

Rapport U3

Jag började mitt program med ett "flödesschema", där jag skrev ut tydligt vad som skulle göras och lite om hur det skulle kunna tänkas att göras. Det finns kvar längst upp i rapporten. Sedan gick jag igenom html och css koden, studerade hur allt är uppbyggt för att få en klarare bild om hur implementeringen av javascript koden skulle kunna se ut. Efter detta gjorde jag programets skal, alla funktioner (tomma), för att sätta en sorts bas och sedan utveckla utifrån det. Sen la jag in den flödesschema/algoritmer för koden i diverse funktioner där jag tänkte att de skulle utföras. Det blev en del ändringar under skapandet av javascriptskoden, fick använda mig mycket av iteration av både kodning, föreläsningss filmer osv. Men känner ändå att det definitivt var effektivaste sättet att påbörja uppgiften. Även enklare att kolla i koden i programmet än att kolla på hemsidan varje gång.

Jag väljer att starta längst upp i koden och sedan gå ner steg för steg.

Mitt nästa steg var att deklarera alla globala variabler jag skulle tänka mig att kunna använda mig av i programmet. I slutändan använde jag näst intill alla – om inte alla. Efter det skapade jag init funktionen. Där initierade jag mina globala variabler och la kopplingar till gränssnittet. Här fick jag även en ännu bättre bild på hur programmet kommer struktureras. I init funktionen sparar jag den kopierade kortleken som senare kommer att användas i getnewbricks – för att få fram nya brickor. I init tänkte jag också att det var smidigt att spara localStorage, med tanke på att det är det första användaren ser när man går in på programmet. Jag lägger på händelsehanterare för knapparna och avaktiverar newbricksbtn direkt när sidan uppdateras. Självklart tänkte jag inte på sparandet av localStorage i början, men jag förklarar så här, så om du kollar på min kod – slipper hoppa fram och tillbaka.

Vid startGame hämtar jag alla brickor och bockar (mark), för att sedan kunna rensa brädan och X och bock när spelet är slut (då startgamebtn knappen klickas på). Med tanke på att det inte finns något relevant som tas bort i brädan initialt, så påverkas det inte. Vet inte om det är meningen att alla ska göra så, men kändes som två flugor i en smäll med en funktion där. Jag avaktiverar även startgame efter klick och aktiverar newbricksbtn.

I getNewBricks plockar jag fram bilden de 4 bilderna med math random och sedan tar bort brickorna med splice, så de inte kan kommas att dra flertal gånger. Jag lägger på händelsehanterare för att aktivera drag.

dragStart sätter jag också på händelsehanterare, men för drop zone istället för tidigare i getnewbricks där jag sätter på för att kunna dra de fyra brickorna. Jag sparar även bilden som dras i dragnumberelem för att sedan kunna plocka fram den. Sätter även draggable true.

I dragEnd tar jag bort, så inget objekt i webbskärmen kan dras. Så inte något som inte ska dras kan dras.

I dragDrop funktionen gör jag en hel del mer. Sätter bakgrundsfärg vid dragover, tar bort vid dragleave och sedan kodar vad som ska hända i drop. Jag sätter också e.preventDefault för att förhindra webbsidans default beteende. Jag börjar med att ta bort bakgrundsfärg när man släpper brickan och sedan lägger in vita bilden i dragnumberelems src attribut för att få med den till de 16 brickorna. Initialt tänkte jag inte att det skulle tillkomma så mycket i drop funktionen men kändes som att det näste var ett måste (inte för det spelar någon roll?). Sedan hämtar jag ut bildens url som jag skickade med datatransfer innan och lägger in i min variabel. Sedan tar jag bort classen empty och lägger till brick till alla och tar även där bort eventlisteners. Genom classList här tar jag bort css klassen t.ex och ersätter den med en annan (tog mig ett litet tag att klura ut). För att kolla om de 16 brickorna använde jag mig av brickselem.length == 0 och om det är 0, så kallar jag på checkbox och avaktiverar newgamebtn. Jag går även igenom de fyra brickorna till höger (bricksimgelem) och plockar fram nummer via min funktion changetonumber, som jag går igenom senare. Jag kollar där då om min variabel num inte har några nummer i sig (NaN), då vet jag att den är tom och isåfall aktivera newbricksbtn så man kan klicka fram fyra nya brickor. Och eftersom jag har newbricksbtn.disabled = true i init() funktionen så avaktiveras knappen direkt vid start. Och till sist kollar jag om om brickselems längd är 0. Om den är 0 så har sista brickan lagts, den sista brickan i arrayen som håller de 16 brickorna (brickselem). Då är den tom och då är det dags att avaktivera newbricksbtn igen – för då är ju spelet klart.

Sedan kommer jag till funktionen changetonumber där jag plockar ut min bilds nummer i dess url, för att sedan kunna jämföra för en stigande serie. Men senare såg jag att det var möjligt att nyttja funktionen igen för att kolla de 4 brickorna NaNs, som jag beskrev innan. Jag skickar också med picNummer (return).

I min checkBoard funktion, framförallt så uppdaterar jag hela brädan efter sista brickan har lagts. Jag hämtar alla klasser för att också kunna jämföra stigande serie och mina id referenser för att sedan kunna skriva ut bock eller X. Jag uppdaterar även totala poäng då jag lägger in points += i totalPoints variabeln, samt antal spel ++ för varje gång användaren brädan uppdateras. Till sist sparar jag localStorage för att sedan kunna hämta det i init. Jag tänkte att det var smidigast att ha de här istället för i en ny funktion, men också för att jag hade problem med localStorage till en början. Men nu när jag tänker efter kanske det är snyggare med en ny funktion – vad tycker du? Jag sätter även Number när jag skickar iväg resultaten då de annars kan vara strings.

Min checkLine funktion tar jag fram koden för att kunna jämföra om det är en stigande serie eller inte. Om första brickan är större än andra, isåfall gå vidare mot en stigande serie, men är det inte - avbryt. Jag kallar denna funktion i checkbox.

I min sista funktion updateBock, uppdaterar jag för bock eller X och om det är bock så plussar jag även points. Jag hade nog kunnat lägga den någon annanstans, men fungerade bra under hela programmets gång. Jag lägger även till rött och grönt och skriver ut meddelande med msgelem. Om det är en stigande serie grön bock, annars rött X.

I helhet tycker jag att min förberedelse med programflödes texten och funktioner satte en bra grund för klarheten av arbetet och vad som ska varas med och lite om hur det ska implementeras. Utan den hade det blivit tuffare. Även att jag studerade html och css koden innan jag började kändes också givande. Många gånger man bara dyker rätt in i uppgiften utan förberedelse.