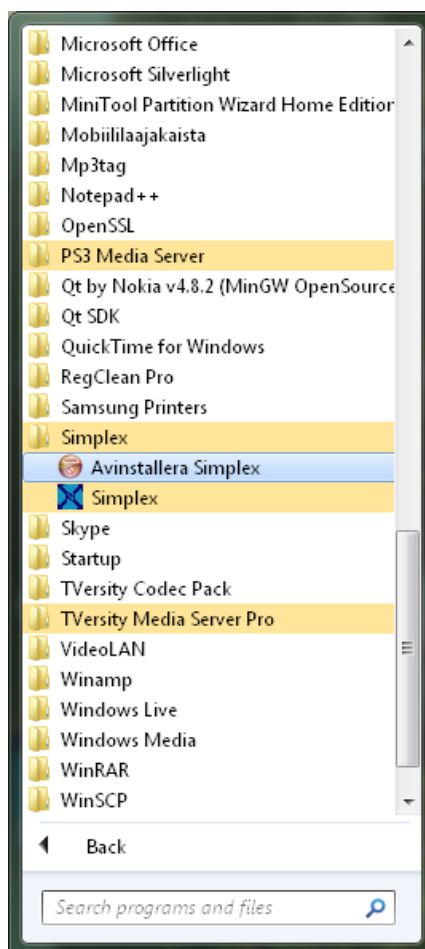
Denna guide går igenom hur du avinstallerar Simplex på din dator. Att avinstallera Simplex kan göras på åtminstone tre sätt.

#### Alternativ 1 – Via startmenyn

Klicka på knappen "Start" som finns längst ner till vänster i Windows aktivitetsfält. (se bild nedan)



Klicka på "Alla program" (All programs) för att se alla program. Gå neråt i listan tills du hittar mappen "Simplex". Klicka på mappen så att du ser någonting som heter "Avinstallera Simplex" och klicka sedan på det.



En dialog dyker upp som frågar om man vill avinstallera Simplex. Klicka på "Ja" (Yes).



Simplex avinstalleras och en dialog som meddelar att Simplex har avinstallerats dyker upp när det är klart. Klicka på "OK".



### Alternativ 2 – Via kontrollpanelen

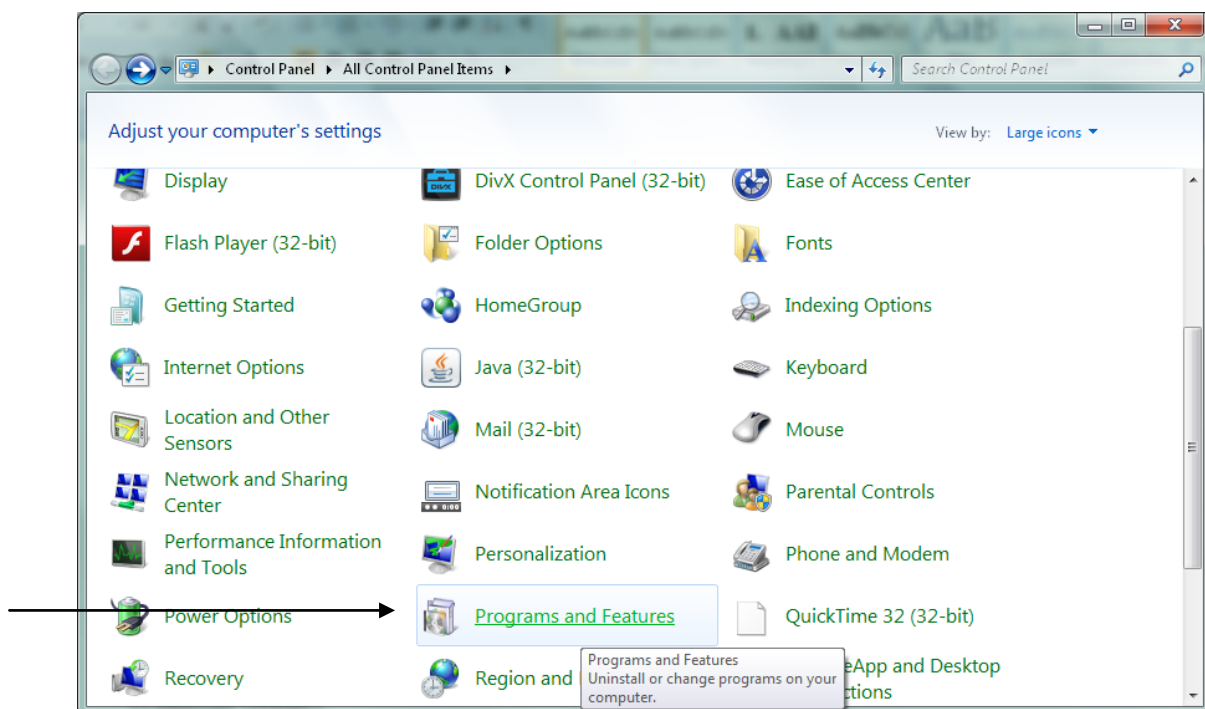
Klicka på knappen "Start" som finns längst ner till vänster i Windows aktivitetsfält. (se bild nedan)



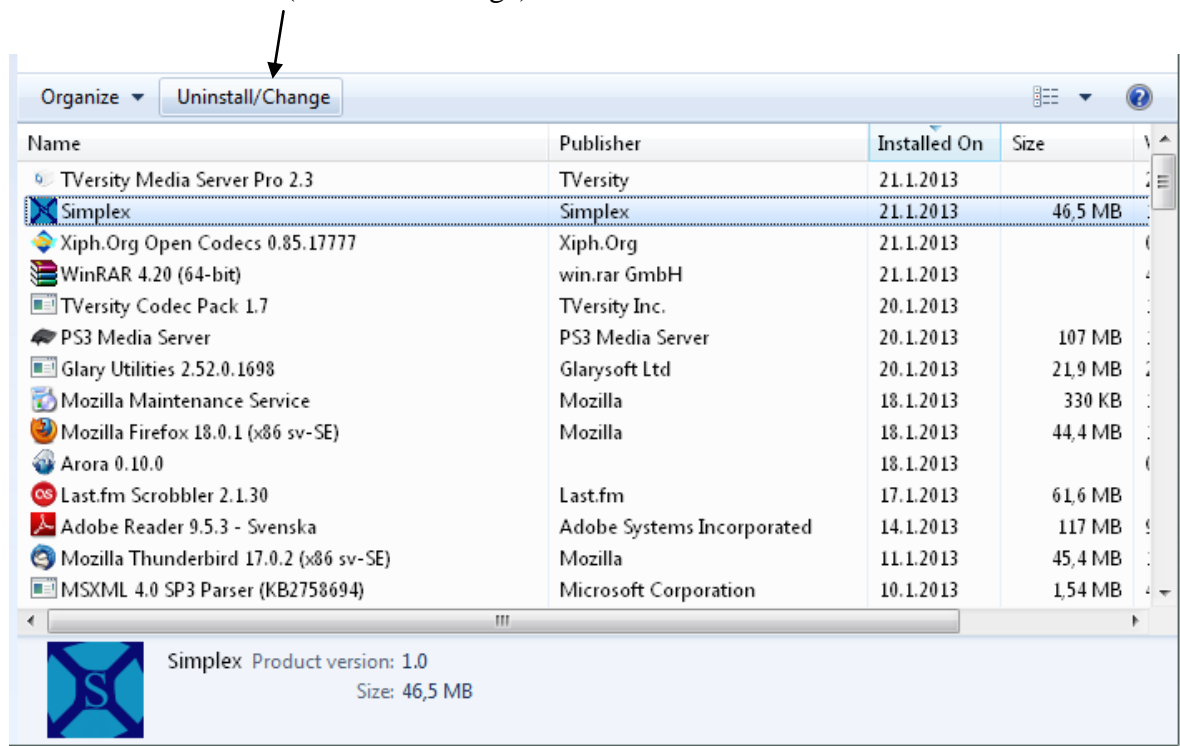
Klicka sedan på "Kontrollpanelen" (Control Panel).



Kontrollpanelen öppnas. När den har öppnats kan du klicka på "Program" och sedan "Program och funktioner" (Programs and features) eller direkt på "Program och funktioner", beroende på versionen av Windows. I Windows XP heter det "Lägg till eller ta bort program" (Add or remove programs).



Klicka på programmet Simplex i listan över program och välj "Avinstallera/Ändra" (Uninstall/Change).



En dialog dyker upp som frågar om man vill avinstallera Simplex. Klicka på "Ja" (Yes).



Simplex avinstalleras och en dialog som meddelar att Simplex har avinstallerats dyker upp när det är klart. Klicka på "OK".

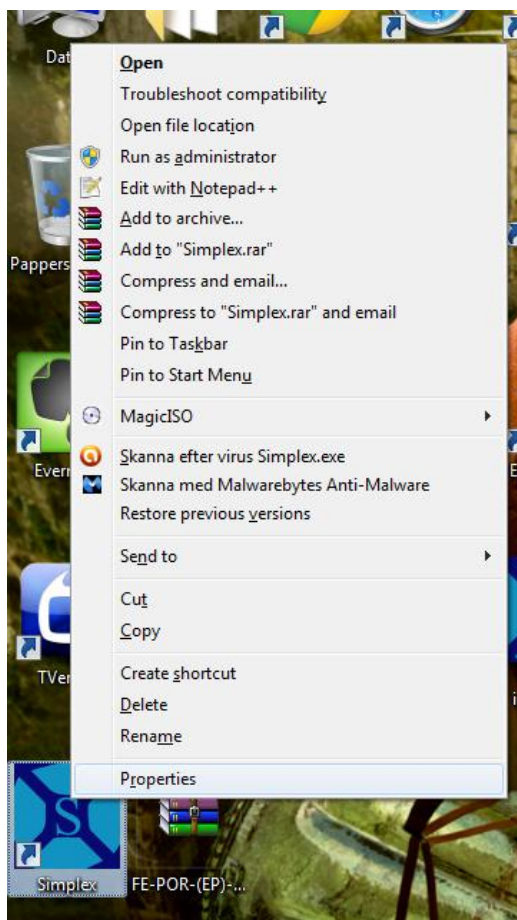


### Alternativ 3 – Gå in i installationsmappen

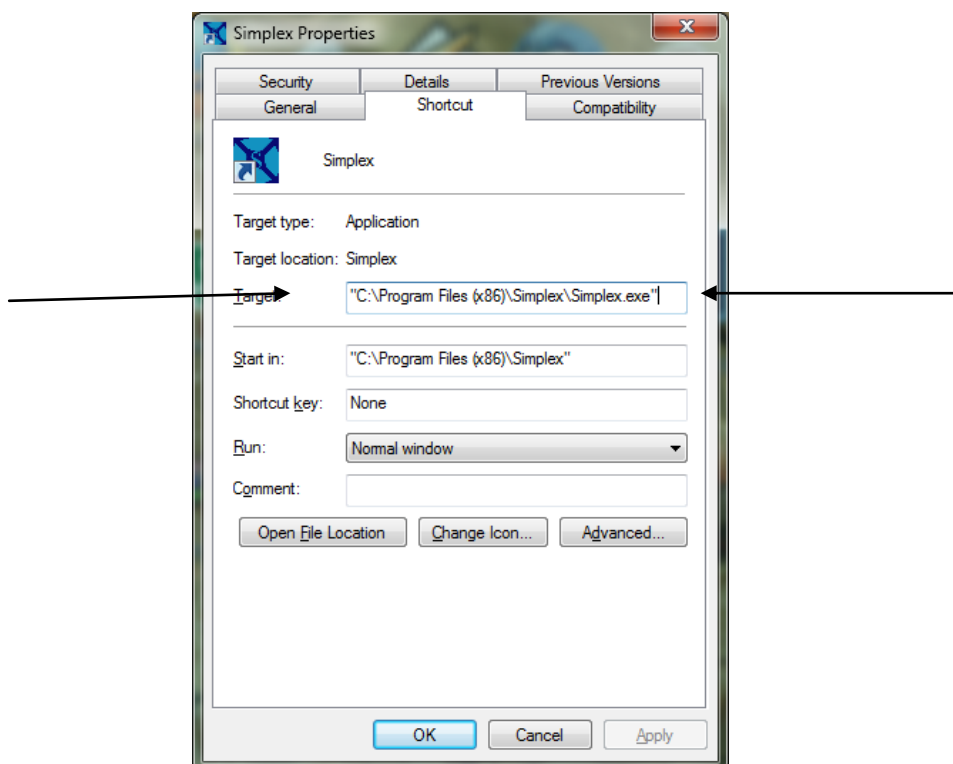
Klicka på "Dator" (Computer) eller "Den här datorn" (My Computer) antingen via skrivbordet eller via startmenyn.



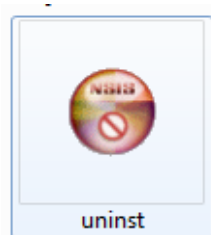
Gå in i mappen var Simplex installerades i. Om du inte vet vart Simplex installerades kan du kontrollera detta genom att högerklicka på Simplex genvägen på skrivbordet och välja "Egenskaper" (Properties). Simplex borde ha installerats i "Program Files (x86)" oftast på enhet "C" eller vilken bokstav man nu använder för enheten och sedan i en mapp som heter Simplex.







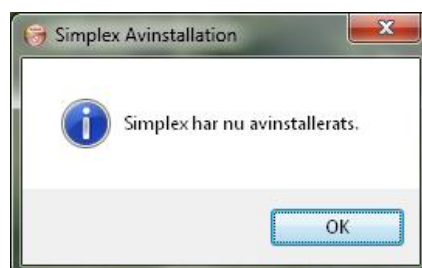
Om du hittat fram till mappen borde du se en fil som heter "uninst".  
Dubbelklicka på den för att avinstallera programmet.



En dialog dyker upp som frågar om man vill avinstallera Simplex. Klicka på "Ja" (Yes).



Simplex avinstalleras och en dialog som meddelar att Simplex har avinstallerats dyker upp när det är klart. Klicka på "OK".



## UNDERSÖKNINGEN I E-LOMAKE

### En enkel webbläsare med ett simpelt användargränssnitt

Blankett är tidsinställd: offentlighet upphör 14.10.2012 0.00

Jag har valt att göra mitt lärdomsprov om att undersöka de webbläsare som finns på marknaden samt att till slut programmera en enkel webbläsare med ett simpelt användargränssnitt för en tilltänkt målgrupp. Webbläsaren skall vara lätt att använda och utan många onödiga krusiduller. För att göra webbläsaren behöver jag åsikter från en målgrupp som jag sedan sammanfattar och försöker tillämpa i själva webbläsaren. Med detta frågeformulär försöker jag nå min målgrupp.

**Webbläsaren är främst till för åldersgruppen 50 år och uppåt.**

**Lägg muspekaren över frågetecknet så kommer en förklaring!**

//Anders Palmroos

#### Fråga 1

Vilka funktioner av dessa är för dig mest nödvändiga i en webbläsare? ?

- Bokmärken
- Användning av flikar
- Historik
- Tillägg
- Sökfunktionen (Ctrl+F)
- Privat surfning
- Ändring av utseendet
- Zooma in/ut

#### Fråga 2

Hur tycker du att en enkel webbläsare skall se ut? ?

- Stora fält och knappar (adressfält, bakåt/framåtknapp, refresh)
- Lagom stora fält och knappar
- Logiskt placerade fält och knappar
- Logiskt utformad i sin helhet
- Bara det nödvändigaste skall finnas och det skall vara lätt att hitta

#### Fråga 3

Välj av en skala från 1-5 där 1 = helt av annan åsikt, 2 = delvis av annan åsikt, 3 = ingen åsikt, 4 = delvis av samma åsikt, 5 = helt av samma åsikt

Se bild nedan för mera förklaringar!

	1	2	3	4	5
En enkel webbläsare skall bara innehålla det mest nödvändiga	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Sökrutan skall vara till höger om adressfältet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Bakåt och framåtknappen skall vara till vänster om adressfältet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Uppdatera sidan knappen skall vara till höger om adressfältet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
En meny för inställningar skall finnas längst upp till vänster	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
En knapp som tar dig tillbaka till startsidan skall vara till höger om sökfältet	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



Navigeringsfältet i Mozilla Firefox. Påståenden 2-6 från fråga 3 överensstämmer med Mozilla Firefox, vilket jag utgick mina påståenden ifrån.

#### Fråga 4

Vilken bakgrundsfärg tycker du webbläsaren skall ha?

Skicka

Skicka uppgifterna

**LOGON FÖR WEBBLÄSAREN SIMPLEX**

