

PM

Modul 3: Animering

Inledning

Jag sparade denna modul till sist. Den tog längs tid av alla att genomföra, inte minst för att den introducerade så mycket nytt, nya begrepp, nya tekniker. Det var väldigt mycket teoretiskt material om teknikerna vi skulle använda. Flash hade vi visserligen gått igenom, men HTML5-Canvas och SVG/VML var för mig nya och nästintill utforskade ämnen. Det höll faktiskt på att bli överväldigande, men lärarens tips om vilken ordning vi skulle bearbeta materialet och lösa uppgifterna, hjälpte mycket.

Flash

Mycket riktigt, direkt efter att ha tillgodogjort mig teorin, startade jag med Flash-delen. Det hade gått rätt mycket tid sen Action-script-kursen, men det var inga större svårigheter att komma tillbaka in i kodningen med Flash Builder. Jag fick ganska snart ihop en animering som i stora drag liknade den i kursmaterialet. Jag ska här inte redogöra för detta i detalj, då källkoden i filen *Multimedia_animation.mxml* är välkommenterad.

SVG/VML- Raphael

Sedan var det dags att ta itu med Raphael-animeringen. Det var inga problem att förstå sig på det grundläggande i koden och att experimentera fram hur man kunde modifiera animeringarna. Så jag ritade upp figurerna så nära kursmaterialets som möjligt och började göra ändringar som skulle få objekten att rotera och röra på sig.

Ett litet problem som uppstod var hur kunna få den blå rektangeln att rotera flera gånger utan att den kom tillbaka till utgångsläget. Det krävdes ganska mycket sökande på nätet innan jag hittade att notationen *...r120* instruerar Raphael att lägga till rotationsgrader på de föregående.

I början hade jag ytterligare ett litet problem, men det var nog bara slarv: jag deklarerade inte vissa variabler globalt, utan inne i *init()* och fick därför inte åtkomst till dem senare i

koden. Detta åtgärdades dock snabbt. Lärdom: man ska inte koda när man är trött!

Burst-engine

HTML5_canvas och Burst-engine var lite knepigare och tog nog längst tid. Jag hade sparat denna del till slutet. Dels var dokumentationen knapp, dels var här ett helt nytt sätt att komma åt variabler och metoder som gjorde mig i början väldigt osäker ock konfunderad. Jag läste dokumentationen parallellt som jag experimenterade med koden och observerade resultaten i webbläsaren. Begreppen med timeline, shape, track, keyframes osv var svåra att sätta sig in i och notationen med de långa raderna gjorde koden ännu mer ogenomränglig. Men till slut började bilden klarna en smula. Jag började misstänka att lösningen till uppgiften låg i någon kombination av de två exemplen. Mycket riktigt fann jag att om man kastade om rektanglarnas färg samt satte egenskapen för animeringen på den blåa från *scl* till *rot*, så började lösningen av uppgiften att skymta. Jag kopierade således funktionerna *fram()* och *back()* in i koden för uppgiftens blåa rektangel och det fungerade hyfsat. Sedan tog det en stund för att få de rätta värdena för rotationpunkten att stämma.

För den röda rektangeln som skulle förflyttas fick jag leta en stund i dokumentationen och på nätet innan jag fann att det är egenskapen *left* som kan åstadkomma detta. Två olika ease-egenskper användes också. I källkoden har jag bredvid de ursprungliga kommentarerna lagt till mina egna.

Slutsats

Detta var, tillsammans med 3D-modulen, bland de mest intressanta och givande i hela kursen. Synd bara att tiden är knapp när man vill fördjupa sig i och gå vidare med tutorials och exempel.

320×180, 2 min 17,2 MB

video: 24 fps, 1200 kbps

Audio: 32 kHz, 64 kbps – som Mixdown valde jag den förinställda för HandBrake: "Dolby Pro Logic II"

Detta ger oss en (uträknad) medel-bithastighet på:

Filens storlek: 17,2 MB = 17.200.000 Bytes = 137.600.00 bits

Tiden 2 min = 120 sek

$137.600.000/120 = 1147.667 \text{ bps} = 1147 \text{ kbps} (1,15 \text{ Mbps})$

Originalfilmen hade jag fått till:

2638 kbps (om detta nu kan stämma)

Slutsats: Värden på bithastighet som är lägre än 1200 kbps ger dålig bildkvalitet, medan högre värden gör filen för tung (över 20 MB) och verkar överflödiga.

De övriga formaten blev:

640×360, 2min 22,3, MB.

Den beräknade bithastigheten skulle isf vara:

120 sek.

$22,3 * 1000.000 * 8 / 120 / 1000 =$

1487 kbps (medel med ljud) =

1,49 Mbps

1280×720 2 min 43,7 MB.

Den beräknade bithastigheten skulle isf vara:

120 sek.

$43,7 * 1000.000 * 8 / 120 / 1000 =$

2913 kbps (medel med ljud) =

2,9 Mbps