

Debreceni Egyetem Informatika Kar

Macromedia Flash Mx oktatóprogram

Témavezető:
Nyakóné dr. Juhász Katalin
tudományos főmunkatárs

Készítette:
Kovács Katalin
Informatikatanár

Debrecen
2009

Tartalomjegyzék

<i>Tartalomjegyzék</i>	2
<i>Bevezetés – Témaválasztás indoklása</i>	3
<i>Hatékony oktatóprogramok kiválasztása, elemzése</i>	5
A programok kiválasztásának szempontjai.....	5
A korszerű oktatóprogramokkal szemben támasztott követelmények elemzése ..	6
<i>Szoftverergonómia</i>	8
<i>Irányelvek az elektronikus kiadványok készítéséhez- tipográfia a monitoron</i>	10
A tipográfia alapvető céljai	10
Elektronikus kiadványok tipográfia tervezése	11
Képernyő tervének meghatározása	11
Betűtípus és betűméret meghatározása	13
Színek használata a monitoron.....	14
Képek használat egy elektronikus kiadványban.....	18
<i>Flash keretrendszere</i>	19
Flash története.....	19
Macromedia Flash Mx bemutatása.....	20
Munkaablak felépítése.....	20
Animáció típusai	21
Animációs technikák a Flash-ben:	22
Szimbólumok.....	23
ActionScript a Flash-ben.....	24
A Flash file típusai	25
<i>Oktatóprogram tervezése</i>	26
Tartalmi rész tervezése.....	26
Kinézet megtervezése.....	29
Színek meghatározása	30
<i>Oktatóprogram megvalósítása</i>	31
Kezdő oldal létrehozása.....	31
Főoldal elkészítése.....	35
Maszkolás tanegység elkészítése	37
<i>Oktatóprogram publikálása, használata</i>	39
<i>Összefoglalás</i>	42
<i>Irodalomjegyzék</i>	43

Bevezetés – Témaválasztás indoklása

Amikor jelenlegi tanulmányaim során eljutottam a szakdolgozatom témájának megválasztásához, úgy döntöttem, egy régóta bennem motoszkáló ötletet valósítok meg, konkrétan oktatóprogramot készíték a Macromedia Flash programhoz.

Mivel már a főiskola első évfolyamának befejezése óta tanítok, és mint oktató nekem is szembe kellett néznem azzal a ténnyel, hogy a számítástechnika oktatása nagyon sokrétű, nagyon sok olyan téma van, amit az iskola kötelező órakeretei között nem lehet kellőképpen megtanítani, ugyanakkor a tanulók nagyon nagy érdeklődést mutatnak iránta.

Ezeknek a témáknak a feldolgozására vagy szakkörök formájában, vagy egyéni tanulás során van lehetőség.

Egyik ilyen téma az internetes oldalak és a szorosan hozzá kapcsolódó animációk létrehozása. Rengeteg szakkönyvet megnéztem az animációkészítéssel kapcsolatban és arra a következtetésre jutottam, hogy számtalan vastag könyvet kell átnézni ahhoz, hogy átfogó ismeretet szerezzünk.

Szakdolgozatomban egy oktatóprogramot szeretnék létrehozni, melyben a tanulók elsajátíthatják az egyszerű animációkészítés fortélyait a Macromedia Flash Mx program segítségével, nagy hangsúlyt fektetve a gyakorlatközpontú oktatásra.

Macromedia Flash Mx programot egyrészt azért választottam, mert nem szükséges megvenni, hanem létezik úgynevezett próbaverziója, mely az internetről könnyen letölthető. Másodsorban pedig azért, mert manapság nagyon elterjedt a használata, keresettek a Flash szakértők a webtervező cégek körében, legyen az teljes munkaidős vagy távmunkás állásajánlat.

A Flash Mx interaktív multimédiás anyagok, interaktív weboldalak és bannerek elkészítésére önmagában is alkalmas szoftver a webtervezők kezében, amely ráadásul olyan kisméretű állományokkal dolgozik, ami a szokásos médiaállományok töredéke.

Az oktatóprogramom témájának feldolgozásához szükséges idő egyénfüggő, hiszen a programot használók saját tempójukban tanulhatják a program használatát. Próbáltam

olyan gyakorlati példákat beleépíteni, amelyeket továbbgondolva folyamatosan felhasználhatók lehetnek egy weblap fejlesztésnél.

A tervezésnél fontosnak tartottam, hogy megfeleljen a szoftverergonómiai alapelveknek, vagyis könnyen kezelhető és főleg átlátható legyen egy felhasználói szintű tudással rendelkező tanuló számára is.

Az oktatóprogramot azoknak ajánlom, akik szeretnék megtanulni, hogy az internet oldalakon lévő animációk hogyan készülnek, esetleg a már meglévő oldalukat szeretnék érdekesebbé tenni egy jó kis animációval, vagy esetleg a későbbiekben ezzel akarnak egy kis zsebpénzt keresni.

Dolgozatom első részében tanácsot adok a hatékony oktatóprogramok kiválasztásához, hiszen manapság az oktatóprogramok piacán rengeteg található, de kevés olyan programmal találkozhatunk, amely mind tartalmi, mind pedagógiai szempontból is felhasználható lehetne a tanítási órák keretén belül.

Fontosnak tartom, hogy egy átfogó képet adjak a szoftverergonómiáról is melynek segítségével még hatékonyabb, és alkalmazhatóbb oktatóprogramot választhatunk, illetve készíthetünk.

Megvizsgáltam azokat az irányelveket is, melyek egy elektronikus kiadvány elkészítésekor célszerű figyelembe venni, azaz hogyan jelenik meg a tipográfia a monitoron.

A következő részében pedig a Macromedia Flash programot mutatom be, melynek segítségével készítettem el az oktatóprogramomat, amellyel ennek a programnak a használatát tanulhatjuk meg.

A dolgozat utolsó részében pedig az elkészített oktatóprogramomnak a tervezését, megvalósítását és használatát írom le részletesen.

Hatékony oktatóprogramok kiválasztása, elemzése

Az utóbbi két évtizedben robbanásszerű volt a számítógépek térhódítása, amelynek következtében jelentősen megváltozott az információs környezet, az emberek mindennapi információszerzése, tájékozódási lehetősége, információfeldolgozási szokásai.

Ennek következtében a tanítás-tanulás folyamat színterét már nemcsak a hagyományosnak mondható könyv, papír és ceruza, illetve a szemléltetésre használt audiovizuális eszközök biztosítják, hanem a multimédiás eszközök is, melyek segítségével igénybe vehetők a különböző oktatóprogramok. Ez a változás mind a pedagógusok, mind pedig a diákok számára új kihívást jelent.

A digitalizált információk könnyen egymásba illeszthetők. Így oldható meg, hogy a bemutató során egyszerre megjelenő képi és hanghatások egy időben több érzékszervre ható, és ezáltal könnyebben befogadható szemléltetést eredményeznek. Az on-line rendszerek révén pedig az adatok gyors kicserélése, aktualizálása is lehetséges. A pedagógus ezáltal testre szabhatja a már meglévő prezentációját, az oktatóprogramok pedig mindig alkalmazkodhatnak az új információkhoz. [9]

A programok kiválasztásának szempontjai

Az oktatóprogram kiválasztásánál figyelembe kell venni:[5]

- tanítási óra tananyagát:
 - nem minden tananyag és témakör elsajátításához célszerű számítógépet használni, de vannak, amelyekhez viszont kifejezetten ajánlott;
- célját:
 - amely szerint megkülönböztetünk új ismeretközlő, alkalmazó-gyakorló, összefoglaló, rendszerező, diagnosztizáló órátípusokat;
- a tanulásszervezési eljárásokat:
 - csoportmunka;
 - páros munka;
 - egyénre szabott (individualizált) munkaforma;
 - részben egyénre szabott szervezeti forma.

A korszerű oktatóprogramokkal szemben támasztott követelmények elemzése

Az elemzés során a következő alapkérdésekre kell megtalálni a választ:

Van-e értelme egyáltalán az adott feladat digitalizálásának? Ha igen, megfelel-e a megfogalmazott kritériumoknak?

A programozott oktatás kialakulásával megfogalmazódott az interaktív oktatás hármas kritériuma is, melyet röviden csak TAR-ciklusnak neveznek.

A rövidítés:

- Tanítsd meg az anyagot! (Teach)
- Mérd fel, hogy jól tanítottad-e meg, (Assess), illetve, hogy a diák megértette-e!
- A válasz (Respond), azaz a megtanulás stádiumának megfelelően irányítsd a hallgatót!

Mindezek tükrében leszögezhető, hogy a digitális taneszközök minősítésekor a következő 3 alapszempontra kell figyelembe venni:

- szakmai tartalom;
- pedagógiai tartalom;
- technológiai, esztétikai tartalom.

Szakmai tartalom

- illeszkedés a tantervi szempontokhoz;
- illeszkedés a célcsoporthoz: életkori sajátosságok, előképzettség;
- interdiszciplináris megközelítés (nem lineáris);
- médiumok aránya;
- strukturális felépítés: tartalmi, logikai;
- nyelvezete helyes és érthető;
- a tartalom dinamikus, igény szerinti megjelenítése, továbbfejleszthetősége;
- szerzői jogok figyelembevétele.

Pedagógiai tartalom

- meglévő és fejlesztendő képességeknek való megfelelés;
- kezelés nehézsége, és a célcsoport informatikai tudásszintjének szinkronja;
- tanórai-tanórán kívüli munka támogatása;
- motiváció megteremtése;

- tartalmaz-e méréseket (önellenőrzés, felmérő, szummatív,...);
- támogat-e többféle tanulási módot, dinamikus egyéni haladási útvonalat;
- kooperatív munka támogatására alkalmas-e;
- referenciák, gyakorlati tapasztalatok;
- van-e hozzá tanári felhasználói kézikönyv .

Technológiai, esztétikai tartalom

- on-line off-line lehetőségek;
- programozás nyelve, hordozhatósága, plug-in-ek, installáció, hibamentesség;
- interaktivitás foka, a megoldás alakíthatósága;
- multimédia elemek minősége, illeszkedése a tartalomhoz;
- modularitás, nyitottság, testreszabhatóság, rugalmasság;
- beépített tanulói segítségek: help, jegyzetelés, könyvjelzők;
- szoftver ergonomiai szempontok, képernyő információk elhelyezése, felhasználóbarát kialakítás;
- grafika, tipográfia, esztétika, színek használata, ikonok kialakítása – rendszere;
- navigáció típusa, testreszabhatósága, követhetősége;
- linkek ellenőrzése;
- kereső testreszabhatósága, a kapott listák kezelhetősége;
- kommunikációs lehetőségek, kooperatív munka támogatása;
- szabványoknak megfelelés (SCORM, LOM, IMS, AICC,...).

Mindezek figyelembe vételével, elemzésével a pedagógus megtalálhatja a számára legmegfelelőbb oktatói programot, illetve képes eldönteni, hogy a már meglévő oktatóprogram mennyire felel meg a tőle elvárható kritériumoknak. [5]

Szoftverergonómia

A szoftverergonómia célja a számítógéppel végzett munka egyszerűsítése, a feladatokra fordított költség és idő csökkentése. Egy szoftvert akkor használható jól, ha kevés erőfeszítést igényel használatának megtanulása, nem fárasztja el a felhasználót használat közben, magas teljesítmény elérést tesz lehetővé miközben a felhasználó elégedett a szoftverrel.

A szoftverergonómia egységes jellemzéséről még ma nem beszélhetünk, hiszen nagyon szerteágazó vizsgálati szempontok alakultak ki. Régebben, míg a szoftver kezelőfelületét vizsgálták, napjainkban egyre inkább a használhatóság került előtérbe.

Szoftverergonómiai követelmények oktatóprogram esetében:

A szoftver feleljen meg a kitűzött célnak, feladatnak

Ez azt jelenti, hogy mindazt, amit a tervezés előtt kitűztünk megvalósítandó célnak az valósuljon meg, mindezek mellett a felhasználó számára érthetőek és átláthatóak a megoldandó feladatok. Ha ez megvalósul, akkor elmondható, hogy a szoftver segíti, és nem hátráltatja a feladat elvégzését.

A program legyen könnyen megtanulható

Minden felhasználó arra alapoz, amit eddig megtanult a különböző szoftverekből. Fontos, hogy azokat az alapvető tulajdonságokat jelenjenek meg a programokban, ami a többi, esetleg régebbi verziókban voltak. Gondolva itt a például a piros szín mint veszély, nyíl mint irányt jelző stb. A összetettség csökkentése a legjobb tanulást könnyítő sajátossága egy programnak. A program átláthatósága nélkül nem sajátítható el könnyen az együtműködés.

A program legyen könnyen irányítható

A felhasználó a szoftvert úgy tudja használni, hogy saját ütemében tudjon haladni, ha szükséges, tudjon megállni, és ha kész, akkor haladjon csak tovább.

A program adjon viszonylag nagy szerepet a felhasználó választásainak. Ide tartozik a logikus menü felépítése és érthető tartalmi tagolása, egyértelműsége.

A program feleljen meg az elvárásoknak

A szoftverhasználat során kialakul egy elképzelés a felhasználóban, hogy az egyes műveletek milyen eredményekhez kell juttassa. Törekedni kell arra hogy ezt a felhasználó hosszabb – rövidebb időn felül elérhesse, ha szükséges a művelet lépéseit megismételhesse, újra feldolgozhassa. Megfelelő gyakorlás után önállóan is képes legyen végrehajtani a feladatot.

A program legyen hibaálló

Egy program esetében a legjobb, ha nincs lehetősége a felhasználónak hibázni. Ez azonban szinte megoldhatatlan feladat. Mindazonáltal szükséges, hogy a felmerülő hibák számát a minimálisra csökkentsük.

A program legyen átlátható

A képernyőn megjelenő információk olvashatóságát tekintve jobb, ha a terjedelme kisebb, mint a nyomtatott oldal esetében. A képernyőnek maximálisan 50%-át szabad igénybe venni a programnak. [11]

A menüfák inkább szélesek, mint mélyek legyenek, fontos hogy a menü szélessége ne haladja meg a képernyő nagyságát egy almenüben pedig 10-12 pontnál több ne legyen. Világosan legyenek megkülönböztetve azok a funkciók, amelyek rendelkezésre állnak, és azok amik nem (pl. színhatással vagy hanghatással).

Mivel a felhasználók hajlamosak egy programot annak külső kinézete alapján megítélni, így fontos az is, hogy milyen betűtípust választunk.

Egyes típusok kimondottan jól mutatnak a képernyőn és könnyen olvashatóak is, míg egy másik kimondottan zavaró lehet.

A monitoron megfelelően használt színek lényegesen javíthatják az átláthatóságot, de a rosszul kiválasztott színek teljesen elronthatják a program használatát. A helyes színek megválasztása sokkal nehezebb, mint látszik. Gyakran találkozni az interneten olyan oldalakkal, ami a háttér és a betűszín kiválasztása miatt élvezhetetlen. Az ellenszíneket: vöröset zölddel, illetve sárgát késsel nem célszerű előtér-háttér párosításban alkalmazni, mert a szemmozgások miatt fellépő komplementer utókép a közöttük lévő határt elmosódottá teszi.[10]

Előtér színe	Háttér színe
sárga	fekete
fehér	kék
fekete	narancs
fekete	sárga
narancs	fekete
fehér	fehér
fehér	vörös
vörös	fehér
zöld	fehér
narancs	fehér

Irányelvek az elektronikus kiadványok készítéséhez- tipográfia a monitoron

A tipográfia a betűk, a szöveg beállításának a művészete. Magába foglalja a szöveg elhelyezését, méretét, tagolását, betűtípusát és minden egyéb paraméter kezelését, ami a szöveg megjelenítéséhez kapcsolódik.[1]

A nyomdai tipográfia története Gutenbergig és a nyomtatásban használható mozgatható betűk feltalálásáig nyúlik vissza, de az írásképek kialakítása már évezredekkel ezelőtt külön művészeti és tudományos feladatot adott az embereknek.

A tipográfia alapvető céljai

- felkelti a figyelmet;
- értéket ad;
- javítja az olvashatóságot;
- egyszerűsít;
- kiemeli a fontos részeket;
- rendszerez;
- egységet teremt;
- részekre tagol;

Az elektronikus kiadvány, mivel többnyire nem a szokványos papír alapon keresztül prezentálja a tartalmat, rendelkezik jó pár speciális tulajdonsággal, ami miatt másképpen kell kezelnünk a monitorra szánt tartalmat, mint a nyomtatásra szánt tartalmat.

A papíralapú és a elektronikus kiadvány megjelenítési különbségek: [2]

Jellemző	Papír	Elektronikus kiadvány
betűméretek	fix	nem fix
szín	fix	nem fix
betűtípus	fix	nem fix
elhelyezések	fix	nem fix

Elektronikus kiadványok tipográfia tervezése

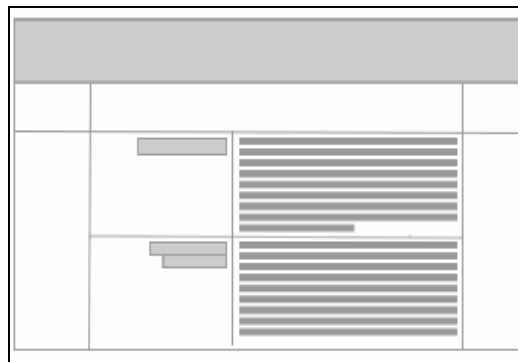
Mielőtt nekilátnánk a kiadvány konkrét megvalósításához, itt is nagyon fontos a tervek elkészítése. A tervezés segít rendszerezni tevékenységünket, iránymutatást ad a cselekvéshez. A jól elkészített tervvel sok időt és energiát spórolhatunk meg a fejlesztés során. Az elektronikus kiadványunk elkészültekor, a tervekhez viszonyítva tudjuk eldönteni, hogy munkánk során mennyire jártunk sikerrel, mivel lehetünk elégedettek és mi az, amin még javítani kell.

A tervezés során követendő lépések

- célkitűzés
- formátum meghatározása
- színek meghatározása
- megfelelő betűtípus és betűméret meghatározása
- képek
- mitől lesz lendülete?
- javítás, finomítás
- ellenőrzés
- tesztelés

Képernyő tervének meghatározása

A tervezésénél ez praktikusán egy táblázatot jelenthet, melynek segítségével az oldalon található fejléc, menüsor helyét meghatározhatjuk, illetve beállíthatjuk a szöveg szélességét. A kialakításánál célszerű lehet a képernyő felbontásának figyelembevétele (pl. 800x600, 1024x768), azaz az oldal magasságának és szélességének aránya. Az átláthatóbb tervezés és praktikusabb megvalósítás érdekében célszerű több, egymásba ágyazott táblázatot is használni.



A hangsúlyosabb részt helyezük el a bal felső sarokban, hiszen több kísérlet is bebizonyította, hogy a szemünk egy oldalon először azt pillantja meg. A következő két képen a szemünk mozgásának útvonalát és a megjelenítés utáni első pillantásokat szemlélteti.



Betűtípus és betűméret meghatározása

Amikor képernyőre tervezünk egy szöveget, rendszerint annak reményében tesszük ezt, hogy valaki el is olvassa. Amint megnehezíti valami az olvasását, elfárad, vagy nem sikerül fenntartani az érdeklődését, azonnal abbahagyja és motiválatlan lesz. Hogy ez ne következzen be, célszerű néhány szabályt betartani. Ilyen például, hogy ne írjunk csupa nagybetűvel.

A kisbetűk egyedi alakja megkönnyíti az olvasást, a törzsszöveget szedjük mindig kisbetűvel. Arról nem is beszélve, hogy a nagybetűvel szedett szöveg úgy hat, mintha torkunk szakadtából ÜVÖLTÖZNÉNK!



Törekednünk kell, hogy a szöveg jól látható és olvasható legyen. A monitoron jól olvasható betűtípusok a talpatlan betűk. Betűinek szára szigorúan egyenes, semmiféle talpban nem végződik. Ilyen például az Arial; Tahoma; Impact; Helvetica; stb.



Soha ne használjunk Serif betűtípust pl. Times New Roman, Garamond, Bookman. Kerüljük a túldíszített vagy nagyon különleges betűtípust, szűk betűtípust, árnyékolást, és fontos, hogy minél kevesebb dőlt betűt és aláhúzást alkalmazzunk.

Bizonyos szövegeknél szükség van a kulcsszavak, vagy fontos szövegrészek kiemelésére. Ezzel jelentősen megkönnyíthetjük az olvasó dolgát is az új ismeretek rendszerezésében. A fontos részeket kiemelhetjük **Kövéritéssel**, vagy *döntéssel*. Sok esetben **b e t ű k ö z z e l** is elérhetjük a kívánt eredményt. [1]

A betűfokozatok méreteit tipográfiai pontban adjuk meg. A pont a francia F. Didot és a német H. Berthold által kidolgozott tipográfiai mértékegység (a méter 2660-ad része, kb. 0,376 mm). A legkisebb betűméret az egyponthoz. Az általánosan elfogadott betűméret a 12 pont. Elektronikus kiadványok esetében a szakirodalom nem igazán tesz említést a betűmérettel kapcsolatban hiszen, a képernyő felbontásával az változhat.

Színek használata a monitoron

Az elektronikus kiadványok egyik legnagyobb előnye, hogy különböző színek előállítása nem kerül külön költségbe. A színekkel hangulatot közvetíthetünk az olvasó felé: feszültséget, izgatottságot vagy esetleg nyugalmat, boldogságérzetet.

Megpróbáltam néhány olyan színt és jelentést összegyűjteni, amelyet sűrűn használnak elektronikus kiadványokban.

Fekete:

- Az ősi sötétség, a káosz, az éjszaka, a feneketlen, titokzatos mélység, a világűr jelképe.
- A színekben kifejeződő élet ellentéte, a halál szimbóluma.
- A lemondás színe, a gyász kifejezője; számos hagyományban a gonosz erőkhöz kötődik.
- Mélység érzetét kelti, de egy fekete háttér csökkenti az olvashatóságot!

Fehér

- A fény, a tisztaság, a tökéletesség, az egyszerűség jelképe.
- Leginkább a tökéletesség színéhez társítják.
- A feketével ellentétben leginkább pozitív jelentése van.
- Sikeres kezdetet is jelenthet.

Kék

- Az égbolt, a tenger színe, a megfoghatatlanság, a végtelenség, szimbóluma.
- A kékről gyakran a biztonságra és a stabilitásra gondolunk.
- Ez a szín szimbolizálja a reményt, a hűséget, a bölcsességet, a bizalmat, az intelligenciát, a becsületet, és az igazságot.

Bíbor

- Ez szimbolizálja az erőt, a nemességet, a luxust és a becsvágyat.
- Ez a szín a világosságot és a pazarlás érzését kelti.
- A bíborról gondolhatunk a bölcsességre, a méltóságra, a függetlenségre, a kreativitásra, a titokzatosságra és a mágiára.
- A világos bíbor jó választás női designhoz.
- Az élénk bíbor gyermek közönségnek szóló termékeknél ideális.
- A világos bíbor romantikus és nosztalgikus, ezzel szemben a sötét bíbornak komor, gyászos és szomorú hangulata van. Ezzel csalódottságot tudunk kifejezni.

Zöld

- A zöld a természet színe.
- Ez szimbolizálja fejlődést, harmóniát, szemtelenséget, és a termékenységet.
- Erős érzelmi összefüggés van a zöld szín és a biztonság között.
- Nagyszerű nyugtató ereje van, a legpihentetőbb szín az emberi szem számára, javítja a látványt.
- A sötétzöldről a becsvágyra, a kapzsiságra és a féltékenységre gondolunk.
- A zöldessárga a betegségre, a gyávaságra, a viszályra és a féltékenységre utal.
- Kiegészítő színe a piros.

Sárga

- A sárga a napsütés színe.
- Az örömmel, a boldogsággal, az értelemmel és az energiával szokás összekapcsolni.
- A sárga meleg hatást, boldogságot kelt, elősegíti a mentális aktivitást, és erőt ad.
- Világos, igazi figyelemfelkeltő.
- Ha túl sokat használják, már zavaró is lehet.
- Kitűnő például gyermekeknek szánt termékek reklámozására.
- A sárga egy nagyon célravezető szín, így a legfontosabb információk kiemelésére alkalmas.
- Megbízhatatlan és spontán szín.

Narancssárga

- A narancssárga ötvözi a piros által sugalmazott energiát és a sárga boldogságát.
- Általában az örömmel, napfényel és a trópusokkal kötik össze.
- A narancssárga a lelkesedést, az elragadtatást, a boldogságot, a kreativitást, az elhatározottságot, a vonzerőt, a sikert, a bátorítást, és az izgatást fejezi ki.
- Az emberi szem számára a narancssárga egy nagyon meleg szín, így a forróság érzetét kelti, de nem olyan agresszívan, mint a piros.
- A narancssárga növeli az agyba jutó oxigén mennyiségét, élénkítő hatású és növeli a szellemi aktivitást.
- Nagyon elfogadott a fiatalok körében.
- Erről a színről az egészséges étel jut eszünkbe és az étvágyunkat serkenti.
- Az ősz és az aratás színe.

Egy elektronikus kiadvány elkészítése során az egyik legnehezebb feladat a megfelelő színek kiválasztása.

A megfelelő színek kiválasztása sokkal nehezebb probléma, mint valójában látszik. Gondoljunk csak arra, hogy vannak bizonyos színkombinációk, amelyeket együtt használva a szemünk gyorsan elfárad, vagy gondoljunk azokra az emberekre, akik színtévesztők. Bizony, ha úgy választunk színeket egymás mellé, hogy azok között kicsi a színkontraszt, akkor nagy valószínűséggel ezek az emberek nem fognak semmit látni a szövegből, hisz az beleolvad a háttérbe.

A színeket és az árnyalatokat már régóta felosztották elsődleges, másodlagos és harmadlagos színekre. Az elsődleges színek a piros, a sárga és a kék, amelyek azért elsődlegesek, mert az előállításukhoz nem szükséges színeket összekeverni. A másodlagos színek az elsődleges színek keverékei, mégpedig a következők: piros + sárga = narancssárga; sárga + kék = zöld; kék + piros = lila. A harmadlagos színek már a másodlagos színek keverékei, és az elsődleges és másodlagos színek között helyezkednek el.



A következőkben bemutatok néhány színsémát, melyet használva garantált a jó háttér és szövegszín választás, alapját pedig a színekör adja.[3]

Monokróm színséma

Egy monokróm színsémát egyetlen színből, és annak világos és sötét árnyalataiból állítjuk elő. Bár ezt a színsémát elég egyszerű használni, a legtöbb webdesigner nem lelkesedik érte.



Tiszta és könnyed hatású, az egymáshoz illő színek nyugtató hatásúak.

Különösen a kék, illetve zöld árnyalatokra jó ránézni. *Előnye:* könnyen elkészíthető, mindig kiegyensúlyozottságot sugall. *Hátránya:* fontos tartalom kiemelése nehéz.[4]

Például:



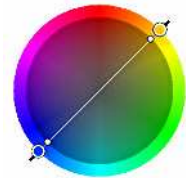
Analóg színséma:

A színekörön egymással szomszédos színeket használunk, ahol a domináns színt a többi erősíti. Hasonlít a monokróm színsémához, de több árnyalatot enged. Óvakodjunk a hideg és a meleg színek együttes használatától! *Előnye:* egyszerűen elkészíthető. *Hátránya:* a szíkontrasztok hiányoznak. [4] Például:



Kiegészítő színséma

A kiegészítő színsémában, mindig a színekerek ellentétes oldalán található színeket párosítjuk össze. Ez a legjobb választás, ha hideg és meleg színeket szeretnénk együtt használni (pl. piros-kék). *Előnye:* erősebb ellentét, figyelemfelkeltő. *Hátránya:* nehéz az egyensúlyt megtartani, főleg ha kevésbé telített meleg színeket használunk. [4] Például:



Hármas felosztás

Három, a színekörön egymástól egyenlő távolságra elhelyezkedő színeket használ. Kedvelt a művészek körében, mert erős kontrasztot ad az egyensúly és a színgazdagság megtartása mellett. Akkor igazán hatásos, ha egy színt dominánssá választva nagyobb arányban használjuk, mint a többi. *Előnye:* a harmónia megtartása mellett erős kontrasztot nyújt. *Hátránya:* nem nagyon kiemelhető. [4] Például:



Négyes felosztás

Négy színt használ, két kiegészítő párba osztva. Ha mind a négy színt egyenlő arányban használjuk, akkor nehéz harmonikus képet előállítani, válasszunk inkább egy domináns színt, amit a legtöbb helyen használunk. Ha nincs egyensúlyban az összkép, próbáljunk kevésbé élénk színt választani. *Előnye:* több színvariáció. *Hátránya:* négy szín már nehezen hozható egyensúlyba. [4] Például:



A következő két webcím esetén az elsőben olyan program tölthető le, mely segítségével könnyen kipróbálhatjuk a színsémákat.

A második link pedig egy webes alkalmazás, melyben szintén az egyes sémákon kísérletezhetünk, hogy megfelelő színeket találjunk.

<http://www.color-wheel-pro.com/>

<http://colorschemedesigner.com/>

Képek használat egy elektronikus kiadványban

Egy kép többet mondhat ezer szónál. A képekkel nagyszerűen lehet befolyásolni egy oldal hangulatát, és az illusztrációkon keresztül könnyen be lehet mutatni komplex információkat is az olvasónak, de ezeknek is megvannak a sajátos szabályai:

- Ügyeljünk arra, hogy a fényképen található arc ne nézzen ki a képből.
- Egy keret nélküli kép modern hatást kelthet.
- Egy kerettel hangsúlyosabbá tehetjük képeinket.
- Fontos, hogy a kép színei hangulatban legyenek az oldal színével.
- Ügyeljünk arra is, hogy a képnek ne legyen túl sok holttere.

Flash keretrendszere

Flash története

1996-ban egy FutureSplash nevű cég mutatta be a Flash-t. A technológia egy webes multimédia megoldásként keletkezett, amivel könnyen, gyorsan letölthető interaktív gombokat készíthettünk az internetre.

Amivel igazán elbűvölt mindenkit az az, hogy vektorgrafikát lehet vele készíteni. A Macromedia továbbfejlesztette a technikát, és a Flash-t bitképes, audio támogatással bővítette. A Flash 3-as verziójánál jelent meg az átlátszóság (Transparency) és a morfózis (Shape Morphing) képessége.

A Macromedia Flash 4.0 megjelenése a web fejlődésében újabb nagy mérföldkő. A Flash ezen túl már nemcsak az egyik legjobb webes animációs megoldás, hanem az egyik legjobb a webes multimédiafejlesztéshez. A 4-es verzió újdonságai: MP3 audio, szerkeszthető szöveges mezők, újratervezett Library (könyvtár) ablak, bővített grafikus felület.

Az Flash 5.0 új tulajdonságai megnövelt szolgáltatást nyújtanak a különböző grafikák készítéséhez, a munkafolyamatokat egyszerűsítették és bővítették az ActionScript felhasználásával készíthető interaktív munkák lehetőségeit. Az 5-ös verzió főbb újdonságai: kibővített color controls (színbeállítás) ablak, új kiemelő színek alkalmazása a jobb megkülönböztetés miatt, saját gyorsbillentyűk definiálásának lehetősége a kényelmes, gyors munka érdekében.

A következő változat az MX jelzést viseli. Kezelőfelülete sokkal átgondoltabb felépítésű, mint a korábbi változatoké, s nem utolsó sorban testre szabható. Használata könnyebben megtanulható, dolgozni pedig kész öröm vele.

A Flash 8 megjelenésével mind a grafikai tervezés, mind a programozás területén jelentős fejlesztéseknek lehettünk tanúi.

2006-ban a Macromedia az Adobe tulajdonába került. Az Adobe és a Macromedia egyesülésének oka a jövőbeli növekedési lehetőségek kihasználása, nem pedig a költségek csökkentése volt. 2007 tavaszán pedig megjelent immár a kilencedik verzió, az Adobe Flash CS3 Professional. Egyik legfontosabb újdonsága a teljes integrálhatóság

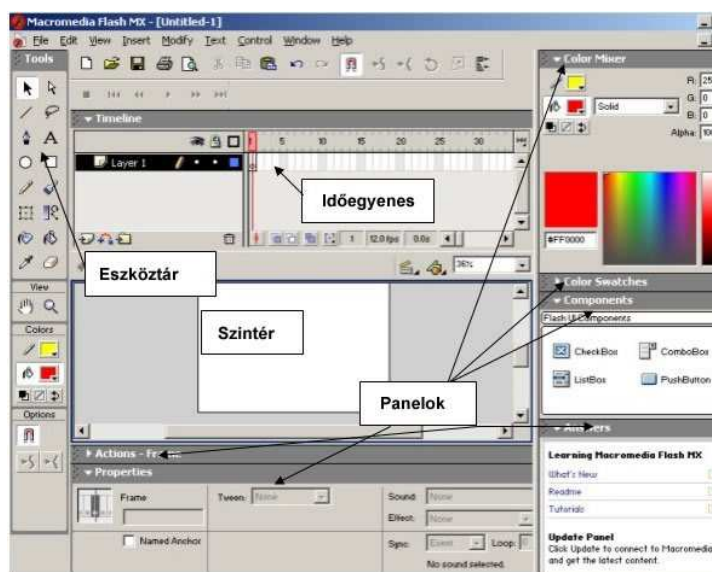
az Adobe és Macromedia szoftverek között. Az ActionScript 3.0 egy új struktúrájú környezetet kapott így még könnyebben, és hatékonyabban fejleszthetünk. Új eszközökkel bővült az eszköztára. Egyszerűsödött a médiavezérlők és a navigációs nyomógombok kezeléséhez szükséges szkriptek megírása. A viselkedéstípusokat a fejlesztő saját maga is megírhatja, de akár le is töltheti azokat az Adobe oldaláról.[16]

Macromedia Flash Mx bemutatása

Jelen dolgozatomban azért választottam a Flash változatok közül az Mx-et, mert a Hunyadi János Gimnáziumban, ahol dolgozom, ez a típus áll rendelkezésünkre, hogy megismertessük a tanulókkal az animációkészítés rejtélyeit.

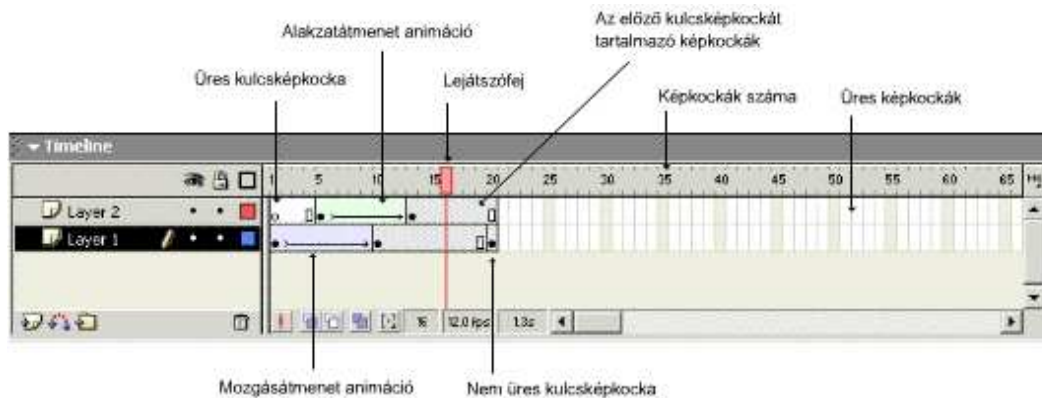
A Flash-t elsősorban webes környezetben használják reklámok, bannerek készítésére, mivel kis mérete ellenére nagyon látványos animációkat lehet vele létrehozni. Apró méretét annak köszönheti, hogy vektorgrafikus leírást használ, amely nem az összes pixelét írja le egy alakzatnak, hanem csak a matematikai függvényeit. A másik előnye a vektorgrafikus leírásnak, hogy nagyításnál, kicsinyítésnél nem torzul a kép, ezzel megtartva a minőségét. Manapság a Flash Player az internetre csatlakozott számítógépek nagy részén megtalálható, mely segítségével lejátszhatók a webbe ágyazott Flash animációk. Másik nagy előnye még, hogy gyakorlatilag böngészőfüggetlen. [15]

Munkaablak felépítése



Az **eszköztáron** (Tools) található elemek között megtalálhatók olyan ikonok, amelyeket más grafikai rajzolóprogramból is ismerhetünk, (ceruza, radír, kijelölő eszközök stb.). Különböző rajzelemek készíthetők és szerkeszthetők segítségével.

Az **időegyenes** (Timeline) melynek az egyik részén a rétegek (Layer-ek) találhatóak, a másik pedig a hozzájuk tartozó Frame-ek (képkockák), ezekkel az animáció működését irányíthatjuk és meghatározhatjuk, hogy mi jelenjen meg a munkaterületen. A piros csuszka (lejátszófej) a függőleges vonallal azt a filmkockát jelzi, ami éppen a színtéren megjelenik. Új elemeket úgy tudunk hozzáadni a flash mozihoz, hogy a időegyenes bal oldali oszlopába új réteget hozunk létre. A mozi időtartamát is meghosszabbíthatjuk úgy, hogy a jobb szélén új képkockákat adunk hozzá. Bármikor létrehozhatunk új képkockákat és rétegeket, áthelyezhetjük őket, vagy megváltoztathatjuk a tulajdonságaikat. Hozzáadhatunk a időegyeneshez olyan speciális képkockákat is, amikkel a moziban lévő eseményeket és átmeneteket szabályozhatjuk. Ezeket kulcskockáknak hívjuk.



A fehér téglalapot, ami kitölti a képernyő közepét **színtérnek** (Stage) hívjuk. Ez a terület az, ahová rajzolhatunk, és ahol megtekinthetjük az animációt, amit készítünk.

Színtér alatt és tőle jobbra lévő, ki-be kapcsolható **panelek** számos beállítási lehetőséggel szolgálnak a vonalszíntől a betűtípusokon át az objektumokhoz rendelt ActionScript funkciókig.[14]

Animáció típusai

Az animáció beállított figurákról készített állóképek, vagy rajzolt képek sorozata, amely megfelelő sebességgel lejátszva mozgás hatását kelti.

Típusai:

- Állandó előtérrel készülő animáció: az előtér változatlan, miközben a háttér változik. A háttér gyors változtatása folyamatos mozgás látványát kelti.

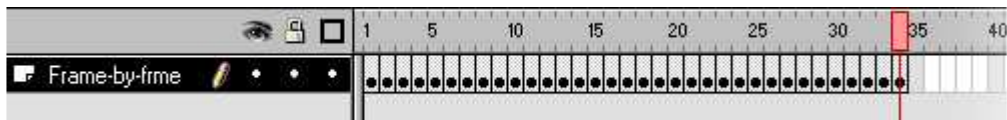
- Állandó háttérrel készülő animáció: egyszerűbb állandó hátteret az animáció minden egyes lapjára felvenni, és utólag rárajzolni a változó előteret. Az így készülő animáció könnyebben és gyorsabban hozható létre.
- Objektum animáció: egy előre elkészített grafikus objektumot mozgat a képernyőn egy előre megadott útvonalon. Az animáció hatásának fokozása érdekében a mozgatás közben az objektummal bizonyos műveletek végezhetők, de az objektum nem változhat.[13]

Animációs technikák a Flashben:

Képkockáról képkockára:

Ez az eljárás még a rajzfilmkészítés eső napjaira nyúlik vissza. Ekkor még minden egyes képkockát külön megrajzoltak, és az ezeken lévő apró változások eredményezték lejátszáskor a mozgás látszatát. A Flash-ben ezt **Frame-By-Frame Animation (képről képre animációnak)** nevezzük.

Ha ilyen technikát szeretnénk alkalmazni a Flash-ben, akkor minden egyes képkocka, amin változás van kulcs képkockaként fog megjelenni. Tehát első lépésként készíteni kell egy kulcs képkockákból álló folyamatot, mely tartalmazza a mozgatni kívánt elemeket. Ezután képkockáról képkockára el kell elkészíteni a rajzokat, valamint azokon a kívánt változtatásokat.[11]

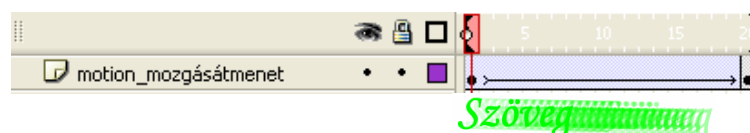


Mozgásátmenet

A mozgásátmenet szó a hagyományos animációs terminológiából származik. A mozgásátmenet (más néven közbeszúrás) lényegesen lerövidíti az animációs effektusok létrehozási idejét.

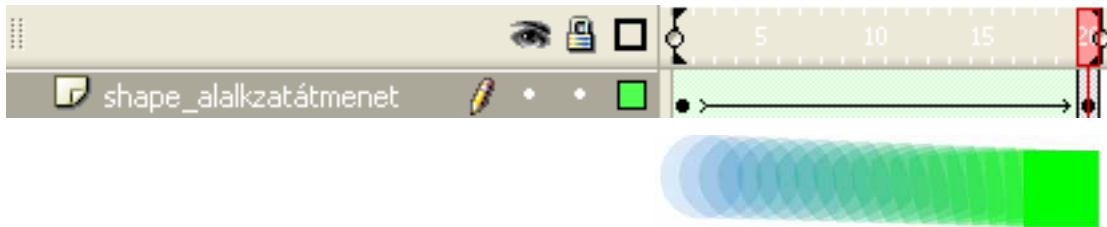
A mozgásátmenetekkel csoportokat, szimbólumokat és szerkeszthető szövegeket hozhatunk mozgásba.

Elsősorban arra használjuk, hogy egy elemet átvigyünk egyik helyről a másikra, megváltoztathatjuk az elemek méretét, dőlését, elforgatását.



Alakzatátmenet

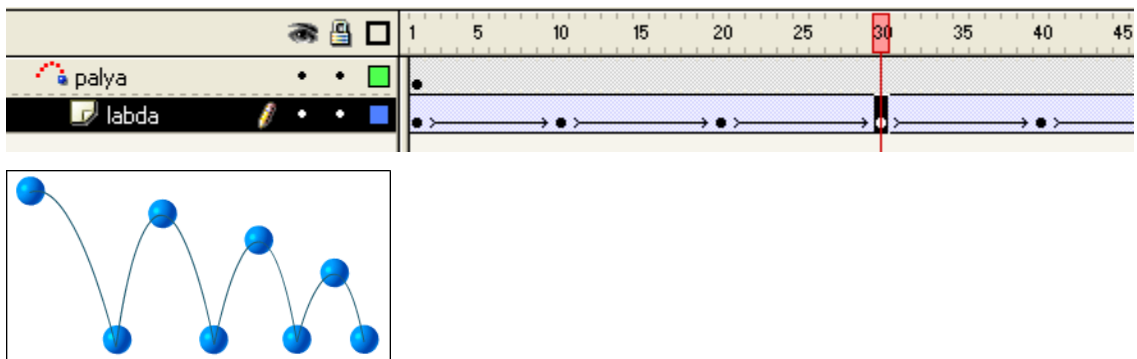
Az alakzatátmenetnek akkor vehetjük hasznát, ha egy alakzatot fokozatosan egy másik alakzatba szeretnénk átvinni. Kizárólag ilyenkor használható tehát nem alkalmas csoportok, szimbólumok egymásba átvitelére. Egy rétegen belül több alakzatra is kiterjeszhető. Segítségével az alakzat színe is megváltoztatható.



Mozgásvezetők

A mozgásátmenet segítségével az elemeket csak egy egyenes mentén tudjuk mozgatni. A Flash lehetőséget ad arra is, hogy mi rajzoljuk meg az útvonalat.

Amennyiben a mozgásátmenetet egy általunk megrajzolt útvonalhoz szeretnénk igazítani, szükségünk van egy mozgásvezető rétegre. A vezető réteghez egy vagy több olyan vezetett réteg tartozhat, amelyek követik a megadott útvonalat. A vezetőréteg nem marad meg a filmben, kizárólag csak a szerkesztőben látszik.[6]



Szimbólumok

A *szimbólum* olyan grafika, gomb vagy filmklip, amelyet ha egyszer létrehoztunk Flash-ben, számtalanszor felhasználhatunk dokumentumunkban vagy más projektekben. Ezek alkalmazásával sok animáció esetében csökkenthető a file méret és a letöltési idő. Akárhányszor használjuk a szimbólumot a projektben, a Flash csak egyszer építi be az adatait. A szimbólumokat a Library panel tárolja. Amikor szimbólumot húzunk a Színpadra, az alkalmazás annak egy példányát hozza létre, az eredeti (az úgynevezett összszimbólum) a könyvtárban marad. A példány az összszimbólumnak a Színpadon elhelyezkedő vagy más szimbólumba beágyazott másolata. Háromféle szimbólummal dolgozhatunk a Flash-ben, mindegyiknek vannak előnyei és korlátai.

Filmklip szimbólum

Sok tervező szinte kizárólag filmklip szimbólumot használ a különféle Flash objektumokhoz, mivel ez a legrugalmasabb szimbólumtípus. Rendelhetünk hozzá szűrőket, színbeállításokat és összhatásmódokat. Egy filmklip szimbólum minden példányát elnevezhetjük, és ActionScript programozással vezérelhetjük azokat. Hasonlóan rugalmasak a filmklip időegyenesek. Tartalmazhatnak hangot, interaktív vezérlőket, sőt akár más filmklippéldányokat is.[8]

Gombszimbólum

Használatával gyorsan készíthetünk szabványos rollover állapotokkal rendelkező, interaktív gombokat. A filmklip időegyeneseikkel ellentétben a gomb szimbólum időegyenésének négy kiemelt képkockája van: Up (Fent), Over (Föle), Down (Lent) és Hit (Találat)

- Az Up az alapértelmezett állapot.
- Az Over állapot határozza meg, hogy hogyan mutat a gomb, amikor az egérmutató fölé kerül.
- A Down állapot azt mutatja meg, hogyan fest a gomb, amikor rákattintunk.
- A Hit állapot a gomb érzékeny, az egérrel való kattintásra reagáló területét határozza meg.

Az egyes gombpéldányokat elnevezhetjük, és ActionScript segítségével vezérelhetjük őket. Szűrőket, összhatásmódokat és színbeállításokat rendelhetünk a gomb szimbólumokhoz.[6]

Grafikai szimbólum

A grafika típusú szimbólumokat a fő időegyeneshöz kötött statikus – vagyis nem animálandó – képekhez használjuk. A grafikák a legkevésbé rugalmas szimbólumok. Nem támogatják az ActionScript programozást, és – bár grafika szimbólumot animálhatunk – az interaktív vezérlők és a hang nem fog működni. Ráadásul szűrőket és összhatásmódokat sem rendelhetünk hozzájuk.

ActionScript a Flash-ben

A Flash figyelemreméltó animációs lehetőségekkel rendelkezik, de a haladóbb Flash-felhasználási módok egyik legfontosabb része az interaktivitás. Az ActionScript, a Flash scriptnyelve, egy objektumorientált nyelv, mely az ECMA-262 előírásokra épült,

ahogyan a JavaScript is. A két nyelv között csak kis különbség van. Ha valaki ismeri a JavaScriptet, az ActionScriptet is könnyen megérti. A Flash scriptnyelvének segítségével intelligenciával vétezhethük fel az objektumokat, hogy a felhasználó tevékenységére reagálni tudjanak.[17]

A Flash file típusai

Forrásfile (.fla)

Munkánk elmentésére a Flash a .fla file formátumot használja. Bármely .fla file-t megnyithatjuk és szerkeszthetjük, feltéve, ha rendelkezünk a Flash programmal. Ez a forrásfile az, amellyel bármely más formátumú file előállítható, viszont ezt semmi nem pótolhatja.

Terjesztett file (.swf)

Amikor a forrásfile szerkesztése befejeződött, és készen állunk az alkotás terjesztésére egyszerűen mentjük azt .swf file formátumba. Ezt bárki megnézheti, aki böngészővel és Flash Player bővítménnyel rendelkezik. A közönség az .swf file-t nem tudja szerkeszteni csak megtekinteni.[7]

Oktatóprogram tervezése

Mindenek előtt, ha oktatóprogramot akarunk készíteni, nem árt egy tervet létrehozni, ami alapján dolgozni fogunk. Erre azért van szükség, hogy már az első pillanatokban letisztuljon, mit is szeretnénk megvalósítani. Nem utolsó szempontként fontos a létrehozása, hiszen a konkrét megvalósítás során időt takaríthatunk meg, ha nem azon kell gondolkodjunk, hogy mit is szeretnénk célként elérni.

Egy oktatóprogram létrehozásánál két fő dolgot kell megtervezni, mielőtt nekivágunk a konkrét megvalósításnak. Az egyik a tartalmi része, a másik pedig a kinézet vagy a mostanság divatos kifejezéssel élve a „design”kialakítása.

Fordítsunk elegendő időt a terv elkészítésére, mert a későbbi munka során ez kifizetődik. Minél részletesebben dolgozzuk ki, annál könnyebb lesz a munka, és jobb lesz a végeredmény is.

Tartalmi rész tervezése

A tanári pályafutásom során már 5 éve van alkalmam Flash-t oktatni a tanulóknak. Ennyi idő alatt kialakultak olyan témakörök, amiket fontosnak tartok, hogy a tanulók megértsék, és könnyűszerrel megoldják a hozzá tartozó gyakorlati példákat, hiszen minden Flash alkalmazásnak azok az alapjai.

Az évek alatt összegyűjtött feladatok és leírások közül válogattam ki azokat, amiket elengedhetetlenül fontosnak tartok, hogy bekerüljön, egy kezdő szintű Flash oktatóprogramba, ugyanakkor képes legyen a tanuló az oktatóprogram befejezésével saját maga is tovább fejlődni. A programom célja az, hogy olyan tananyagot öleljen fel, ami szükséges, ahhoz hogy a Macromedia Flash Mx programot a felhasználó készségszinten használni tudja, és olyan animációkat hozzon létre, amit fel tud használni egy weblap készítésénél is, esetleg a feladatokat tovább csinosítva, fejlesztve be tudjon építeni egy már meglévő weblapba, esetleg egy teljes Flash alapú weblapot tudjon létrehozni. Fontosnak tartom, hogy miután végez a tananyagok feldolgozásával, motiválva legyen további fejlődésre, amihez segítséget is adok néhány olyan webes oldal linkjének megadásával, ahol új ismeretekre tehet szert.

A feldolgozásra váró témaköröket egy táblázatban foglaltam össze, melyben meghatároztam a menüpontokat, annak almenüpontjait és a hozzájuk tartozó rövid ismertetőt.

Témakörök az oktatóprogramban	
Bevezető	
<i>Történet</i>	
	rövid ismertető a Flash verziók változásairól;
<i>Jellemzők</i>	
	gyorsaság; interaktivitás; sokoldalúság; standard;
<i>File formátum</i>	
	.fla; .swf;
Munkafelület	
<i>Ablak felépítése</i>	
	ablak képe és annak részeinek szakszerű elnevezése;
<i>Kijelölő eszközök</i>	
	az eszköztáron lévő kijelölő eszközök használata, lehetőségei;
<i>Rajzeszközök</i>	
	az eszköztáron lévő rajz eszközök használata, lehetőségei (Nyíl rajzolás);
<i>Transzformálás</i>	
	az eszköztáron lévő transzformáló eszközök használata, lehetőségei (delfin rajzolása);
<i>Navigálás</i>	
	munkafelületen történő navigálás eszközei;
<i>Színkeverő</i>	
	Hogyan tudok létrehozni egyéni színeket, ha ismerem az RGB színkódot? Hogyan hozhatok létre színátmeneteket?
<i>Csoportosítás</i>	
	Hogyan tudom az összetett alakzataimat könnyebben kezelni, mozgatni? CTRL+G; funkciói;
<i>Rétegek</i>	
	rétegek használata; rétegtípusok; rétegműveletek; készítsünk háttérrel a delfinnek, néhány feladat amelyet eddigi ismereteinkkel létre tudunk hozni;
<i>Elrendezés</i>	
	align panel bemutatása, lehetőségei;
<i>Éllágyítás</i>	
	Hogyan tudjuk alakzataink határozott éleit éllágyítani? Hogyan tudunk árnyékot adni egy alakzatnak?

	Hogyan lehet fényudvart létrehozni az alakzatok köré?
<i>Szöveg eszköz</i>	
	szövegdoboz tulajdonságai, karakter formázási lehetőségek; szövegdoboz típusai: statikus, dinamikus, beviteli; szöveg darabolása: karakterekre, bitképpé; FLASH MX felírat létrehozása;
<i>Kép importálás</i>	
	importálás a könyvtárba; importálás a színpadra; kép használata kitöltőszíneként;
Animációk	
<i>Típusai</i>	
	állandó előtérrel rendelkező animáció; állandó háttérrel rendelkező animáció; objektum animáció;
<i>Technikák</i>	
	képkockáról képkockára: pálcikaember tornáztatása; gyakorló feladat: Majom; mozgásátmenet (Motion): felírat mozgatása, forgatása; gyakorló feladat: Felírat méretének növelése, fogatása; alakzatátmenet (Shape): ; gyakorló feladat: körből négyszög létrehozása; gyakorló feladat: alakzatból szöveg létrehozása; gyakorló feladat: szövegből szöveg létrehozása;
<i>Morphing</i>	
	az alakzatok egyes pontjait megadhatom, hogy melyik pontba képződjenek le; számok leképezése egyesből kettesbe morphing illetve nem morphing technikával;
<i>Mozgásvezetők</i>	
	Hogyan adhatom meg én egy objektum mozgásának irányát? labda pattogatatása; gyakorló feladat: Repülő mozgatása a felhőkarcok között;
<i>Maszkolás</i>	
	maszk és maszkolt réteg használata; szöveg maszkolása; gyakorló feladat: Betű színének folyamatos változtatása; gyakorló feladat: Fényszóró használata;
Szimbólumok	
<i>Grafika</i>	
	szimbólumokról általánosan; hópihe rajzolása;
<i>Mozi klip</i>	
	autó mozgatása miközben a kereke valóságosan forog; gyakorló feladat: fejecske forgatása miközben a szája hol mosolyog hol nem; gyakorló feladat: Reklámszöveg mozgásának animálása és szöveg színének változása;
<i>Gomb</i>	

	nyomógomb állapotai; egyszerű szöveges nyomógomb készítése; action Sript panel használata; actions Sript írása a gombra ; gyakorló feladat: Moziklipet tartalmazó gomb készítés; gyakorló feladat: Legördülő menü készítése; gyakorló feladat: Láthatatlan gomb előnye;
Egyéb lehetőségek	
	<i>Egér kurzor megváltoztatása</i>
	<i>Jelszavas védelem</i>
	<i>Nagyító maszkolás segítségével</i>
	<i>Hasznos oldalak</i>

Kinézet megtervezése

Képernyőterv elkészítése

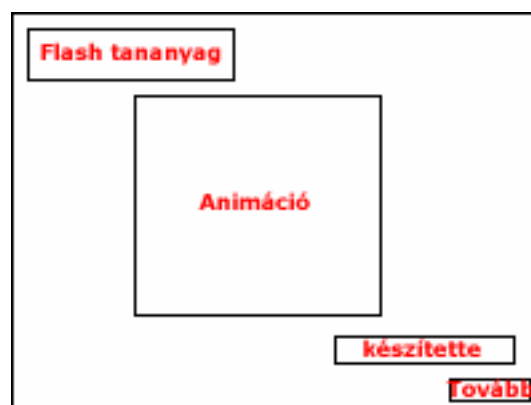
Az oktatóprogram ablak mérete 800x600 pixel. Ezt a felbontást azért választottam, mert az oktatóprogramomat szeretném feltölteni a webre is, és ott tapasztalatom szerint, ez az optimális, annak érdekében, hogy jól látható legyen.

A választott felbontás mellett szólt még az is, hogy a Flash program szerkesztése közben ez a legoptimálisabb méret.

Az oktatóprogram háttértárolóról való futtatása esetén viszont teljes képernyő méretben fog megjelenni. Ezt úgy fogom elérni, hogy használom az első frame-ben a `fscommand("fullscreen", "true");` actionscript parancsot, és publikálásnál .exe kiterjesztést fogok választani. [12]

A konkrét képernyőtervet táblázat segítségével készítettem el, melyben meghatároztam a menü, az animációk, a szövegek és a nyomógombok helyét..

Kezdő oldal terve



Program fő oldalának terve



Színek meghatározása

A színek kiválasztása mindenki számára nehéz feladat. Én úgy választottam ki a színeket, hogy a Flash jellemző színei fellelhetőek legyenek az oktatóprogramomban, ugyanakkor szoftverergonómiailag megfelelő legyen.



Háttér szín: #235676-os színkód. Ez a színkód az, ami a Flash szimbólumban az f betű színe.

Szöveg színe: #FFFFFF, vagyis a fehér. Ezt azért választottam, mert a háttéren ez a szín olvasható a legjobban.

Nyomógombok színe: #FF000, vagyis piros és #999999, azaz világos szürke.

Oktatóprogram megvalósítása

Kezdő oldal létrehozása

Mielőtt nekiláttam volna a munkához, beállítottam a háttérszínt (#235676), és az oldal felbontását (800x600). Majd az első képkockára beírtam a `fscommand("fullscreen", "true"); stop();` Action Script parancsot, melynek hatására teljes képernyőben fogom látni az oktatóprogramot, és mindaddig vár, amíg a tovább gombra nem kattintunk.



A kezdő oldal időegyesén 4 réteget helyeztem el. Az intro rétegen vannak a feliratok és az animáció, a gomb rétegen van a tovább gomb és az egér, illetve az eger2 réteg az egérkurzor megváltoztatása miatt kellett.



Feliratok elkészítése

A kezdő oldalon lévő feliratokat először a szövegeszköz (Text) segítségével hoztam létre, melynek beállításait a properties panelon tudjuk megváltoztatni.

Beállításai a következők:

- Betűszín: FF0000 - piros;
- Betű típusa: Monotype Corsiva;
- Betűmérete: 49pt és 32 pt.

A következő lépésben a szöveget feldaraboltam bitképpé. Ezt úgy érhettem el, ha jobb egérgombbal rákattintottam a szövegre és kiválasztottam a Break Apart parancsot. Első alkalommal még csak betűkre tördelte fel, viszont ha még egyszer kiválasztjuk ugyanezt a parancsot, akkor már megkapjuk a bitképet.

A színátmenetet úgy állítottam be, hogy a Mixer panelon kiválasztottam a Linear típusú színkitöltést, majd lehetőségem lett a használni kívánt két szín beállítására. Miután beállítottam a fekete és a piros színt, a festékes vödör (Paint Bucket) segítségével elértem, hogy a színátmenet hatóköre ne betűkre, hanem az egész szövegre vonatkozzon.

Animáció létrehozása

Mielőtt nekiláttam volna az animáció létrehozásához, be kellett importálnom a flash könyvtárba a képet, melyet felhasználok az animációba. A flash könyvtárát az F11-es billentyű leütésével tudom megjeleníteni.

Kép importálása:

- File/Import to Library;
- kép elérési útvonalának megadása;
- megnyitás gomb.

Az animációt egy új filmklip szimbólumba hoztam létre.

Filmklip szimbólum létrehozása:

- Insert/New Symbol menü kiválasztása. Itt meg kell adni a filmklip nevét (intro), és ki kell választani a létrehozni kívánt szimbólum típust.
- Két rétegre volt szükségem az animáció elkészítéséhez, ezért fel kell tennem még egy réteget (kezd, logo).
- Logo réteg első képkockájára betettem a beimportált képet. A flash könyvtárból egyszerűen csak a színpadra húztam. A 30, 50, 65, 66, 80, 81,

95, 96, 110 és 140-es képkockára kattintva beszúrtam egy kulcs képkockát (F6 billentyű lenyomása). A 65-ös képkockába átméreteztem a képnak a szélességét 17,7 pixelre. A 66-os képkockán tükröztem a függőleges tengelyre a képet, majd szintén lecsökkentettem a szélességét 17,7 pixelre. 81,82-es képkockán tükröztem a képet. 95-ös képkockán tükröztem, majd szélességét átállítottam 17,7 pixelre. 96-os képkockán csak a szélességét állítottam át 17,7 pixelre.


A 30 – 50; 50 – 65; 66 – 80, 81 – 95, 96 – 110, 110 – 140 közötti átmenetet átállítottam Motinonra a properties panel Tween beállításánál.

- Kezd nevű réteg első képkockájára létrehoztam egy 351x351 pixel méretű téglalapot, kitöltő színe ugyanaz, mint a háttérszín (#235676). A 30, 110, 140-es képkockán beszúrtam egy kulcs képkockát. A 30-as és a 110-es képkockán lévő négyzet kitöltő színének átlátszóságát 0%-ra állítottam a Color Mixer panel Alpha legördülő listájánál. Az 1 – 30 és a 110 – 140-es képkockák közötti átmenetek közül a Shape típust választottam.

Miután kész lettem az animációval, vissza kellett lépnem a fő oldalra (Scane 1).

Következő lépésként elhelyeztem az animáció egy példányát a fő oldalon. Ezt a legegyszerűbben úgy tudtam megoldani, hogy a könyvtárban található intro nevű objektumot megfogtam és ráhúztam az oldalra.


Egér kurzor megváltoztatása

- Az első réteg első framejébe beírtam a következő Action Scriptet.
`Mouse.hide();`
Ez a parancs az alap egérmutató kódját tünteti el.
- Feltettem egy új réteget. Ide rajzoltam egy ábrát, amiből a mutató lett. 
- Az alakzatot, miután kijelöltem az F8-al billentyűvel filmklippé konvertáltam. A filmklippet elneveztem egérnek. A regisztrációs pontját a bal felső sarkába helyeztem át, hogy az egérkurzor már a bal felső sarokban érzékeny legyen.
- Majd a filmklipre kattintottam duplán, ezzel beléptem a filmklip szerkesztőbe. Ezen belül lehet készíteni kis animációt a szimbólumnak.

- Miután kész lett az animáció, ki kellett lépni a szerkesztésből (Scene1) és az actions panelbe be kellett írni a következő Actions Script parancsokat (természetesen a szimbólum ki kell legyen jelölve):

```
onClipEvent (mouseMove) {
    _x = _root._xmouse;
    _y = _root._ymouse;
    updateAfterEvent();
}
```

Tovább nyomógomb elkészítése

- Készítettem egy filmklip szimbólumot és tovább nevet adtam neki. Az első képkockára rajzoltam egy nyilat, és alá írtam a tovább szót.  Tovább
- Beszúrtam a 15-ös képkockára egy kulcs képkockát, és az alakzatomat elmozdítottam jobbra. Az 1 – 15-ös kulcsképkocka közé beállítottam a Motion átmenetet. Ezzel elértem azt, hogy a nyíl és a szöveg folyamatosan jobbra mozog.
- Készítettem egy gomb szimbólumot, melynek a tova nevet adtam.
- Az UP képkockára elhelyeztem az előbb elkészített moziklip egy példányát. Ezzel oldottam azt meg, hogy a nyíl, amikor az egér nincs rajta, akkor mozog balról jobbra.
- OWER képkockára szintén megrajzoltam a nyilat és a feliratot. Ennek hatására, ha az egérkurzort ráhelyezem a gombra, megáll a nyíl.
- DOWN képkockára csak a nyilat rajzoltam. Ebben az esetben, ha az egérrel kattintok, és lenyomva tartom az egérgombot, akkor csak a nyíl látszik.
- HIT képkockára pedig egy akkora téglalapot rajzoltam, mely lefedi a nyilat és a feliratot. Ez lesz a nyomógomb találati felülete.

Mindezek után vissza kellett lépni a fő idősíkra (Scene 1), és behúzni a könyvtárból a nyomógomb egy példányát. Ezek után meg kellett adni a következő Action Script parancsot a gombra:

```
on (press)
{gotoAndPlay(2);}
```

Főoldal elkészítése

A főoldal szövegében egy kis ismertetőt olvashatunk, hogy miért is jött létre ez az oktatóprogram és hogyan használható.


A menüpontokat és annak almenüpontjait már a tervezés során meghatároztam, így már nem kellett azon gondolkodni, hogyan is nevezzem el az egyes menüpontokat.

A menü alapja egy nyomógomb, melyen nincs felirat. A feliratokat a filmklippen belül fogja megkapni minden egyes gomb.

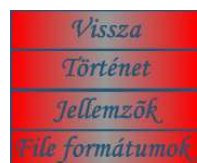
Menü elkészítése

- A menükhöz szükséges egy filmklip elkészítése, annak érdekében, hogy ha az egérkurzort ráhelyezzük a gombra, akkor a gombháttére, dinamikusan változzon.

A háttér alapja egy színátmenettel kitöltött téglalap. Ezt elhelyeztem az 1 képkockán, majd az 5-ös képkockán kitöltést tükrözve beszúrta egy kulcsképkockát, a 10-es képkockán pedig egy ugyanolyan négyzetet helyeztem el, mint az első. Ezek után a 1 – 5 és a 5 – 10 kulcsképkocka közé egy Shape típusú átmenetet tettem. Így egy olyan filmklippet kaptam, melyben egy téglalap kitöltő színe, változik.

- Következő lépésben egy gomb típusú szimbólumot hoztam létre, ahol az UP képkockára egy ugyanolyan téglalapot rajzoltam, mint az filmklippen az első képkockán van. Az OWER képkockára behúztam az elkészített filmklip szimbólum egy példányát. A DOWN és a HIT képkockára szintén egy olyan téglalapot helyeztem el, mint az Up képkockán.
- Amikor készen lett a nyomógomb, akkor egy új filmklippet kellett létrehozni. Első képkockájára beállítottam a `Stop();` parancsot, majd behúztam a gombnak egy példányát a könyvtárból. Ráírtam a Bevezető szót. 

Majd beállítottam a gombra az `on (release) {gotoAndPlay(2);}` Action Sript parancsot, így elértem azt, hogy ha rákattintunk a gombra, akkor a 2-es képkockára ugrik. A második képkockára ismét kiadtam a `Stop();` parancsot majd egymás alá 4 példányt húztam be, és ráírtam a négy almenüpontnak a nevet.



A **Vissza** gomb parancsa:

```
on (release) {gotoAndPlay(1);}
```

Ezzel a vezérlés visszakerül az első képkockára.

A **Történet** gomb parancsa:

```
on (release) {gotoAndPlay(3);}
```

Ezzel átugrunk a harmadik képkockára, ahol a Flash történetét olvashatjuk el.

A **Jellemzők** gomb parancsa:

```
on (release) {gotoAndPlay(4);}
```

Itt a Flash jellemzőit olvashatjuk.

A **File formátumok** gomb parancsa:

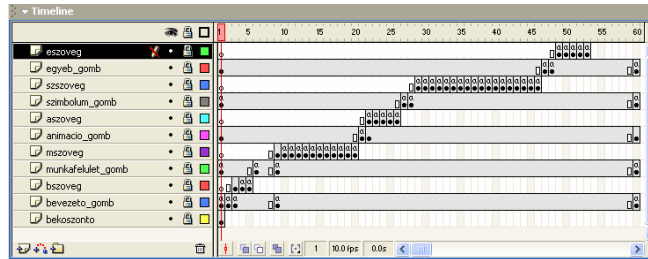
```
on (release) {gotoAndPlay(5);}
```

A teljes menü képe:

Bevezető	Munkafelület	Animációk	Szimbólumok	Egyéb lehetőségek
Vissza	Vissza	Vissza	Vissza	Vissza
Történet	Ablak felépítése	Tipusai	Grafika	Egér váltás
Jellemzők	Kijelölők	Technikák	Moziklip	Írásvédelem
File formátumok	Rajzeszközök	Morphing	Gomb	Nagyító
	Transzformálás	Mozgásvezetés		Hasznos oldalak
	Navigálás	Maszkálás		
	Színkeverő			
	Csoportosítás			
	Rétegek			
	Elrendezés			
	Éllágyítás			
	Szöveg			
	Kép importálás			

A tananyag minden egyes leckéjét külön filmklipben valósítottam meg, ha az egyes menüpontokat kiválasztjuk, akkor az ahhoz tartozó almenü jelenik meg, majd ha ezek közül választunk, akkor a hozzá tartozó filmklip objektumot olvashatjuk, azon a képkockán ahová az `on (release) {gotoAndPlay(x);}` paranccsal ugortunk.

Az oktatóprogramban minden egyes menüpontot és a hozzá tartozó tananyagot külön rétegen kezeltem.



Maszkolás tanegység elkészítése

A tanegység kinézete az oktatóprogramban

Bevezető	Munkafelület	Animációk	Szimbólumok	Egyéb lehetőségek
<p>Maszkolás</p> <p>A flash maszkjainak feladata ugyanaz, mint a valódi maszkoknak: eltakarni a mögéjük helyezett dolgokat. A Flashban a maszkoláshoz mindig kétféle réteg tartozik: egy maszkréteg és egy vagy több maszkolt réteg.</p> <p>Amikor bekapcsoljuk a maszkot, a maszkolt rétegnek kizárólag azok a részei látszanak, amelyek a maszk réteg kitöltött területei alatt helyezkednek el. A maszk elkészítéséhez a vonalakon kívül szinte bármit felhasználhatunk. A maszk lehet animált, de statikus is. Az egyetlen kikötés, hogy a maszkhoz nem kapcsolódhat mozgásvezető réteg és nem maszkolhatjuk a gombok rétegeit sem.</p> <p>Szöveg maszkolása</p> <ol style="list-style-type: none"> Hozz létre egy új 200x200 pixeles Flash dokumentumot! A Frame Rate-t (másodpercenként lejátszott képkockák száma) állítsd 25 fps-re! Állítsd át a háttérszínt egy neked tetszőre! Ne maradjon fehér! A Text Tool (T) segítségével írd egy tetszőleges szöveget a színpad közepére! Készíts egy új réteget az előző fölé és másold le a begépelte szöveget! (CTRL+C) Az új rétegre állva add ki az Edit/Paste in Place (CTRL+Shift+V) parancsot, beillesztve a szöveg másolatát változatlan pozícióban. Az új szöveg színét állítsd át fehérre! 	<p>8. Hozz létre egy harmadik réteget a másik kettő fölött!</p> <p>9.1 de rajzolj egy téglalapot! A kitöltése lehet fehér, szegély nem kell rá!</p> <p>10. Ezt a téglalapot alakítsd át szimbólummá! (F8 - MovieClip)</p> <p>11. A téglalapot tartalmazó rétegre készíts egy 30 képkockából álló Motion Tween animációt! Az 1. képkockán a színpad bal, a 30. képkockán a színpad jobb oldalán helyezkedjen el a téglalapot!</p> <p>12. Állítsd át a téglalapot tartalmazó réteget Mask típusra az alatta lévő pedig Masked-ra!</p>	<p>MyPixel.hu</p> <p>Gyakorlati feladat: 1 Feladat 2 Feladat</p>		

A maszkolás tanegységet is, mint mindegyik tanegységet, külön filmklipben valósítottam meg.

Létrehoztam a maszk nevű filmklippet. Beállítottam az első képkockára a `stop()`; Action Sript parancsot annak érdekében, hogy az animáció lejátszása megálljon, ha ezt a tananyagot olvassa a tanuló. A következő lépésben beírtam a szöveges részt, mivel a teljes szöveg nem fért fel egy oldalra, így a következő képkockán folytattam, ahol szintén egy `stop()`; paranccsal megállítottam a lejátszást. Az első képkockára beillesztettem egy tovább nyomógomb egy példányát, amit már a kezdőoldalnál létrehoztam, de itt is nyugodtan felhasználhatom, hisz ebben rejlik a Flash előnye, hogy az elkészített objektumokat bármikor és bármennyi alkalommal újrahaználhatjuk anélkül, hogy a Flash file-unck mérete növekedne. Beállítottam az Actions Script parancsot a gombra, úgy hogy a 2-es képkockára ugorjon.

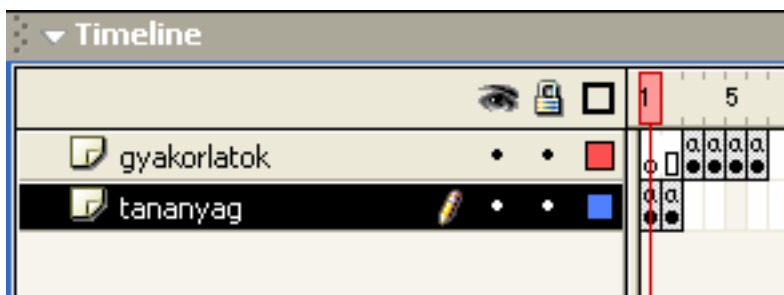
A 2-es képkockán folytattam a szöveg beírását, és szemléltető képet is beillesztettem mellé, amit előtte a Flash könyvtárba importáltam. A tananyag végére pedig betettem azt az animációt, amit a tananyag elvégzése után a tanulónak eredményül kellene, hogy

kapjon. A bal alsó sarokban található egy vissza nyíl, ami az előző oldalra irányít vissza. Elkészítésének menete hasonló, mint a tovább gombnak.

A jobb alsó sarkában kapott helyet a gyakorló feladat, ahol még két darab feladatot készíthet el a tanuló annak érdekében, hogy a tudása még alaposabb legyen.

A **1 Feladat** és a **2 Feladat** gomb is egy-egy filmklip szimbólumot tölt be. Ezek a maszk nevű filmklip gyakorlatok rétegén helyezkednek el. Itt is minden képkockára be kellett állítani a `stop()` parancsot.

A maszk filmklip időegyenese:

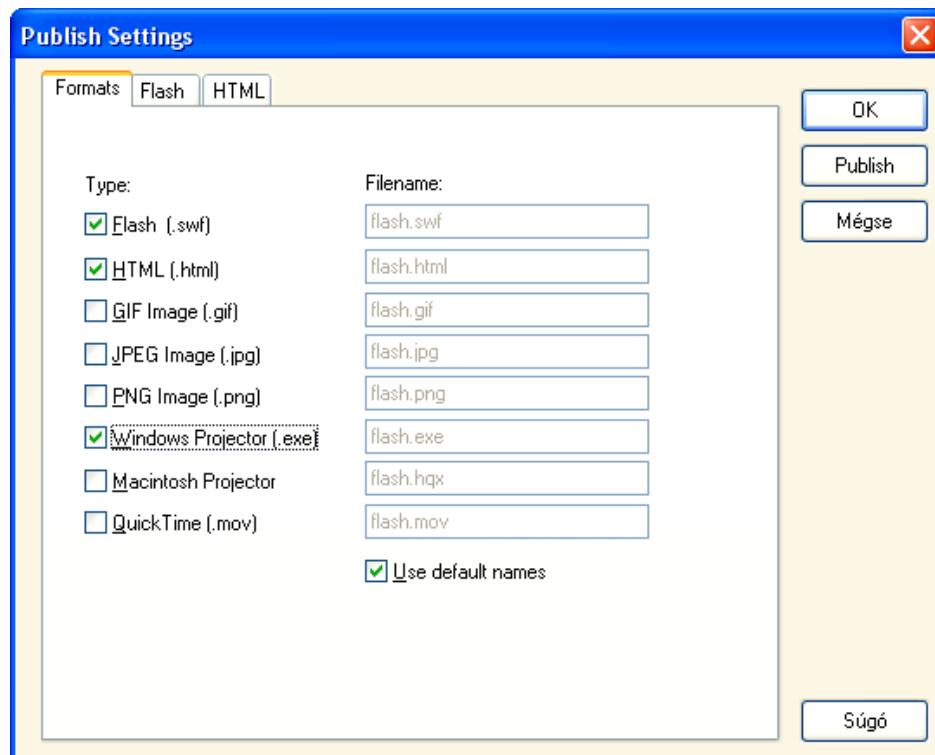


Oktatóprogram publikálása, használata

Publikálás

A kész Flash mozit egyben megnézhetjük, ha a Control/Test Movie menüre kattintunk. Ha a működését rendben lévőnek találjuk, akkor nekiláthatunk a publikálásnak. Ez egy két lépésből álló eljárás. Az első lépésben a File/Publish menüvel elkészítjük a közlésre szánt állományt.

A Publish Settings dialógusablakban egyszerre többféle file formátumot is választhatunk:



Az első az SWF, azaz a Shockwave Flash file, amit a lejátszó megjeleníthet. A HTML file egy egyszerű, üres web-oldal, ami tartalmazza az SWF file-t beágyazásához szükséges kódrészletet. Ezt betölthetjük majd egy Mozilla-val vagy internet Explorerrel, de ahhoz, hogy a böngészőben megjelenjen a mozi, a HTML mellett az SWF file-ra is szükség van.

Mindezekén túl készíthetünk önálló projektor (vetítő) file-t MS Windows vagy Macintosh rendszerre, ami egy futatható exe file-t jelent, valamint QuickTime mozi, PNG, GIF vagy JPG képsorozatot is.

A Flash nem önállóan futtatható animációkat készít. A kész mozi megtekintéséhez egy lejátszó program, a Flash player szükséges. A lejátszó tipikusan a böngészőbe beépülő modul (Shockwave plug-in), ami az újabb web-böngészőprogramokkal együtt, és általában alapértelmezés szerint telepítésre kerül gépünkre. Ha böngészőnk nem ismeri a Flash-t, akkor telepítsük újra úgy, hogy telepítéskor válasszuk a Flash plug-in-t is, vagy töltsük le azt a www.adobe.com-ról. Az újabb böngészők, amikor először találkoznak Flash-t tartalmazó oldallal, akkor automatikusan felkínálják a plug-in telepítését az Adobe (Macromedia) honlapjáról. A plug-in telepítőjét a fejlesztőkörnyezet is tartalmazza, installálás után a C:\Program Files\Macromedia\Flex MX\Players\ mappában találjuk. Macintosh, Windows és JAVA VM platformra.

Én az oktatóprogramomat HTML és EXE file-ba publikáltam. A HTML file-t azért választottam, mert miután kész a program, felmásoltam az iskola szerver gépére, hogy a tanulóim ki tudják próbálni. Az EXE file-t pedig azért, hogy tudjam CD tárolón is terjeszteni azoknak, akik nem tudnak internet közelébe kerülni. [11]

Használata

Az oktatóprogram indítása után célszerű elindítani a Macromedia Flash MX programot is, ugyanis a tananyagok többsége gyakorlati feladat, amit célszerű az olvasás közben ki is próbálni.

Indítás után a kezdő oldal jelenik meg, ahonnan a tovább gombra kattintva juthatunk el a főoldalra. A főoldalon találunk egy kis bevezető szöveget, egy rövid leírást, hogy hogyan használhatjuk a programot és a menüpontokat. A menüpontokra kattintva megjelenik egy legördülő menü, ahol megjelennek az almenüpontok, ha mégsem ezt a menüt akarjuk kiválasztani, akkor a menü tetején lévő vissza gombra kattintva újra a főoldalra mehetünk.

A megfelelő almenüre kattintva jelennek meg a tananyag egységek. Abban az esetben, ha egy tananyag egység nem fért el egy oldalon, akkor a jobb alsó sarokban található egy tovább gomb, amivel folytathatjuk a kiválasztott rész olvasását. Ha vissza szeretnénk jutni az előző oldalra, akkor a bal sarokban található vissza gombot kell választani.

Egyes tananyagrészekhez gyakorló feladat is található. Ezeket a gyakorló feladat alatt található 1 feladat illetve 2 feladat gombra kattintva érhetjük el.

Kilépés a programból úgy történik, hogy az ESC billentyű lenyomásával elérjük, hogy ablakos nézetbe kerüljünk, így már látható lesz a windows-os ablak bezárás gombja a jobb felső sarokban.

Összefoglalás

Véleményem szerint sikerült a szakdolgozatomban egy átfogó képet adnom, hogy hogyan is készíthető el egy oktató program, ami megfelel mind szakmailag, mind pedig pedagógiailag, és nem utolsó szempontként megfelel a szoftver ergonómiának is.

A szakdolgozat elkészítésekor sok tapasztalatot szereztem minden olyan témakörben, amit feldolgoztam, és biztos vagyok benne, hogy a tanítási óráim során is hasznomra válik.

Az oktatóprogramomba sikerült minden olyan tananyagot beletennem, ami elengedhetetlen, ahhoz hogy egy alapszintű ismeretet szerezzon a tanuló a Flash használatához, ugyanakkor észrevegye, hogy mi minden lehetőség rejlik az animáció készítésben.

A programot leteszteltem az egyik multimédia fejlesztő csoportomnál, ahol a cél, hogy egy multimédiás weblapot hozzanak létre Flash programmal. A csoport egyik felének megmutattam az oktatóprogramot, és ők a program segítségével ismerkedtek meg a Flash lehetőségeivel, a másik pedig hagyományos oktatási módszerrel tanulta meg a program kezelését és lehetőségeit.

A tapasztalatok azt mutatják, hogy a program elérte a célját, hiszen azok a tanulók, akik használták az oktatóprogramot, sokkal gyorsabban és lelkesebben tanulták meg az animációkészítés alapvető lépéseit, és a tananyag végére fantáziadús weblapot hoztak létre. Ezzel szemben, azok a tanulók, akik nem használták az oktatóprogramot, sokkal nehezebben és kevésbé motiváltan készítették el a flash alapú weblapjukat.

Az oktatóprogram lényeges tulajdonságának tekintem, hogy bővíthető, hiszen további tananyag részek elkészítése után könnyen integrálható. A Flash lehetőséget ad arra, hogy egy külön file-ban elkészített egységet beimportáljunk filmklippé, majd egy nyomógomb egy példányának hozzáadásával és annak felirattal való ellátásával már bővítve van a programunk a tananyagrészzel.

Irodalomjegyzék

- [1] <http://edutech.elte.hu/jegyzet/tipografia/tartalom.htm>
- [2] <http://webmania.cc/web-tipografia-1/>
- [3] http://webszabvany.blog.hu/2008/a_szinek_elmelete
- [4] <http://www.color-wheel-pro.com/color-schemes.html>
- [5] http://www.sulinet.hu/ikt/tovabbkepzesek/html/lesson16_3_19.html
- [6] <http://www.swf.hu>
- [7] <http://www.tutorialized.com/tutorials/Flash/1>
- [8] <http://www.entheosweb.com/Flash/default.asp>
- [9] Komenczi Bertalan: Orbis sensualium pictus - Multimédia az oktatásban
- [10] Cserjés Á.: A szoftver ergonómia alapvető problémái. Ergonómia. VI. Ergonómiai konferencia Budapest 1994.
- [11] Bonnie Blake: Macromedia FlashTM 5, Panem Kiadó Budapest, 2001.
- [12] Sham Bhangal, Ben Renow-Clarke: Actions Script programozás alapjai
- [13] Robert Reinhardt, Jon Warren Lentz: Flash 5 Biblia, Kiskapu Kiadó, 2001.
- [14] Tay Vaughan: Multimédia, Making it work Macromedia, 1996.
- [15] Phillip Kerman: Tanuljuk meg a Macromedia Flash MX 2004 használatát 24 óra alatt
- [16] Sikos László Macromedia Flash (normál és professional változat) MX 2004 és 8 verziók
- [17] James L. M.S.Ed.: Flash MX 2004: Graphics, Animation & Interactivity