

SZAKDOLGOZAT

Barkóczy Zsolt

DEBRECEN, 2008

**DEBRECENI EGYETEM
INFORMATIKA KAR**

HÁLÓZATOKAT OKTATÓ MULTIMÉDIÁS CD KÉSZÍTÉSE

Témavezető:
Dr. Nyakóné Dr. Juhász Katalin

Készítette:
Barkóczi Zsolt
informatika tanár szakos hallgató

DEBRECEN, 2008

TARTALOMJEGYZÉK

Bevezetés.....	1
1. FEJEZET.....	3
Vezérelvek a képernyőn megjelenő oktatóprogramok tervezéséhez	3
1.1 Üzenettervezés	3
1.2 Az elrendezés vezérelvei	4
1.3 A kiemelés és figyelemfelkeltés vezérelvei	6
1.4 A betűtípusok és stílusok használatának elvei	7
1.5 A szövegelhelyezés elvei	8
1.6 A színhasználat elvei.....	9
1.7 Grafikus és interfész útmutató	12
2. FEJEZET.....	13
A Multimédia oktatási anyagok fejlesztésének pedagógiai szempontjai.....	13
2.1 Pedagógiai alapelvek	13
2.2 A Multimédia oktatási anyagok fejlesztésének pszichológiai szempontjai.....	13
2.2.1 Az egyéni tanulási stílusok figyelembe vétele	14
2.2.2 Az információfeldolgozás	15
2.2.3 A tanuláspszichológia alapkérdése	15
2.3 Az egyéni tanulási stílusok	16
3. FEJEZET.....	17
A Flash keretrendszer	17
3.1 A Flash 5 keretrendszer bemutatása.....	17
3.2 Bitkép-kezelő	17
3.3 Vektoros rajzoló.....	18
3.4 Vektoranimáció-készítő	18
3.5 Szerzői környezet.....	18
3.6 Animációs program.....	19
3.7 Programozói és adatbázis-felület	20
3.8 A Flash 5 filmek alkalmazási területei.....	21
4. FEJEZET.....	23
A projekt során használt médiaanyagok bemutatása.....	23
4.1 Az interaktív Flash anyagok megtervezése	23
4.2 Flash film fájl típusok	25
4.3 Szövegek kezelése	26
4.3.1 Szöveg címke	28
4.3.2 Szövegblokk.....	28
4.3.3 Szerkeszthető szöveg	29
4.3.4 Statikus (nem szerkeszthető) szövegek	29
4.3.5 Szerkeszthető szöveg mezők.....	30
4.3.6 Dinamikus szövegmezők	30
4.4 Képek, rajzok, animációk a Flasben	31
4.4.1 Az animáció és az állóképtervezés kapcsolata	32
4.4.2 Animáció különböző időszekvenciákon, különböző rétegeken	33
4.5 Hangok a Flashben.....	34
4.5.1 Mintavételezési frekvencia	34
4.5.2 Bitfelbontás	36

5. FEJEZET	38
A Flash projekt kivitelezése	38
5.1 A projekt szerkezete.....	39
5.2 A főmenü szerkezete (fejlec.swf).....	42
6. FEJEZET	43
Az oktatóanyag tanítási egységeinek szerkezete	43
6.1 A program működése	43
6.2 A projekt objektumainak készítéséhez alkalmazott megoldások	45
6.3 A lecke témaköreit tartalmazó filmszimbólum szerkezete.....	46
6.3.1 A Flash gombok eseménykezelői	48
6.4 A leckék témaköreinek szerkezete és működése.....	51
6.4.1 Az animációs klipet vezérlő gombok.....	51
6.5 A szöveges rész működése.....	52
6.6 A szintfelmérő panel működési elve	55
7. FEJEZET	61
A multimédiás CD elkészítése	61
7.1 A CD automatikus indítása	61
7.2 Interneten való publikálás	63
8. FEJEZET	64
A tanuló értékelés	64
8.1 Kommunikáció, interakció.....	66
8.2 Általános (pedagógiai, pszichológiai, ergonómiai) szempontok.....	67
8.3 Multimédia komponensek.....	68
Összefoglalás	71
Irodalomjegyzék	72
Melléklet	75

Bevezetés

A multimédia megérett arra, hogy mindennapos eszköznek tekintsük a tanítás-tanulás folyamatában. Mint eszközrendszer felhasználása összhangban van a didaktikai alapelvekkel, ugyanakkor azonban megkívánja, hogy újra rendszerezzük és kibővítsük azokat az ismereteket, amelyek ennek az eszközrendszernek az oktatási folyamatba való hatékony beépítését, alkalmazásának didaktikai-módszertani alapjait jelentik. Formálódó társadalmunkban az „informatika” az információ tudománya hatalmas léptekkel halad előre. A tanítás nem maradhat el ezektől az eredményektől, kihívásoktól. Minél előbb a felnövekvő generáció kezébe kell adnunk ezeket az eszközöket, akiknek ezek az eszközök épp oly megszokott mindennapi eszközök, mint az eddig használatosak.

Az általam készített interaktív CD-vel megpróbálok egy olyan eszközt adni a tanulók kezébe, amivel remélhetőleg nagyobb hatásokkal sajátítják el a számítógépes hálózatok használatához nélkülözhetetlen ismereteket. A CD elkészítése során a „Macromedia Flash 5” nevű programot alkalmaztam. A program kiválasztásánál az elsődleges szempont amit szemelőt tartottam, hogy lehetőleg egyetlen szoftver segítségével tudjam elkészíteni a szükséges grafikákat, klipeket, animációkat, valamint ezeket interaktivitásra tudjam bírni.

Napjainkban a Flash közismert szakmai kifejezéssé vált: ritka az olyan honlapfejlesztő vagy programozó, aki ne hallotta volna még ezt a nevet. Amellett, hogy a program maga is széles körben ismert, manapság Flash-filmekkel találkozunk a média szinte minden területén. A rajzfilmsorozatok nagy része, illetve a számítógépes játékszoftverek is Flash-technikával készülnek. A Flash 5 interaktív weboldalak és interaktív multimédiás anyagok elkészítésére önmagában is alkalmas eszköz a programozók kezében, amely ráadásul olyan kisméretű állományokkal dolgozik, ami a szokásos médiaállományok töredéke.

Egy multimédiás CD-n szereplő szemléltető ábrák, az anyag elsajátítását segítő filmek és maga a tananyag szövege is csak akkor lesz igazán hasznos

a tanulók számára, ha a program kezelése nem ütközik akadályba a számukra. (A szöveges felületeket állóképek, táblázatok, animációk egészítik ki). A szöveghez értelemszerűen mozgóképanyag is tartozik, melynek hozzáférése közvetlenül az erre szolgáló nyomógomb segítségével biztosított. A szöveges részbe a felhasznált irodalom és saját konkrét tapasztalataim alapján azokat a lehetőségeket próbáltam kiemelni a tananyaggal kapcsolatban, amelyek a mai valós technikai környezetben az iskolák rendelkezésére állnak.

1. FEJEZET

Vezérelvek a képernyőn megjelenő oktatóprogramok tervezéséhez

A közelmúlt kutatásai között egy sincs, amely átfogóan összefoglalná a képernyőn megjelenő üzenettervezést. A tervezésnek a szakmában régebben létezett hangsúlyai nem érvényesek, és azok a cikkek tárgyalják, amelyek nem elérhető tanácsokat tartalmaznak. A látványtervező tehát kutatásait csak több forrás egyidejű tanulmányozása mellett a „jó” vezérelvek kiválasztásával, az elavult vezérelvek elkülönítésével, óvatos előrehaladással végezheti.

1.1 Üzenettervezés

Üzenettervezésen „egy üzenet formai kialakításának művészetét és tudományát” értjük. Az üzenet illetve a látványtervezőnek rendelkeznie kell az elrendezés és egyensúly iránti művészi érzéssel, kreativitással, valamint a képernyőt majdan néző embernek lelki világa iránti megértéssel, empátiával, mivel a vizuális tervezési képességek gyakorlatilag az üzenettervezés minden aspektusára hatással vannak. Ezen adottságon felül egy hatékony képernyőn megjelenő (videó és számítógépes programokkal dolgozó) látványtervezőnek, az audiovizuális média, az ipari tervezés, a kognitív pszichológia, az emberi tényezők, az audiovizuális tervezés, lehetőleg a számítógépes tudomány és a konvencionális publikáció grafikus és szerkesztői tervezése terén is szüksége van némi háttérre. Ez nem kis követelmény egy látványtervezővel szemben. [13]

A szakértelemnek ez a szintje nemcsak esztétikai jellegű. A képernyőn megjelenő oktatóprogramok sokkal érzékenyebbek az üzenettervezés minőségére, mint a nyomtatott formájú oktatóanyag. Más szóval, egy jó üzenettervezésű, képernyőn megjelenő oktatóprogrammal hatékonyabban oldható meg az oktatás, mint egy jó nyomtatott anyaggal. Fordítva viszont:

egy rosszul megtervezett képernyőn megjelenő oktatóprogram kevésbé hatékony oktatóanyag, mint egy rosszul megtervezett nyomtatott oktatóanyag. [13]

A szabványok hiánya tovább bonyolítja a képernyőn megjelenő programokon dolgozó látványtervezők munkáját. Annak idején 100 évbe telt, amíg az első modern könyv (Gutenberg Bibliája) megjelenését követően a könyvkiadásban meghonosodott az oldalszámolás, az indexek, a tartalomjegyzékek és a címlapok gyakorlata. [13] A képernyőre alapozó látványtervezőknek még ma sem áll rendelkezésre néhány általános vezérelvénél több. Nemcsak képernyőn megjelenő programok, különösen nem túl régi számítógépes programok állnak rendelkezésükre, hanem, nagy választékban, integrálásra alkalmas tervezési elemkészlet is. A könyvtervezők szöveget, számokat, állóképeket és fényképeket használnak fel. A képernyőn megjelenő programok ezenkívül animált képsorozatokot, térbeli és numerikus információkat, hiperlinkeket és digitális audio-vizuális anyagot is beépíthetnek. Ezekre viszont egész egyszerűen nincsenek megtanulható szabványos eljárások. Jelenleg a számítógépes megjelenítésre alkalmas grafikus tervezési minták a hagyományos nyomtatott dokumentumok stílusbeli és szervezési konvencióinak sok szempontját utánozzák. Ez nem a legjobb út a képernyőn megjelenő programok tervezése irányába.

1.2 Az elrendezés vezérelvei

Bizonyos értelemben az üzenettervezés a választott médiumtól függetlenül is értelmezhető. A legfőbb elv az, hogy úgy tervezzünk, hogy a fontos információk azonnal kiugorjanak, míg minden egyéb ezeknek legyen alárendelve, úgy, hogy emellett a figyelmet ne vonja el. Ha ez az elv korrekten megvalósul a képernyőn megjelenő programokban, a médium áttekinthető. Sok tervező, ennek az elvnek ellenében, a képernyőn megjelenő program tervezése során a nyomtatott oldalt próbálja utánozni. Ez nem járható út, mert a legtöbb felhasználó szívesen lát egy oldalt

„egészben”. Egy tipikus nyomtatott oldal megjelenítése egy számítógépes felhasználásban lehetséges ugyan, de ez a felhasználót állandóan arra kényszeríti, hogy előre-hátra keressen, ha az egész oldalt látni akarja, ami meglehetősen kényelmetlen oldalfelbontást igényel, ha nagyobb terjedelmű szöveg megjelenítésére van szükség.

A kísértést a legjobb úgy legyőzni, hogy képernyőnként csak egy téma jelenjen meg. Ez két előnnyel is jár: egyrészt korlátozza az egy képernyőre jutó szöveget; másrészt kényszeríti a látványtervezőt, hogy koncentráljon a rendelkezésre álló képernyőterületre.

Egy konzisztens formátumhoz való ragaszkodás előnyös a látványtervező és a fejlesztő számára is. Az elv olyan szerkezet tervezése, amely lehetővé teszi a fejlesztő számára a grafikus elemek képernyőről képernyőre való újrafelhasználását, és a képernyőkoordináták oldalról oldalra való megőrzését. Ez lehetővé teszi, hogy a képernyőt anélkül alkothassa meg, hogy az elrendezést mindig újra ki kelljen találnia, és új tervezési problémákat kelljen megoldania.

„A nyomtatott médiával ellentétben, a nagy üresen hagyott helyek nem növelik meg a költségeket. A koncepcionális vagy esztétikai okokból logikailag elkülönítendő információk könnyedén elhelyezhetők külön képernyőkön, mert nem drágább dolog két képernyőt használni egy helyett.” [13]

Kutatások azt mutatják, hogy egy nyomtatott oldal könnyen háromszor annyi karaktert tartalmazhat, mint egy on-line képernyő. E kutatás szerint míg az általánosan megszokott nyomtatás az oldal nagyjából 50%-át tölti ki, a tanulmányok a képernyőn megjelenő programok esetében általában a képernyő 80%-át telítve találják. Az olvashatóságot tekintve a képernyőn megjelenő programok jobb, ha kisebb telítettségűek. Képernyőn megjelenő programok szövegsűrűsége a képernyőterület 25%-a esetén optimális. Az információ térbeli elkülönítése megnöveli az

olvashatóságot, és világossá teszi, hogy a képernyő mely területén mely információ érhető el.

1.3 A kiemelés és figyelemfelkeltés vezérelvei

"A kiemelés túlzott használata a felhasználói felület egyik leggyakrabban előforduló hibája." [13]

A kiemelés és figyelemfelkeltés a képernyőn megjelenő programok integráns részei. A kiemelés segíti a felhasználót a fontos információk felismerésében, és figyelmé elterelésének elkerülésében; de túlzott alkalmazása több zavart, mint hasznot eredményez.

A dőlt betű kiemelés céljából való alkalmi használata nem mindig könnyíti meg a szövegértést. A szöveg kiemelése megkönnyíti a kiemelt szöveg memorizálását, de nem az egész szövegét. Ebből az következtetik, hogy a nem kiemelt szöveg megjegyzésére a kiemelés hátrányosan hat. Mivel a tanulás nem javul, ha rövid a tanulási idő, jobb elkerülni a szövegben a vizuális figyelemfelkeltést, kivéve, ha a kiemelendő információ nagyon fontos. Ugyanakkor vannak olyan vizsgálatok, melyek azt mutatják, hogy a kiemelés és a figyelemfelkeltés igenis pozitív hatású. A tanulási folyamat javítható az alapvető vizuális figyelemfelkeltések számának növelésével.

A kiemelő módszereknek meg kell felelniük a kiemelt információ fontosságának. Rangsorolva a képernyőn megjelenő programokban használt kiemelő módszereket három csoportba sorolhatjuk azokat.

- A villogás és a hangjelek eszerint elsőrendű figyelemfelkeltő eszközök, tekintve, hogy a legszembetűnőbbek.
- A másodrendű figyelemfelkeltő eszközök közé tartoznak a szín-, intenzitás- és méretkülönbségek, valamint az objektum körüli keret.

- A harmadrendű figyelemfelkeltő eszközök közé tartozik a betűtípus-váltás (például félkövér vagy dőlt betű), a szöveg vagy grafika aláhúzása, a csupa nagybetű használata vagy az eltolás.

A villogás - mint figyelemfelkeltő eszköz - olyan szituációkban használható, mint kritikus hiba- és figyelmeztető üzenetek, megerősítést igénylő helyzetek, amelyekben adattörlesztés vagy - változtatás következhet be, illetve a normális gyakorlattól való komolyabb eltérések. Ez a konzervatív elv azért célszerű, mert a villogás messze a legszembetűnőbb formája a kiemelésnek. Heines szerint a villogást csak nagyon fontos üzenetek esetében alkalmazható, és képernyő-oldalanként egynél nem gyakrabban.

1.4 A betűtípusok és stílusok használatának elvei

A grafika, a videó és az animáció előnyeinek ellenére, sok téma (és költségvetés) a szöveges megjelenítést részesíti előnyben. Az olvashatósággal kapcsolatos okok miatt a szövegtervezés fontosabb a képernyőn megjelenő programok, mint a nyomtatásra alapozott programok esetében.

Mivel a felhasználók hajlamosak egy prezentációt annak külső megjelenése alapján megítélni, fontos, hogy olyan szövegstílust válasszunk, amely jól hat a képernyőn. Ha egy képernyőn megjelenő prezentáció az anyag kinyomtatását is igényli, (egy második nyomtatási verzió átdolgozására való lehetőség nélkül) akkor azt a betűtípust célszerű használni, amely a képernyőn hat a legjobban, tekintet nélkül a nyomtatott formájára. Nyomtatott anyagok készítése esetén a felbontás kérdése ritkán okoz problémát. Még egyszerű folyóiratok is 1200 dpi (pont per inch) sűrűséggel vannak nyomtatva, ami a legtöbb célra elegendő. A képernyőn megjelenő technológiák nem érik el ezt a felbontást. Egy tipikus számítógépes monitor 640 x 480 képpontból áll, miközben a képernyő mérete körülbelül 10,5 x 7,5 inch (14 inch

képátmérő esetén). A felbontás ennek megfelelően 64 képpont inchenként.

Az alacsony felbontással kapcsolatos konvencionális bölcsesség a sans serif betűtípus használatát írja elő a képernyőn megjelenő programoknál. Tekintve a legutóbbi időkben megjelenő, lényegesen megnövekedett felbontású monitorokat, az idézett bölcsesség már kevésbé alkalmazható általánosan. Ebből arra a következtetésre jutunk, hogy a látványtervező figyelembe kell, hogy vegye az alkalmazott monitor felbontását és letapogatási módját, valamint a megjelenített betűméretet.

Ha már választottunk betűtípust, a szöveg jobb megjelenését a betűtávolság alkalmas megválasztásával érhetjük el. Bár ezzel a technikával a betűszélesség túl nagy variációjával vissza is lehet élni, egy változtatható betűszélességű típus általában kellemes a szemnek, és elősegíti az olvashatóságot. Amikor az ember olvas, nem a szavakat alkotó egyes betűket nézi, hanem a szó, mint egész megjelenési formáját. A nagybetűvel írt szó ezt a megjelenési formát megzavarja. Vizsgálatok azt mutatják, hogy a kisbetűs szöveg gyorsabban olvasható. A kisbetűk és nagybetűs írásmód vegyesen is használható, kivételt talán az egysoros címek képeznek.

1.5 A szöveghelyezés elvei

A szöveghelyezés a képernyőn megjelenő üzenettervezés döntő eleme. A nyomtatás stílusa, a sorhossz, a sorkizárás és a sortörési pontok mind javíthatnak is és ronthatnak is az olvashatóságon. A jól strukturált szöveg elrendezése az anyag hierarchikus és szisztematikus kifejtését jelzi. Fejlécek, bekezdések közötti térköz, és irányított figyelemfelkeltés használható. Az egységes külalak érdekében fontos meghatározni, mennyi hely is áll ezek rendelkezésére. Egy adott helyre beillesztendő legnagyobb szövegblokk megtervezésével tudjuk az adott szövegrész számára a sorközt és a szükséges méreteket a leghatásosabban meghatározni. A hagyományos nyomtatásra alapozott szabályok még

mindig használhatóak bizonyos szövegelemek elrendezésére. A rövidítések és a betűszók lehetőleg kerülendők. A keltezést, kódokat az elválasztó karaktereik szerint zárkóztatjuk; az alfabetikus adatokat a legjobb balra zárni, a numerikus adatokat pedig a tizedesvessző szerint, annak hiányában jobbra zárjuk. Szövegblokkokat lehetőleg egyhasábos alakban helyezzünk el. Egy 1988-ban végzett vizsgálat megállapította, hogy a kéthasábos alak 200 szó/perc értékkel csökkenti az olvasási sebességet. Ez a kéthasábos szöveg olvasásához szükséges nagyobb szemmozgásnak tudható be. [5]

Míg az általános felfogás szerint az egyhasábos és dupla sortávolságú szöveg az előnyös, a hasábhosszok nem lehetnek túlságosan nagyok. Az olvashatóságot rendkívüli módon csökkenti, ha képernyőt egyetlen nagy szövegtömbbel töltjük meg. A betűméret és a sorhossz mindig együtt járnak, így a sorhosszt legcélszerűbb a szavak számában kifejezni. A "WYSIWYG" (azt kapod, amit látsz is) technológia használata esetén is, a soronként tizenöt szónál többet tartalmazó szöveg a felhasználó számára már kevésbé világosan olvasható. Jobb az óvatosság irányában tévedni, és a szükségesnél inkább nagyobb betűket és rövidebb sorokat létrehozni.

1.6 A színhasználat elvei

"mindenkinek magának kell kikísérletezni a rendszerében legjobban mutató színeket".

(Heines)

A színek hatásos használata lényegesen javíthatja a felhasználók dolgát a képernyőn megjelenő prezentációk esetében. A tisztán esztétikai megfontolásoktól eltekintve, a színes képernyő legfontosabb előnye az információk nagyobb mélységbeli és felbontásbeli megjeleníthetősége. Gordon (1995) által elvégzett többrendbeli vizsgálatok adatait elemezve egyértelműen kiviláglik, hogy a színek használata a grafikában a

tanuló képességét megnöveli a tanulmányozandó diagramok egyes részeinek felismerésében. A színek megfelelő használata a képernyőn megjelenő prezentációkban egyértelműen megéri a fáradságot. [5]

A szín helyes megválasztása nehezebb, mint látszik. Bár a normális emberi szem 7 millió szín megkülönböztetésére képes, a látható spektrumban csak kb. 150 egymástól megkülönböztethető hullámhossz van. Ezek között egy átlagos személy csak 12-13 hullámhosszt tud kellő pontossággal megnevezni. Ez az egészséges népességre vonatkozik. A férfi népesség kb. 7%-a szenved színtévesztésben vagy színvakóságban. Az 1950-es években végzett jól megalapozott vizsgálatok megállapítják, hogy a korosodó korosztályok elveszítik a kék és a zöld színek megkülönböztetésének képességét, amit feltehetően a cornea néha előforduló elsárgulása okoz. Az átlagos népesség általános elöregedését tekintve, a színválasztás előtt célszerű a megcélzott felhasználói réteg figyelembevétele.

Szükséges a megjelenítő médium és módszerek felmérése is. Az analóg videó nehezebben jeleníti meg a vöröset, a narancsot, a kéket és az ibolyát (a látható színek szélén levő színeket), mint a digitális (számítógépes) videó. Nyomtatásban a színek megjelenítése azonban mindkettőjüknél jobb. A képernyőn megjelenő programok nehézsége bizonyos színek megjelenítésében azonban néha előnyösen alkalmazható. Például a kék nehezen reprodukálható, ezért háttérszínnek jól használható.

A képernyőn megjelenő programok kontrasztarányát 20:1 értéken érdemes tartani. A világos hátterek (mint pl. az erős fehér) akkor igazán hatásosak, ha a használt betűtípus és grafika szintén feltűnő. A világos háttér kislebontású médium esetén súlyosbítja a finom részleteknél felmerülő problémákat. Ennek tudatában választhatunk az olvashatóságot növelő színösszeállítást. A legjobban látható színösszeállításokat az *I. táblázat* mutatja.

Háttér és előtér színek párosítása	
Előtér színe	Háttér színe
sárga	fekete
fehér	kék
fekete	narancs
fekete	sárga
narancs	fekete
fekete	fehér
fehér	vörös
vörös	fehér
zöld	fehér
narancs	fehér
vörös	zöld

I. táblázat

Bár ezek a színkombinációk erősen kontrasztosak, némelyikük vizuálisan nem vonzó, különösen képernyőn megjelenő programok esetében. A legjobb az, ha mindenki saját magának kikísérletezi ki, hogy milyen színeket használ az általa készített programhoz.

A gyors számítógépek megjelenését követően elterjedt az erőteljes színhasználat. Néhány képernyőn megjelenő oktatóprogramot CD-ROM-on szállítanak, másokat viszont az Interneten keresztül. A 24 bites színábrázolás a grafika méretének lényeges megnövekedését okozta, s ez által az adatátviteli sebesség (helyesebben lassúság) hirtelen komoly problémává vált. Ha az oktatóprogramot várhatóan az Interneten keresztül terjesztik célszerű inkább 8 bites színábrázolást használni, hacsak nyomós indok nem szól a 24 bites színmélység mellett. Intranetes használatot ez a megjegyzés nem érint, hiszen ez esetben a kapcsolat sokkal gyorsabb.

1.7 Grafikus és interfész útmutató

A képernyőn megjelenő programok tartalmának illusztrációjára használt grafika használatának elvei hasonlóak a szöveg és szín használat elveivel. Finom vonalakkal, de egymást ütő színekkel és rossz elhelyezéssel elkészített grafika nem alkalmas jó információközvetítésre. Nyomtatásban jól mutató grafika nem feltétlenül felel meg képernyőn megjelenő programokban. A szövegelemekhez hasonlóan a megtervezett grafika jó használhatóságot eredményez. Jól mutató színek kombinációinak használata a grafikában is elengedhetetlen.

A képernyőn megjelenő programok egy előnye, hogy a képernyőterület akárhányszor újrahasználható. Míg nem célszerű egy képernyőre túl sok információt összezsúfolni, ugyanannak az információnak több képernyő között való szétszórása csak minimális többletköltséggel jár.

A Macintosh és a Windows grafikai felhasználói felületeinek (GUI) megjelenése óta a grafika már nem korlátozódik a szövegitillusztrációra. Egy jól megtervezett grafikus felület egységes és kiszámítható viselkedést biztosít a rendszer által megjelenített objektumok számára, így a felhasználó felfüggeszti kételkedését, és elkezdi valóságnak tekinteni a képernyőn ábrázoltakat, a manipulálható objektumokat pedig, mint fizikai dokumentumokat, nyomógombokat, vagy eszközöket kezeli. Fontos, hogy olyan grafikát alkalmazzunk, amelyek jól megfelelnek ennek a célnak. A képek tárolási módjától függetlenül, a vizuális memóriánk nem pontos. Ezért az egyszerű, elkülönülő, egységes grafika a felület részeként való használatára kell törekednünk. Ugyanis a képernyőn megjelenő programokban túl sok részlet vesz el. A grafikának a lehető legegyszerűbbnek kell lennie.

2. FEJEZET

A Multimédia oktatási anyagok fejlesztésének pedagógiai szempontjai

2.1 Pedagógiai alapelvek

Mindazt, amit látunk és hallunk arra épülve értünk meg, amit már tudunk. A tanulást az teszi értelmessé, ha az új ismeretek szervesen összekapcsolódnak azzal, amit a tanuló már tud. Ez kognitív pszichológiai megfogalmazásban azt jelenti, hogy a tanuló hosszú idejű memóriájában új kognitív sémák és séma-kapcsolatok alakulnak ki. Ebből következik, hogy a tananyag-fejlesztés kezdetekor ismerni kell a tanuló alapjait.

A tanulók akkor tanulnak leghatékonyabban és legkönnyebben, ha a tananyag szerkezete és tartalma megfelel egyéni tanulási stílusaiknak. A hatékony tanuláshoz szükséges az anyaggal való aktív foglalkozás. Ennek érdekében a Multimédia tananyag vonzó és kísérletezésre ösztönző kialakításával bátorítani kell a tanulót a tananyaggal való aktív foglalkozásra, segíteni kell az elsajátított anyag emlékezetbe rögzítését és eszközöket kell biztosítani olyan tanulási stratégiák kialakításához, amelyekkel az adott tanulási stílusból adódó viszonylagos gyengeségek leküzdhetőek.

2.2 A Multimédia oktatási anyagok fejlesztésének pszichológiai szempontjai

A pszichológia (lélektan) a lelki jelenségekkel (folyamatokkal, tevékenységekkel és állapotokkal) foglalkozó tudomány. Az ember pszichikus funkciói révén tájékozódik az őt körülvevő világról és ugyancsak pszichikus funkciói révén, egyrészt alkalmazkodik ahhoz, másrészt aktívan alakítja is azt. [13]

A pszichikus funkciók jelentős része az információfeldolgozást szolgálja, amelynek az élet legtöbb területén, így a biztonságos munkavégzésben is kiemelt jelentősége van.

2.2.1 Az egyéni tanulási stílusok figyelembe vétele

Az egyéni tanulási stílusok figyelembe vétele meghatározó jelentőségű lehet, a Multimédia oktatási anyag használhatóságában. A tanulási stílusok figyelembe vételére a fejlesztés során általában a következő három lehetőség van:

- Csoportos felmérés és kiegyensúlyozott megtervezés. Ennél a - legegyszerűbben megvalósítható - módszernél a tanulói célcsoport egészét, vagy annak egy reprezentatív mintáját mérjük fel egyéni tanulási stílusaik szempontjából és ezt követően tervezzük meg a Multimédia oktatási anyag felhasználói felületét úgy, hogy az a tanulók nagy többségének megfelelő legyen. A tanulóknak a használat egyes pontjain választási lehetőségük van.
- Tanulási stílushoz rendelt verziók elkészítése. Ennél az - előzőnél nagyobb ráfordításokat igénylő - eljárásnál a Multimédia oktatási anyag két vagy több olyan verzióját készítjük el, amelyek megfelelnek egy-egy konkrét tanulási stílushoz tartozó tanulói csoport igényeinek.
- Intelligens oktató rendszer fejlesztése. Ebben az esetben a számítógép a rendszer használata közben - az adott tanuló kezdeti választásai vagy egy beépített teszt-modul alapján - felméri a tanuló egyéni tanulási stílusát és a továbbiakban ennek alapján választja ki az anyag számára legmegfelelőbb prezentálási módjait. Ma még ez az eljárás igényli a legnagyobb fejlesztési ráfordításokat, de ugyanakkor ez alkalmazkodik legrugalmasabban a tanuló egyéni tanulási stílusához. Megjegyzendő, hogy a technológia fejlődésével ez az igényesebb módszer is egyre inkább elérhető lesz a fejlesztők számára.

Behatórrolja az alkalmazható lehetőségeket természetesen az is, hogy az adott Multimédia oktatási anyagot

- különálló vagy hálózatba kapcsolt gépen használjuk, illetve
- a tanítási-tanulási folyamat irányított vagy nyitott.

2.2.2 Az információfeldolgozás

Az emberi agy féltékéi sem anatómiaiailag, sem funkcionálisan nem tekinthetők egymás pontos tükörképeinek. A jobb féltéke a balnál lényegesen fejlettebb vizuális képességekkel, tér- és mintaérzékeléssel rendelkezik, és a színeket is pontosabban azonosítja. [13] A Multimédia oktatási anyagok fejlesztése során az a célunk, hogy a tanuló tanulását

- egyfelől a megfelelő szövegekkel, hipertexttel, emberi beszédhanggal, képletekkel, formulákkal a nyelvi, verbális, logikai, analitikus képességek oldaláról,
- másfelől pedig a különböző képi információkkal, mozgóképekkel, térbeli ábrázolásokkal és alkalmasan választott színek alkalmazásával a téri, globális, érzelmi, szintetizáló képességek oldaláról is megtámogassuk.

2.2.3 A tanuláspszichológia alapkérdése

A tanulás a viselkedés viszonylag állandó megváltozása a gyakorlás eredményeképpen. A jó Multimédia oktatási anyag a tanulótól a szükséges előismereten kívül csak azt az alapmotivációt várja el, hogy leüljön a számítógép elé, és interakcióba kerüljön a rendszerrel. Ettől kezdve a programnak hosszabb időn keresztül fent kell tartania a tanuló érdeklődését a sokoldalú és vonzó információközlés és a folyamatos hasznosítható visszajelzések révén.

A Multimédia oktatási anyagok fejlesztésének szempontjából a következő lépéseket kell hatékonyan elvégezni:

1. Az elsajátítandó anyag prezentálása
2. A figyelem megragadása
3. Az anyag megrögzítése
4. Az anyag felidézésének segítése

2.3 Az egyéni tanulási stílusok

A tanulási stílusok azok a kognitív stílusok, amelyeknek az egyén tanulási tevékenysége során meghatározó jelentősége van. Minthogy a tanulási stílusok egyúttal kognitív stílusok is, azok ugyanúgy jellegzetes egyéni működésmódok, mint bármely más kognitív stílus. A tanulási stílusok tehát az egyén rögzített adottságai, amelyek az élet során gyakorlatilag nem változnak. A tanulási stratégiák ezzel szemben az információfeldolgozás megtanulható módjai, amelyek fejleszthetők és így megváltoztathatók. Megfelelő tanulási stratégiák kifejlesztésével például a tanuló olyan tanulási feladattal is eredményesen megbirkózhat, amely egyébként nem felelne meg veleszületett tanulási stílusának. Egyéni kognitív és tanulási stílusaink alakították ki a világ történéseihez való viszonyunkat, a világ megismerésére kialakított megközelítésmódjainkat. Mindenki tanulási stílusában vannak viszonylag erős és gyenge pontok, amelyek megismerése és tudatosítása rendkívül fontos. Ha ugyanis sikerül tudatosítani a gyenge pontokat, akkor kifejleszthetők olyan tanulási stratégiák, amelyek segítségével ezek a hátrányok megszüntethetők vagy jelentősen csökkenthetők.

A tanulási stílusok kérdésével azért foglalkozom a Multimédia oktatási anyagok kapcsán, mert ezek a korszerű oktatási anyagok csak akkor képesek a potenciálisan bennük rejlő lehetőségek megvalósítására - egy rendkívül intenzív és hatékony tanulási folyamat támogatására - ha azok olyan ismeret- és tudásátadási formákat kínálnak a tanulónak, amelyek összhangban vannak a tanuló tanulási stílusával. Ez magyarázza azt is, hogy a tanulási stílusok iránti érdeklődés a legutóbbi időkben erősen megnövekedett és az ezzel kapcsolatos kutatások egyre nagyobb támogatásokat kapnak.

3. FEJEZET

A Flash keretrendszer

3.1 A Flash 5 keretrendszer bemutatása

A Flash olyan kevert alkalmazás, amely egyetlen más programmal sem hasonlítható össze. A felszínes szemlélő számára úgy tűnhet, egy egyszerű webes bitkép-kezelő és egy vektoros rajzprogram (amilyen például a Macromedia Freehand vagy az Adobe Illustrator) keverékével állunk szemben. Mindez természetesen igaz is a Flashre, amely azonban ennél sokkal többre képes. A Flash interaktív multimédia-szerző program. Ezenkívül kifinomult animációs alkalmazás, amely igen sokféle animáció készítésére alkalmas. A Flash 5 saját parancsnyelvvvel rendelkezik (ez az ActionScript), melynek révén filmjeinket sokféle módon összekapcsolhatjuk az XML, HTML és más tartalmakkal.

Mire képes tehát ez a Flash nevű keverékprogram? Ezt a kérdést azok a fejlesztők tudják megválaszolni, akik a programmal dolgoznak. A Flash tehát összetett alkalmazás, amely szinte mindenre alkalmas. Ha szeretnénk jobban megérteni eme erőmű működését, fel kell tennünk a kérdést: milyen alkotóelemekből áll ez a keverék? Ha az egyes részeket sikerült elkülönítenünk egymástól, meg kell vizsgálnunk azt is, mire képesek az egyes programrészek. Ezekre a kérdésekre próbálok választ adni ebben a fejezetben.

3.2 Bitkép-kezelő

Az igazat megvallva, a Flash képfeldolgozó képességei elég korlátozottak, ezért helyesebb a Flash ezen összetevőjét bitkép-kezelőnek, és nem képfeldolgozónak hívni. A bitképek négyzethálóba rendezett, egymástól független képpontokból állnak. Ezeket egyenként kell tárolni, így sok memóriát foglalnak és nagy fájl méretet eredményeznek. További hátrányuk, hogy nagyításuk és kicsinyítésük nem végezhető el minőségromlás nélkül, ami nagyításkor még inkább jelentkezik, mint

kicsinyítéskor. A fényképminőségű képek tárolásához és megjelenítéséhez viszont elengedhetetlen a bitképes formátum.

3.3 Vektoros rajzoló

A Flash legnagyobb részét egy vektoros rajzolóprogram alkotja, amely a Macromedia Freehandhez és az Adobe Illustrator-hoz hasonló képességekkel rendelkezik. Ezek a programok nem különálló képpontokból rakják össze a képet; az alakzatokat koordinátaikkal megadott pontjaik határozzák meg. A pontokat összekötő vonalakat görbéknek nevezzük; ezek görbületét a pontokhoz rendelt vektorok írják le. Mivel ebben az esetben a képeket matematikai leírásukkal adjuk meg, a vektoros ábrázolás tömörebb, ráadásul a képeket tetszőlegesen nagyíthatjuk, mindenféle mellékhatások nélkül.

3.4 Vektoranimáció-készítő

A Flash vektoranimációs része semmilyen más programhoz nem hasonlítható. Bár a Flash képes bitképek kezelésére is, az általa használt fájlformátum vektoros ábrázolást tartalmaz. Más animációs programokkal ellentétben a rövid és tömör vektoros leírást használja a képkockák megjelenítésére, nem pedig képpontok millióit. Míg a bitkép alapú animációs programok esetében a teljes képkockát tárolnunk kell, a Flash a vektoros leírásból helyben állítja elő a képeket. Ez óriási előnyt jelent a Flash animációk és alkalmazások hálózaton (például az interneten) történő továbbítása során.

3.5 Szerzői környezet

Azt mondhatjuk, hogy a Flash alapja egy multimédia-szerző program vagy környezet. A program segítségével olyan filmeket állíthatunk elő, amelyek többféle médiumot (hangot, álló- és mozgóképet) tartalmaznak. Ugyanakkor a Flash interaktív multimédia-program is egyben, mivel a kész

film olyan parancsokat tartalmazhat, amelyek segítségével a néző befolyásolhatja a lejátszás menetét.

3.6 Animációs program

Talán nincs is olyan multimédia-szerzői program, melynek ne lenne animációszerkesztő része. A Flash sem kivétel ez alól. Valójában ez alkotja a Flash gerincét. Az animációk az alábbi módon épülnek fel:

- A film tetszőleges számú felvonást (Scene) tartalmazhat, amelyeket a lejátszáshoz sorba rendezünk. A Flash a jeleneteket az elsőtől az utolsóig sorban játssza le, hacsak a forgatókönyv vagy a néző nem változtat ezen.
- A jelenetek korlátlan számú rétegből (Layer) állhatnak, amelyek a jelenet képkockáin egymás fölött jelennek meg és akár el is takarhatják egymást. Az egyes rétegek jellemzően összetartozó objektumokat tartalmaznak: a Tom és Jerry egyik jelenetében a leghátsó rétegen a falat látjuk egy szobanövénnyel és egy egérlyukkal. Előtte egy másik rétegen Tom sajtot rak a lyuk elé, a harmadikon pedig a sajt után szimatoló Jerry fog előbújni.
- A rétegeken belül elhelyezett objektumoknak is megvan a sorrendje. Az önálló (egymáshoz nem rendelt) vonalak és alakzatok mindig az alsó szinten (Bottom level) vannak. Fölötte, az átlapoló (Overlay) szinten található a bitképek, a szövegek, a csoportok, az összetett objektumok, a szimbólumok és a szimbólumpéldányok.
- Az animáció során a képkockák felelősek az időzítésért. Minden réteg egy vagy több képkocka-sorozatból áll, így egy adott pillanatban látható kép az egyes rétegek megfelelő képkockáinak egymásra rajzolásával áll elő.
- A képkockáknak két típusa van: statikus képkocka, illetve kulcskocka. Minden réteg kulcskockával kezdődik, amely azonban üres is lehet. A

statikus képkockák egyszerűen az előző kocka tartalmát ismétlik. A kulcskockák hordozzák az új vagy megváltozott tartalmat (esetleg üresek maradnak). Az animáció kétféleképpen történhet. Az egyik esetben folyamatosan, kockáról-kockára módosítjuk a képkockák tartalmát, a másikban pedig a Flash számolja ki a két, általunk megadott kulcskocka közötti átmenetet.

3.7 Programozói és adatbázis-felület

A Flash 4-ben a Macromedia egy olyan korlátozott képességekkel rendelkező, de nagyon hatékony programozási nyelvvel egészítette ki termékét, amely lehetőséget adott arra, hogy interaktívvá tehesük Flash filmjeinket. Sőt, ezen lehetőségek révén a Flash alkalmassá vált arra, hogy kifinomult interaktív alkalmazások adatbázisfelületeként működhessen. És ez még csak a Flash 4. A Flash 5 olyan változásokat hozott, amelyek tulajdonképpen korlátlaná tették a program lehetőségeit. Már csak a fantáziánkon múlik, mit csinálunk a Flashsel:

- Az ActionScript, amely korábban meglehetősen korlátozott parancsnyelv volt, mára olyan erőteljes nyelvvé nőtte ki magát, amely összeegyeztethető a Flash 4-gyel, viszont teljes mértékben a JavaScriptre épül. Ha pontosabban szeretnénk fogalmazni, akkor a Macromedia az ECMA-262 szabványnak megfelelően fejlesztette ki az ActionScriptet. Ezt a szabványt a JavaScript alapján dolgozták ki, annak érdekében, hogy nemzetközileg szabványosíthassák a nyelvet.
- A program XML támogatással egészült ki. Flash filmjeink most már XML adatokat is képesek fogadni, illetve küldeni.
- A Math objektum bevezetésével az alkalmazható matematikai műveletek köre jelentősen bővült (használhatjuk például a szinusz, koszinusz és tangens függvényeket is).

- A Flash szimbólumok színeit és hangjait már parancsfájlokon keresztül is vezérelhetjük.
- A könyvtáron keresztül azokhoz a hangokhoz és filmklipekhez is közvetlenül hozzáférhetünk, amelyek nem jelennek meg az időszalagon.

3.8 A Flash 5 filmek alkalmazási területei

A feladattól függően egy Flash film igen sokféle lehet. A Flash filmek nagy utat tettek meg, ráadásul nagyon rövid idő alatt. Egy rövid listában összefoglaltam, mi mindent is készíthetünk a Flash 5-tel:

- Figyelemfelkeltő animáció egy weblapon.
- Interaktív térkép.
- Interaktív adatbázis, amely kiszolgáló-oldali alkalmazásokkal áll kapcsolatban.
- Élő, többfelhasználós játék vagy csevegés, amely valós idejű kapcsolattartást tesz lehetővé az Internet-használók között.
- Világhálós zenegép, amely képes MP3 fájlok lejátszására.
- Önálló webes alkalmazások.
- Önálló multimédiás alkalmazások (játékok, bemutatók, oktató anyagok).
- Teljes honlapok, mindenféle HTML alapú grafika vagy szöveg nélkül.
- Interaktív művészeti bemutatók, amelyek háromdimenziós transzformációkat is tartalmazhatnak és több felhasználót vonhatnak be egyszerre az élménybe.
- Jó minőségű bitképeket és hanganyagokat tartalmazó webes alkotások.
- Interaktív QuickTime Flash filmbemutatók.
- Önálló bemutatók vagy diavetítések CD-ROM-on vagy hajlékonylemezen.
- Téveminőségű rajzfilmek.

A listából egyértelműen kiderül, hogy ha a terv kész, akkor azt valószínűleg meg fogjuk tudni valósítani a Flashben.

4. FEJEZET

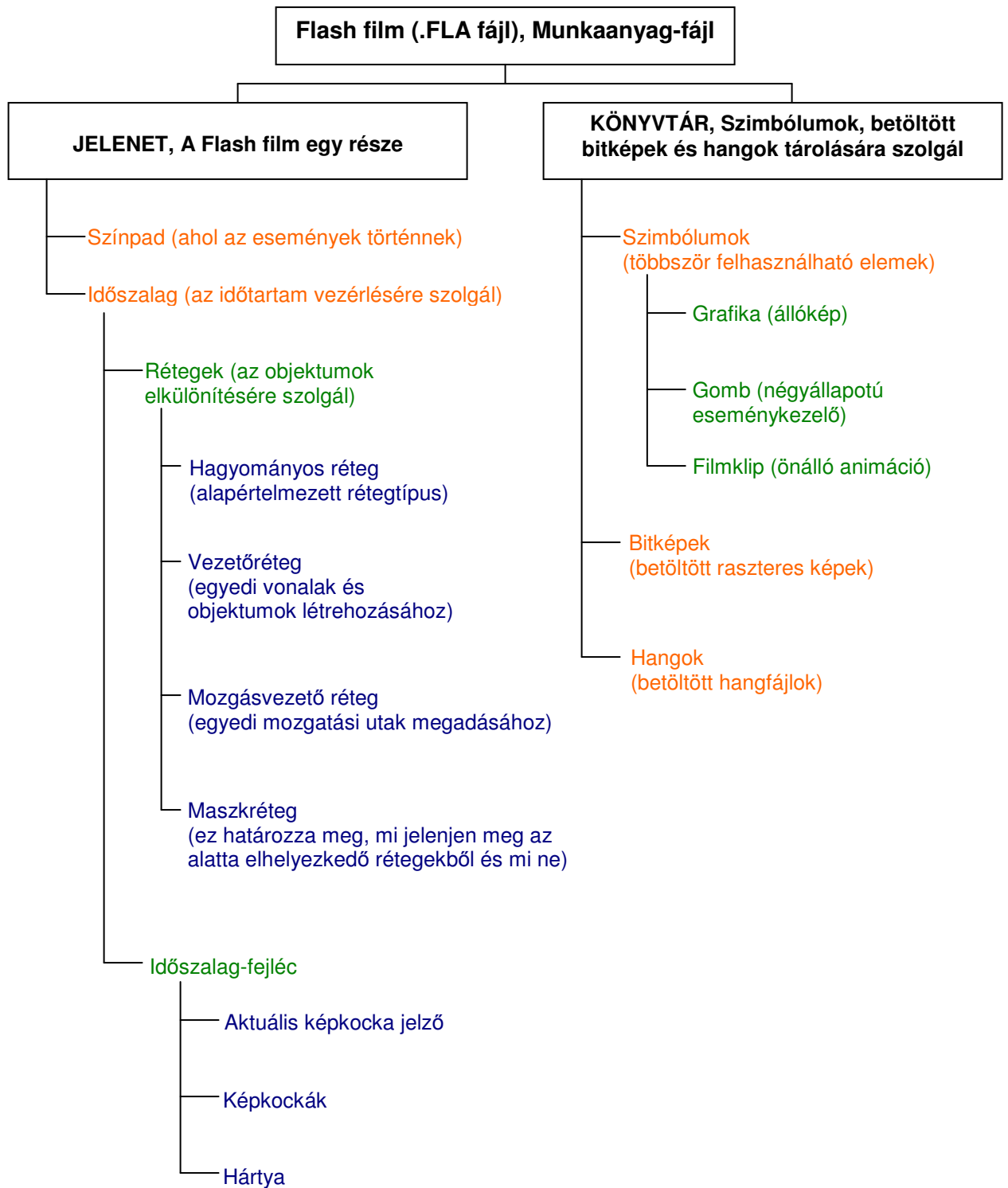
A projekt során használt médiaanyagok bemutatása

4.1 Az interaktív Flash anyagok megtervezése

Az interaktív anyagok létrehozása előtt ajánlott előzetes tervet készítenünk. Ez a későbbi munka során kifizetődik. Fordítsunk elegendő időt a terv elkészítésére, mert ellenkező esetben csak magunkat csapjuk be. Minél jobban néz ki a terv papíron, annál jobb lesz a végeredmény.

Többféle tervezési módszer is létezik. Általában véve mindennek az alapja az, hogy megtanuljuk, hogyan lehet az interaktív elemeket egyszerű folyamatábrákkal ábrázolni. Az *1. ábra* egy ilyen folyamatábrán keresztül mutatja be a Flash környezet felépítését.

Az Flash filmek jelenetekből állnak, a jelenetek pedig kulcskockákat tartalmaznak. Az azonban nem látható, mennyire hatékonyá teszi a munkánkat, hogy a Flash könyvtárakat megoszthatjuk a munkaanyagok (FLA fájlok) között, illetve hogy a LoadMovie utasítás segítségével filmjeinket egy közös szülőfilmhez fűzhetjük. Mielőtt azonban ilyen bonyolult dolgokkal kezdenénk el foglalkozni, mindenképpen meg kell ismerkednünk a Flash filmek és az SWF filmek közti eltérésekkel.

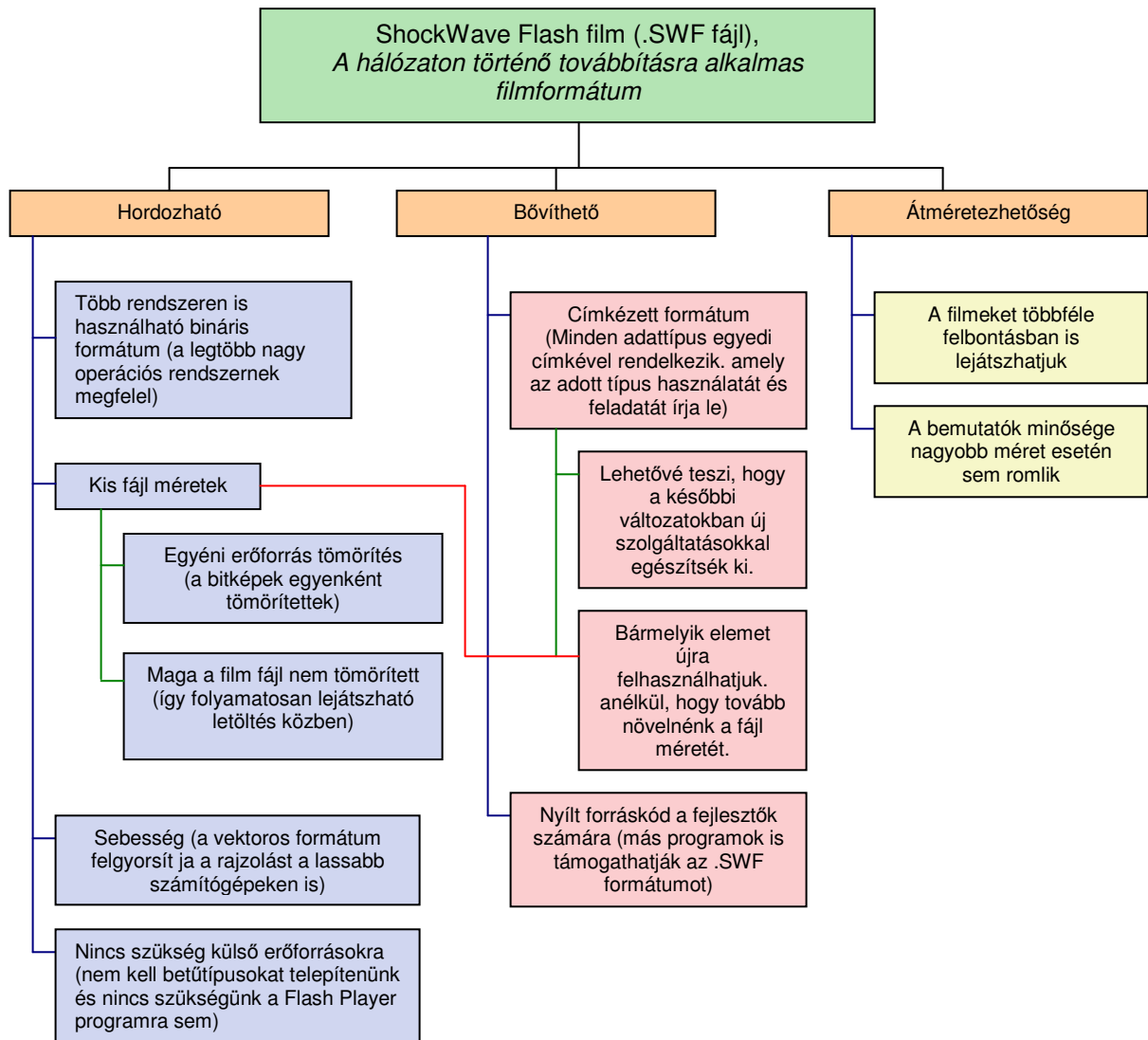


*1. ábra
Flash környezet elemei.*

4.2 Flash film fájl típusok

A Flash film fájlok (FLA) szerkezetét a hatékony szerzői környezet igényei szerint alakították ki. Ebben a környezetben a tartalmat jelenetekbe rendezhetjük, a jelenetek sorrendjét pedig a tervezés során bármikor módosíthatjuk. A rétegek segítségével egyszerűen elválaszthatjuk egymástól a különböző grafikai elemeket egy jeleneten belül. Az időszalag a kulcskockák, a mozgás- és alakzatátmenetek, a címkék és megjegyzések megjelenítésére szolgál. A betöltött bitképek és hangok mindegyike a könyvtárba kerül (amely megosztható a Flash filmfájlok között). A könyvtárban tárolt fájlok (vagy szimbólumok) minősége megegyezik az eredeti fájlok minőségével.

Amikor Flash filmünket .SWF (Small Web Format) formátumban készítjük el, a program a fenti adathalmaz nagy részét elhagyja, hogy a fájlok mérete a lehető legkisebb lehessen. Az eredeti .FLA fájl szinte minden eleme megváltozik valamilyen módon. A könyvtárban lévő elemek abban a kulcskockában tárolódnak, ahol elsőként használjuk azokat (azok az elemek, amelyeket sehol sem használtunk, nem kerülnek bele a fájlba). A többször felhasznált elemek csak egyszer kerülnek be az .SWF fájlba. A program a rétegeket és a színeket az .FLA fájlban meghatározott sorrendben összevonja. Más szóval az .SWF fájl egyetlen jelenetből áll és egyetlen réteget tartalmaz. Magukat az .SWF fájlokat a Flash nem tömöríti, a különböző bitképeket és hangokat azonban -a beállításoktól függően- igen. (2. ábra)



2. ábra

A Shock Wave Flash filmek felépítése

4.3 Szövegek kezelése

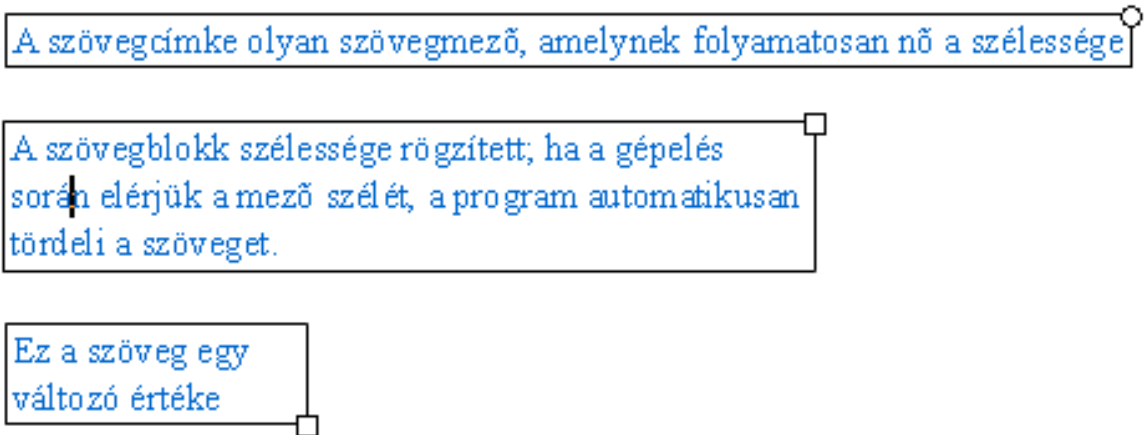
A Flash filmek sokszor szöveges formában jelenítik meg a különféle adatokat. Szöveges információink megjelenítésére különféle betűtípusokat használhatunk. Bár a használt betűtípusok összes adata bekerül a filmbe, előfordulhat, hogy a betűk egy másik rendszerben nem megfelelő módon jelennek meg. Ha például egy adott gépen nem telepítették a munkánk során használt készletet, akkor az biztosan nem úgy jelenik majd meg, ahogy szeretnénk volna. A problémát az okozza, hogy a Flash, bár a szerkesztőben

képes megjeleníteni a betűtípusokat, nem ismeri fel a készlet körvonalait, így az adott betűtípussal írt szöveg nem is kerül bele az általunk készített projektbe. Az ilyen jellegű problémákat könnyen elkerülhetjük, ha csak a rendszer betűtípusait használjuk (`_sans`, `_serif`, `_typewriter`). Tudnunk kell azonban, hogy ezek nem valódi betűtípusok: ha ezeket használjuk, a Flash az adott rendszerben hozzájuk legközelebb álló típusokat használja. A `_sans` kiválasztásakor például általában az Arial vagy a Helvetica, a `_serif` esetében a Times vagy a Times New Roman, a `_typewriter` esetében pedig a Courier típus jelenik meg. Mivel filmünk ilyenkor a rendszer alapértelmezett készleteit használja, nincs szükség arra, hogy a betűtípust a filmbe építsük, így kisebb lesz a fájl méret is. Ez hozzájárul az oktatóanyag Interneten való publikálásához is, mivel a kisebb Flash filmek letöltéséhez rövidebb időre van szükség.

A rendszerbetűtípusokra jellemző, hogy mindig elérhetők, megjelenésük gyors, képük pedig lejátszóról lejátszóra kismértékben változhat. Akkor célszerű ilyen készleteket használni, amikor nem szeretnénk elsimítani a betűk széleit. A betűtípusok használatából eredő nehézséget úgy is elkerülhetjük, ha szövegeinket grafikus formában tároljuk. A grafikus formában tárolt szöveg sajnos jelentős mértékben növeli a fájl méretét, tehát ezzel a módszerrel óvatosan kell bánnunk. Ráadásul az átalakított szöveget nem is szerkeszthetjük többé, ezért az átalakítást csak akkor célszerű elvégezni, ha a szövegen már nem szeretnénk módosítani. Mivel a Flash vektorgrafikus program a legtöbb betűtípust gond nélkül képes beépíteni a filmbe. Az egyszerű szövegblokkok esetében ez azt jelenti, hogy a betűket nem kell bitképpé alakítanunk. A Flash által elkészített `.SWF` fájl tartalmazza a betűk megjelenítéséhez szükséges összes adatot.

A Flashben jelenleg háromféle szövegmezőt használhatunk (3. ábra):

- szöveg címke
- szöveg blokk
- szerkeszthető szöveg



3. ábra

A Flash háromféle szövegmezője (felülről lefele): szöveg címke, szöveg blokk és szerkeszthető szöveg.

4.3.1 Szöveg címke

A szöveg címke olyan szövegmező, amelynek folyamatosan nő a szélessége, miközben újabb szövegeket adunk hozzá. A szövegmező jobb felső sarkában egy kör alakú kart láthatunk. Ha nem alkalmazunk sortörést, a gépelés során a mező szélessége folyamatosan nő. A szövegmező a film keretein kívül is tovább növekszik.

4.3.2 Szöveg blokk

Ha a filmben elhelyezett szövegmező méretét megváltoztatjuk, a mező szöveg blokká válik. A szöveg blokk szélessége rögzített; ha a gépelés során elérjük a mező szélét, a program automatikusan tördeli a szöveget. Ebben az esetben a szöveg doboz szélessége állandó marad, magassága azonban természetesen a szöveg hosszától függően változhat. A

szövegblokk típusú mezők jobb felső sarkában egy négyszögletes kart láthatunk.

4.3.3 Szerkeszthető szöveg

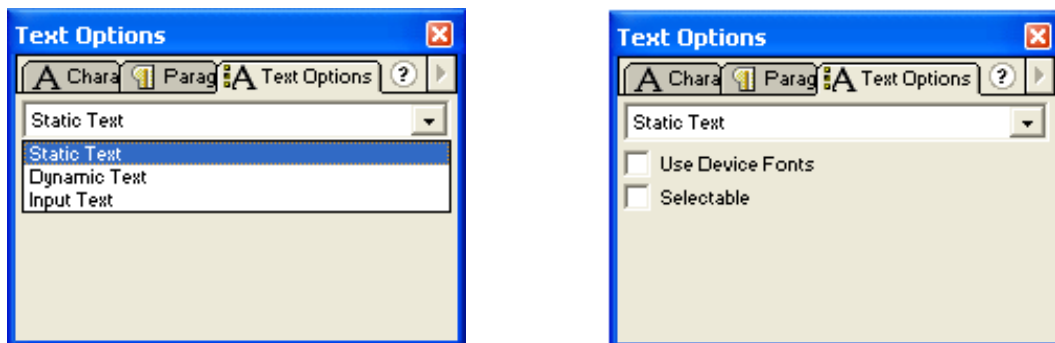
A szerkeszthető szövegmezők tartalma módosítható. Ez azt jelenti, hogy az ilyen mezőket felhasználhatjuk jelszavak megadásához vagy űrlapok készítéséhez is. A szerkeszthető szövegmező átméretezésére szolgáló négyszögletes kart a mező bal alsó sarkában találhatjuk.

4.3.4 Statikus (nem szerkeszthető) szövegek

A Flashben létrehozott vagy oda bemásolt szövegblokkok alapértelmezés szerint nem szerkeszthetők. A CD készítése során előfordult azonban, hogy egy szövegmezőt nem csak egy adott szöveg megjelenítésére kellett használni, hanem lehetővé tettem a felhasználó számára, hogy kijelölhesse és lemásolhassa annak tartalmát. Elképzelhető az is, hogy egészen pontosan szeretnénk beállítani, hogyan jelenjen meg egy adott szövegrész. Ilyenkor jó hasznát vehetjük a Text Options tábla Static (Statikus) beállításának (4. ábra):

Selectable (Kijelölhető).

Use Device Fonts (Rendszer-betűtípusok)



4. ábra

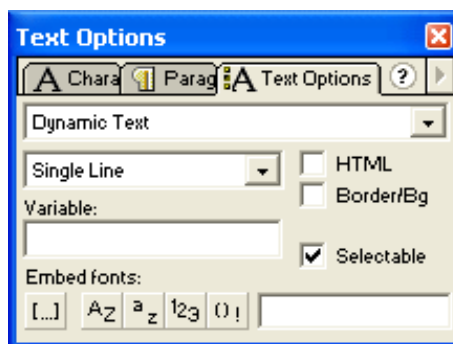
A text Options tábla.

4.3.5 Szerkeszthető szöveg mezők

A Flashben alapvetően kétféle módon hozhatunk létre szövegeket: szövegdobozként vagy szövegmezőként. A szerkeszthető szövegmező tulajdonképpen egy üres ablak, amelyhez egy változót rendelünk. Amikor a Flash film szöveget vagy adatot kap, az adatok ehhez a változóhoz kerülnek és ez gondoskodik arról, hogy megjelenjenek a megfelelő ablakban. A Flash 5-ben kétféle szerkeszthető szövegmezőt használhatunk: Dynamic Text (Dinamikus szöveg) és Input Text (Beviteli mező).

4.3.6 Dinamikus szövegmezők

Általában egy kiszolgáló tölti fel tartalommal. Ilyen mezőket kell használnunk, ha például tőzsdei árfolyamokat, sporteredményeket vagy időjárás-jelentést szeretnénk megjeleníteni. A lehetséges felhasználásnak kizárólag a fejlesztők fantáziája szabhat határt. A tartalom származhat adatbázisból, kiszolgáló-oldali alkalmazástól, másik Flash filmből, vagy az aktuális film másik részéből. A dinamikus szövegmezőket dinamikusan változó tartalom megjelenítésére használjuk, , miután a Text Behaviour (Szöveg viselkedése) mezőből a Dynamic Text (Dinamikus szöveg) lehetőséget választottuk. (5. ábra)



5. ábra

Amikor a felhasználó kitölt egy űrlapot, válaszol egy webes felmérés kérdéseire vagy megad egy jelszót, beviteli szövegmezőket használ.

4.4 Képek, rajzok, animációk a Flasben

A projekt készítése során objektumokat hoztam létre a rajzeszközök segítségével, majd az objektumokat animációra készítettem, az úgynevezett idősv alkalmazásával. A film egyes elemeit az úgynevezett rétegekkel (layers) választottam külön egymástól. Az ActionScript segítségével lehetőség nyílt a film eseményeinek szabályozására és az interaktivitás programozására.

Alapvetően vektorgrafikus képekkel dolgoztam, de a Flash képes kezelni a bittérképes (raszteres) grafikát is. A legjelentősebb eltérés a bittérképes és a vektor alapú grafika között akkor érzékelhető, amikor a kép a képernyőn megjelenik. A bittérképes kép megjelenése függ a felbontástól. Az Interneten a bittérképes grafikák szabványos felbontása 72 képpont inchenként. Amikor egy ilyen képet felnagyítunk, a megjelenítő program a hiányzó részeket adatokkal pótolja. Amikor a képet kicsinyítjük, a gazdaprogram képpontokat távolít el belőlük. Ezek a változtatások szükségszerűen torzításhoz vezetnek.

A vektor alapú képeket a program matematikai képletek és eljárások alapján rajzolja meg. Amikor egy vektoros grafikát átméretezünk, valójában csak az eredeti képlet egy vagy több paraméterét változtatjuk meg, majd a módosított képlet alapján a képet újra megrajzoljuk. Például, amikor megnöveljük egy vektorokkal leírt kör méretét, igazából a kör sugarát változtatjuk meg, és az így módosított alakzat torzítás nélkül jelenik meg. Az ilyen típusú képek különösen jól méretezhetők, még akkor is, ha eredeti méretük sokszorosára nagyítjuk őket. Gyakorlatilag csak akkor fordulhat elő észrevehető mértékű torzulás vektor alapú grafikák esetében, amikor összetett színátmenettel rendelkező képet méretezünk át. A Flash vektorokat használva készíti el a képeket, emellett képes importálni bittérképes grafikákat is. A kétféle formátum együttes kezelhetősége igen nagy rugalmasságot biztosít. *(II. táblázat)*

Abittérképes és a vektoros grafikák összehasonlítása

Bittérképes grafikák	Vektoros grafikák
Nagyobb fájlméretű.	Kisebb fájlméretű.
Jó választás fényképek esetén.	Fényképek esetén rossz határfokú.
Nehéz vonalakat és alakzatokat szerkeszteni.	A vonalak és alakzatok könnyen szerkeszthetők.
Az internetböngésző programok bővítménymodulok (plug-ins) nélkül is felismerik a JPEG és GIF képeket.	Az internetböngésző programoknak általában bővítménymodulokra van szükségük a vektor alapú grafikák megjelenítéséhez.
Kedvezőtlen választás animációk készítéséhez.	Tökéletes választás animációk készítéséhez.
Készítését könnyebb megtanulni.	Készítését nehezebb elsajátítani.
Átméretezésük torzítás hoz vezet.	Átméretezésük nem vezet torzítás hoz.

II. táblázat

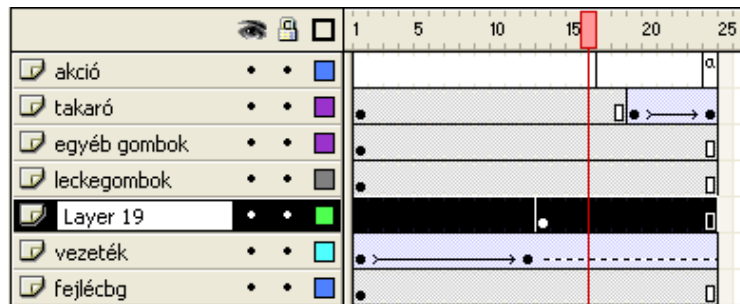
4.4.1 Az animáció és az állóképtervezés kapcsolata

A mozgóképek olyan módon köti le a nézőközönség figyelmét, amelyenre az állókép képtelen. Az animáció csak egy árnyalatnyival különbözik az állóképtől. Az állóképek önmagukban közvetítenek egy üzenetet. Az animációs filmek viszont az idő múlását egy behatárolt, lineáris térben kombinálják az üzenet továbbításához. Mind az álló, mind a mozgóképeket meg kell tervezni, de a tervezési folyamat és az eljárás eltérő. Az időben egymás után megjelenített képek egy ritmust generálnak. Az animáció képkockáinak van egy üteme, amit a képkockák megjelenítési idejének szabályozásával mérhetünk. Természetüknél fogva, a mozgóképek szórakoztatóbbak, mint az álló grafikák, megnövelve a tudatküszöb alatti üzenetátvivő képességüket. Az emberek sokkal tovább nézik a mozgó dolgokat, mint a nyomtatott anyagot, mivel az előbbi esetében nem kell annyira koncentrálniuk, mint az utóbbinál. A kellemes formában bemutatott mozgó grafikák, a hipnotikus hatásuk révén sokkal mélyebb benyomást tesznek a befogadóra, mint a nyomtatott képek. Hiszen a mozgás egyetemes nyelv, melyet mindenki megért.

4.4.2 Animáció különböző időszekvenciákon, különböző rétegeken

A mozifilmeket tipikusan 24 fps (frame per secundum) zársebességgel veszik fel, a rángatásmentes mozgás illúzióját keltve szemünkben. A Flash-animáció alapértelmezett képkockasebessége 12 fps, körülbelül fele a filmekének. A Flash-animációt a sebesség állításával lassíthatjuk, illetve gyorsíthatjuk. A rétegek a mozgó grafikák szerves részét képezik. Az egymással párhuzamosan történő események rétegei egyfajta ritmust hoznak létre, amelyek kiegyenlítik a film fő fókuszát. A valós világot felfogó érzékelésünk azt sugallja, hogy a mozgások és az események a múlt időben egymás után, illetve egymással párhuzamosan történnek. A film és a videó rendelkezik a kétdimenziós síkban a valósvilágbeli, térbeli érzet keltésének képességével. A Flashben ennek megfelelői a rétegek. A rétegek lehetővé teszik, hogy egy vagy több objektumot egyidejűleg mások elé vagy mögé helyezzünk. A rétegek az animáció egyféle takarási sorrendjét állítják fel. Régebben az animátorok kézzel rajzolt animációkban cellaíveket használtak. Mivel ezek átlátszóak voltak, időmegtakarítás végett bizonyos cellákat újra felhasználtak azokon a helyeken, ahol az elemek megegyeztek egymással. A Flash ugyanezt az elvet követi a rétegek esetében. A rétegek nem csupán a más rétegekhez viszonyított takarási sorrend meghatározásában játszanak szerepet, hanem az egyenletesen múlt időben képesek objektumokat vagy háttereket úgy kiterjeszteni, hogy azok látszanak egy rétegen, miközben egy másik rétegen más dolgok történnek.

Az összetett Flash-animációk rendszerint egynél több réteget tartalmaznak. Nem ritka a 35-40, vagy még ennél is több rétegű film. A nagyobb projektek komoly tervezési munkát igényelnek. A színtér előretervezésének folyamatában fontos a rétegek logikus elnevezése. Mint egy színházi szereposztás tagjainak, a Flash-film elemeinek is tudniuk kell a helyüket és szerepüket már a produkció kezdete előtt (6. ábra).



6. ábra

Az időszávon a rétegek nevét kijelölve és felülgépelve nevezhetjük el rétegeinket.

4.5 Hangok a Flashben

Ha gondosan megtervezzük a hangokat és a részletekre is odafigyelünk, a hanganyagok bemutatónk előnyére válnak majd. Lényeges, hogy a hang ne csupán mellékes kiegészítés legyen, közönségünknek igyekezzünk teljes érzéki élményt nyújtani.

A hang a levegőben terjedő rezgések által keltett, akusztikai érzés. A hangerőt a hanghullámok intenzitása, míg a hang magasságát -azt, hogy mély vagy magas hangot hallunk- a rezgések (hullámok) frekvenciája határozza meg. A hang rezgésszámának mértékegysége a hertz (Hz). Normális esetben az ember a 20-20000 hertzig terjedő frekvenciákat képes érzékelni. Ami fontos, hogy ne tévesszük össze a hangfrekvenciát a mintavételezés gyakoriságával. A hangminőséget és fájl méretet meghatározó tényezők közül a két legfontosabb: a mintavételezési frekvencia és a bitfelbontás.

4.5.1 Mintavételezési frekvencia

A hertzben mért mintavételezési frekvencia értéke határozza meg, milyen gyakorisággal veszünk mintát a hangból, amikor digitálisan rögzítjük azt. Az 1940-es évek végén Harry Nyquist és Claude Shannon fejtette ki azt a mintavételezési tételt, amely szerint a mintavételezési

frekvenciának legalább kétszer akkorának kell lennie, mint amekkora a jelben előforduló legmagasabb frekvencia értéke. Minél nagyobb ez az érték, annál jobb minőségű lesz a digitális hanganyag. A magasabb frekvencia általában gazdagabb, teljesebb hangélményt eredményez. A tétel szerint, amennyiben megfelelő módon szeretnénk visszaadni a 20 és 20 000 Hz közötti tartományt, legalább 40 kHz-es mintavételezési frekvenciával kell dolgoznunk. Ez az oka annak, hogy a hang CD-ket 44,1 kHz-es mintavételezéssel készítik, hiszen ez az eredeti hangot viszonylag pontosan adja vissza. Minél kevesebb mintát veszünk a hangból, annál kevésbé hasonlít majd a végeredmény az eredeti hangra, a hangfájl mintavételezési frekvenciájának csökkentésével viszont arányosan csökken a fájl mérete, ami előnyös lehet. Például egy 44.1 kHz-en rögzített 300 KB-os fájl mérete 22,05 kHz-es frekvencia mellett mindössze 150 KB lenne. A *III. táblázat* az általánosan használt mintavételezési frekvenciáknak a minőségre gyakorolt hatását mutatja.

A mintavételezés és a minőség

Mintavételezési frekvencia	Hangminőség	Lehetséges alkalmazások
48 kHz	Stúdióminőség	Digitális hordozóra rögzített hanganyagok.
44,1 kHz	CD minőség	Magas hanghűségű hang és zene.
32 kHz	CD-közeli minőség	Digitális videókamera.
22,05 kHz	FM rádió minőség	Rövid, jó minőségű zenei klipek.
11 kHz	Elfogadható minőség zenéhez	Hosszabb zenei klipek, jó minőségű beszédhang-felvételek, hanghatások.
5kHz	Elfogadható minőség beszédhez	Beszédhangok, egyszerű gombhangok.

III. táblázat

Miután a lejátszás minden hangkártyán 44,1 kHz-en történik, a számítógépen lejátszani kívánt hangok frekvenciájának is ehhez kell igazodnia. Számítógépes használatra ezért 11,025, 22,05 és 44,1 kHz-es mintavételezést alkalmazunk. Lehetséges, hogy a fejlesztésre szolgáló rendszeren és számos más gépen ezektől eltérő értékeket használva is

megfelelő hangot kapunk, de ez nem minden számítógépen biztosított, ezért jobb, ha az esetleges torz lejátszás elkerülése érdekében mindig tartjuk magunkat a fenti szabályhoz. A Flashben ez különösen fontos, mert a program azokat a hanganyagokat, amelyek mintavételezési frekvenciája nem a 11,025 többszöröse, átalakítja, így a hang mélyebb vagy magasabb lesz az eredetihez képest.

4.5.2 Bitfelbontás

A bitfelbontás szintén kulcsfontosságú tényező a hang minőségét illetően. A bitfelbontás (vagy bitmélység) az egyes hangminták leírásához használt bitek számára vonatkozik. A mennyiség hatványozottan nő, azaz egy 8 bites hangmintában 2^8 (256), egy 16 bitesben pedig 2^{16} (65 536) szintet különböztethetünk meg. A 16 bites hangminta tehát lényegesen több adatot tárolhat a hangról, mint egy ugyanolyan hosszúságú 8 bites. A 16 bites mintában lévő további információk révén minimálisra csökken a háttérzaj, így a végeredmény sokkal tisztább lesz, mint a 8 bites esetében. (IV. táblázat)

Néhány bitmélységet és azok minőségi jellemzői

Bitmélység	Hangminőség	Lehetséges alkalmazások
16 bit	CD minőség	Magas hanghűségű hang és zene.
12 bit	CD-közeli minőség	Digitális videókamera.
8 bit	FM rádió minőség	Rövid, jó minőségű zenei klipek.
4 bit	Elfogadható minőség zenéhez.	Hosszabb zenei klipek, jó minőségű beszédhang-felvételek, hanghatások.

IV. táblázat

A hangfájlokat megkülönböztetjük aszerint is, hogy egy vagy két (bal és jobb) hangcsatornát tartalmaznak-e. Az előbbi esetben monó, az utóbbi esetben pedig sztereó felvételtől beszélünk. A sztereó hangfájlok mérete kétszerese a monó fájlokénak, mivel ezekben kétszer annyi adatot kell tárolni. Amikor hanganyagokkal dolgozunk, fontos, hogy az anyag hosszát a

lehetőlegkisebbre csökkentsük. Ez azt jelenti, hogy le kell vágnunk az anyag elejéről és végéről a felesleges részeket (például a bevezető szünetet).

Íme egy egyszerű képlet, amelynek segítségével könnyen meghatározhatjuk egy adott hanganyag méretét (bájtban):

hanganyag hossza (mp) x mintavétési frekvencia hertzben kifejezve. x csatornák száma x (bitmélység / 8) = fájl méret

Ennek megfelelően egy 20 másodperc hosszúságú, 8 bites, 11 kHz-es, sztereó hangfájl méretét az alábbi módon számíthatjuk ki:

$20 \text{ mp} \times 11\,025 \text{ Hz} \times 2 \text{ csatorna} \times (8 \text{ bit} / (8 \text{ bit/bájt})) = 441\,000 \text{ bájt} = 430 \text{ KB}$

5. FEJEZET

A Flash projekt kivitelezése

Néhány papírra vetett vonal kivételével a kezdeti terveket is a Flashben terveztem, a program Pen (Toll) eszközének felhasználásával. Mivel a vektorgrafika mellett raszteres képeket is felhasználtam munkám során, szükségem volt egy raszteres alkalmazásra is. (PhotoShop 8 / Paint Shop Pro 9) A rajzeszközök segítségével hozzákezdtem a pizkozatok elkészítéséhez. Folyamatosan oda kellett figyelni azokra a tényezőkre, amelyek a végső fájl kinézetét befolyásolhatják. Nagy előny, hogy a grafikát, a tartalmat és a szolgáltatásokat is egy helyen készíthettem el. Ez azonban nagy odafigyelést igényelt, hiszen könnyen előfordulhat, hogy egy nem kellőképpen átgondolt tervezési lépéssel végképp sarokba szoríthatjuk magunkat. Előzetes tervezés nélkül a Flash projekt könnyen rémálommá válhat, az olyan előre nem látható tényezőknek köszönhetően, mint a nem lineáris navigáció vagy a fájl méret adta korlátok.

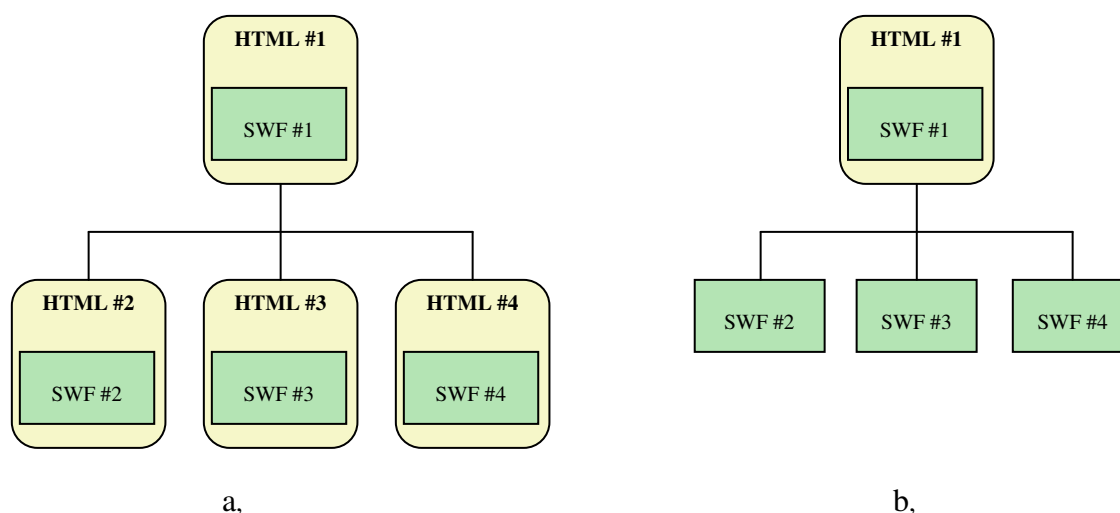
Miután elkészültem néhány pizkozattal és eldöntöttem melyikből fogok kiindulni, nekiláttam a minden részletre kiterjedő tervezésnek. A részletes terv a projekt olyan változata, ahol már működnek az animációk és a felület teljes pompájában megmutatkozik. Ebben a szakaszban három új tervet dolgoztam ki, ezek mind a pizkozat elemeire épültek. A különbség az elemek szervezési módjában volt. Tulajdonképpen elkészítettem ugyanannak a témának, három különböző elrendezésű változatát. A felületi elemek alapvető szintjeit rétegek segítségével terveztem meg, így amikor az animációra került sor már minden ott volt mozgásra készen a saját rétegében.

5.1 A projekt szerkezete

A logikai struktúra kialakításakor figyelembe kellett venni, hogy a CD tartalma a világhálón is publikálható legyen.

Két megoldás kínálkozott:

- több .SWF fájlt készítek, amelyeket aztán saját HTML lapjukon helyezek el. (6/a. ábra)
- egyetlen HTML lapot készítek a fő .SWF fájl számára (alap.swf) és ez a fájl tölti be a lejátszóba a további tartalmat. (6/b. ábra)



6. ábra

Az (a) ábrán egy olyan struktúrát látunk, amely több HTML oldalból áll és az oldalak mindegyike tartalmaz egy-egy .SWF fájlt. A (b) ábra olyan struktúrát szemléltet, amely egyetlen HTML oldalt tartalmaz. Ezen HTML oldalon helyezzük el a tartalom további részének betöltéséért felelős .SWF filmet.

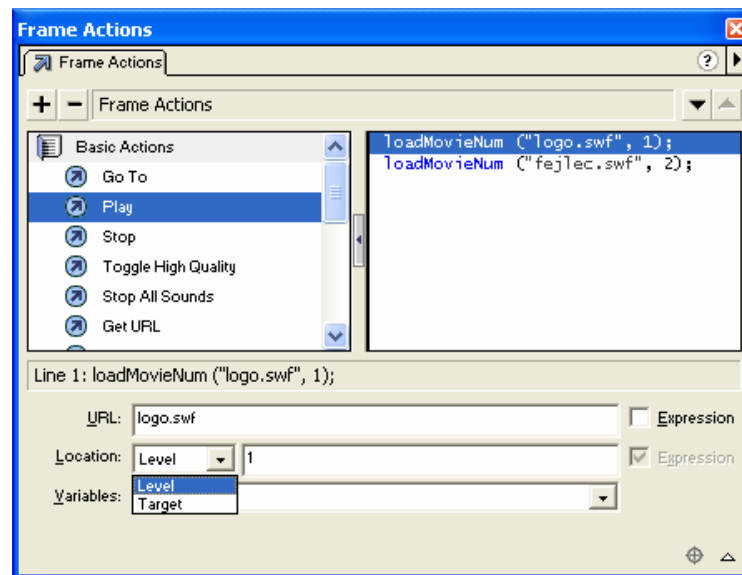
A CD készítésekor a (b) ábrának megfelelő struktúrát alkalmaztam. A teljes megjelenítésért az alap.html fájl felel. Ez a fájl tartalmazza a következő HTML kódot:

```
<OBJECT classid="clsid:D27CDB6E" WIDTH=100% HEIGHT=100%>  
<PARAM NAME=movie VALUE="alap.swf">
```

```
<PARAM NAME=quality VALUE=high>
<PARAM NAME=bgcolor VALUE=#CCCCCC>
<EMBED src="alap.swf" quality=high bgcolor=#CCCCCC
WIDTH=100% HEIGHT=100% TYPE="application/x-shockwave-
flash"></EMBED></OBJECT>
```

Ez a kód betölti a HTML oldalba az alap.swf fájlt, amely a projekt többi részének betöltéséhez szükséges kódokat is tartalmazza.

Hogyan lehetséges ez? A Flash rendelkezik egy programozói felülettel (7. ábra) és egy saját szkriptnyelvvvel az ActionScript-el. A loadMovie akció segítségével további filmeket tölthetünk be a filmünk különböző szintjeire. A szint (Level) tulajdonságot a loadMovi akcióval összefüggésben alkalmazzuk, és meghatározhatjuk vele a filmek takarási sorrendjét, azaz, hogy melyik film melyik másik fölé kerül. A 0. szint az alap szint mely a film jellemzőit definiálja, mint például a háttérszint. Az 1-es szint lesz a következő réteg és így tovább. A loadMovi akciót akkor használjuk, amikor egy aktuális filmből egy másik .SWF fájlt akarunk betölteni a saját ablakába (level parameter) vagy egy moziklipbe (target parameter).



7. ábra

Az Object/Frame Actions panel használat közben

Az alap.swf fájl, mely az alapszinten helyezkedik el (0-ás szint) két másik Flash filmet tölt be, a logo.swf fájlt az 1-es szintre és a fejlec.swf fájlt a 2-es szintre.(8. ábra)



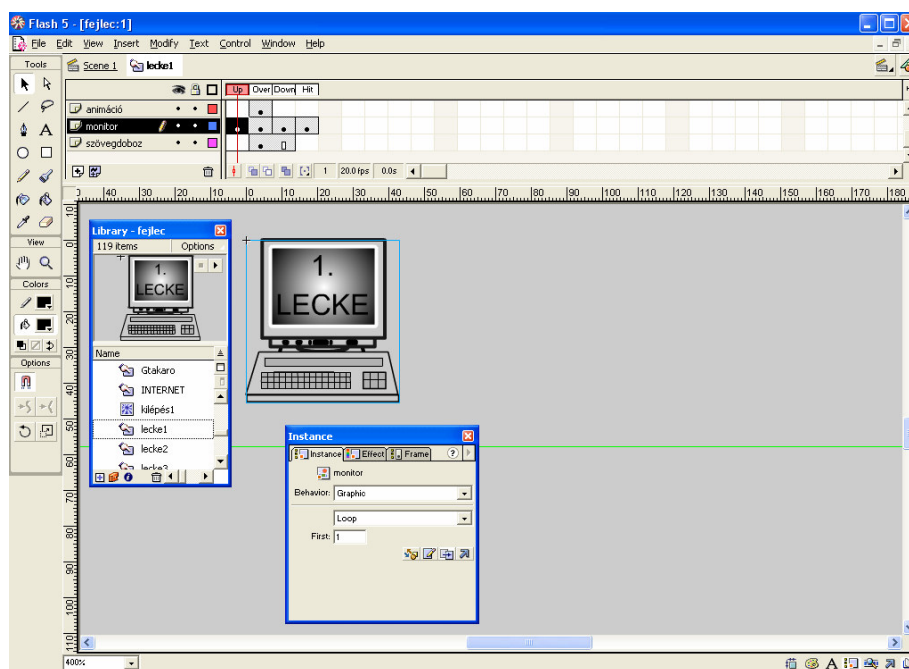
8. ábra
A nyitó képernyő

Akkor mondjuk egy multimédiás oktatóanyagra, hogy interatív, ha a felhasználó (tanuló) befolyásolni, vezérelni képes a film lejátszását, irányát vagy eredményét. Nagyon sok mód létezik, hogy egy felhasználó választásaival képes legyen meghatározni az interatív film kimenetét. Annak a ténynek köszönhetően, hogy sok lehetőség közül lehet választani, minden felhasználó egyedi élményben fog részesülni, mivel sok minden a felhasználó választásán múlik. Ez az interaktivitás lényege és az ActionScript különlegessége.

A 2-es szintre betöltődött fejlec.swf fájl tartalmazza azokat a vezérlő gombokat (főmenü), amelyek kattintásra betöltik a tananyag tanítási egységeit és a programhoz tartozó sűgöt az 1-es szintre. Az 1-es szint tartalmazza a logo.swf fájlt mely a program indításakor látható animációt tartalmazza. Mivel egy szinten egyszerre csak egy fájl tartózkodhat, valamely tanítási egység betöltésekor felülíródik az 1-es szint aktuális tartalma.

5.2 A főmenü szerkezete (fejlec.swf)

A lecke főmenüje egyetlen fájlból tevődik össze, amelynek fő jelenetében (Scene1) 7 réteget helyeztem el. Ezek a rétegek tartalmazzák azokat a szimbólumokat (Flash objektumok), amelyek végső fájl építőelemei. A szimbólum, olyan újrahasznosítható elem, amely az aktuális film könyvtárában(Library) helyezkedik el. Minden alkalommal amikor egy szimbólumot helyezünk el a jeleneten vagy egy másik szimbólum belsejében, valójában a szimbólum egy példányával dolgozunk. A szimbólumokból több példányt is felhasználhatunk, anélkül, hogy ezzel jelentősen növelnénk a végső fájl méretét. Például a tanítási egységek gombsorának elkészítésekor elegendő volt egyetlen grafikus, valamint movi (film) szimbólumot létrehozni és azok példányait újra felhasználni a gombk kialakításában. (9. ábra)



9. ábra

Az 1.lecke gombja szimbólum szerkesztő nézetben

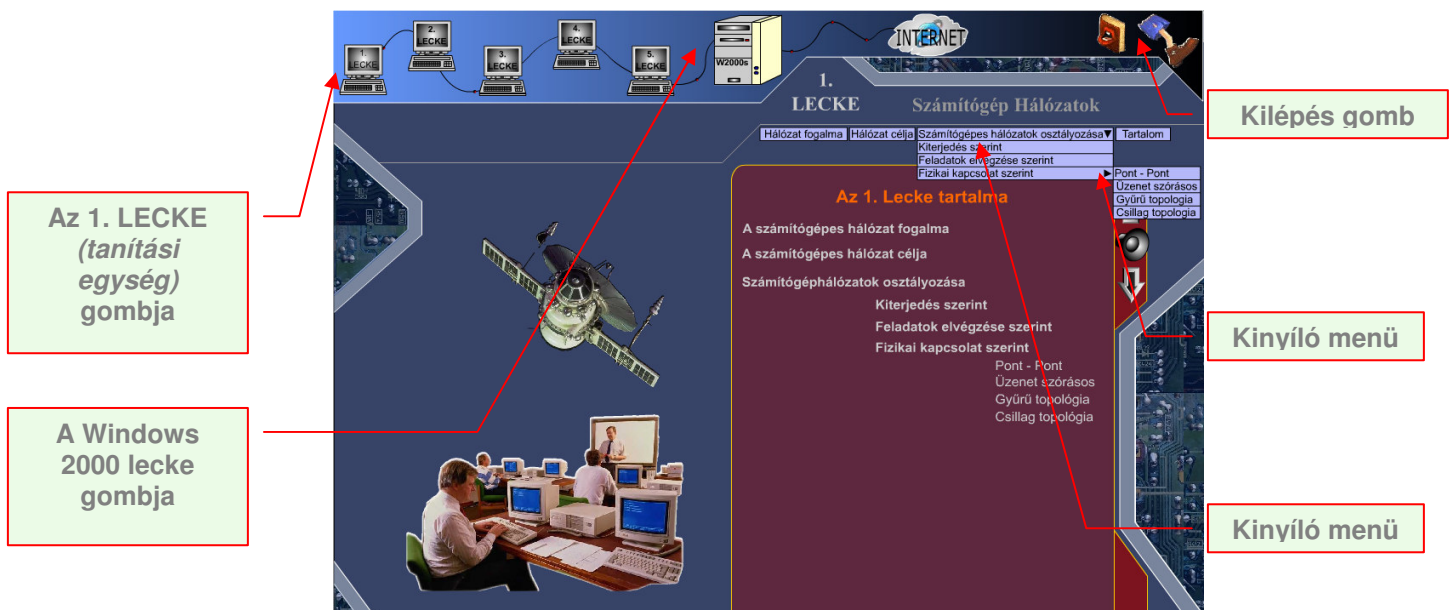
6. FEJEZET

Az oktatóanyag tanítási egységeinek szerkezete

A tanítási egységek szerkezete lényegesen bonyolult kialakítást igényel. A fő időszalagon 7 réteget helyeztem el, ugyanakkor a rétegek tartalma több dimenzió mélységben szerte ágaznak és minden egyes dimenzió saját idő egyenessel rendelkezik, melyek a felhasználó (tanuló) beavatkozásának hatására jelennek meg a képernyőn. A könnyebb érthetőség érdekében a program működésének bemutatásával kezdem, a felhasználás szempontjából. Majd az egyes elemek (szimbólumok) megvalósításának technikai fogásait mutatom be.

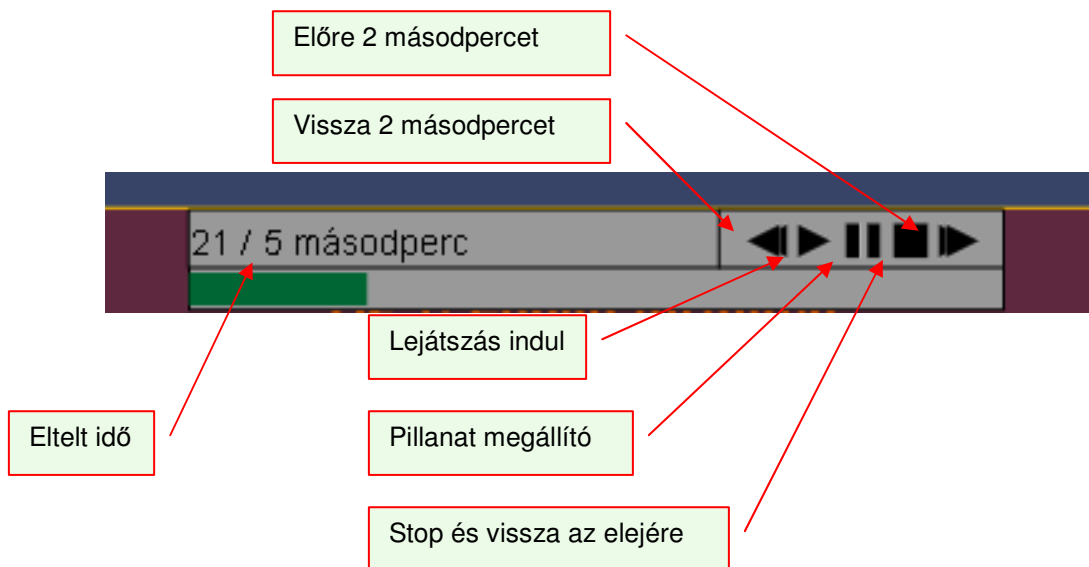
6.1 A program működése

Bármely tanítási egység gombjára kattintva, egységes képernyőobjektumok jelennek meg. Minden esetben a lecke tartalmát megjelenítő panellel, és a lecke felépítését reprezentáló és elérhetőségét biztosító menürendszerrel találkozunk az első képernyőn. (10. ábra)



10. ábra

Ha kiválasztunk egy fejezetet a menürendszerből az addigi tartalom eltűnik a képernyőről és az általunk választott témával kapcsolatos információ töltődik be. Az anyag rész szöveges részét képező szövegblokk a monitor jobb oldalán, míg az interaktív szemléltető animációkat és képeket bemutató rész a képernyő bal oldalán helyezkedik el. A szöveges rész tartalmaz két szöveggörgetésre alkalmas nyíl alakú gombot, valamint egy hangszóró ikont amelyre rákattintva a lecke hanganyagát tartalmazó .swf fájl töltődik be a 3-as szintre. Ez a hanganyagot tartalmazó film elvan látva a hanganyag lejátszását befolyásoló vezérlő gombokkal. (11. ábra)



11. ábra

A hanganyag lejátszását vezérlő gombok

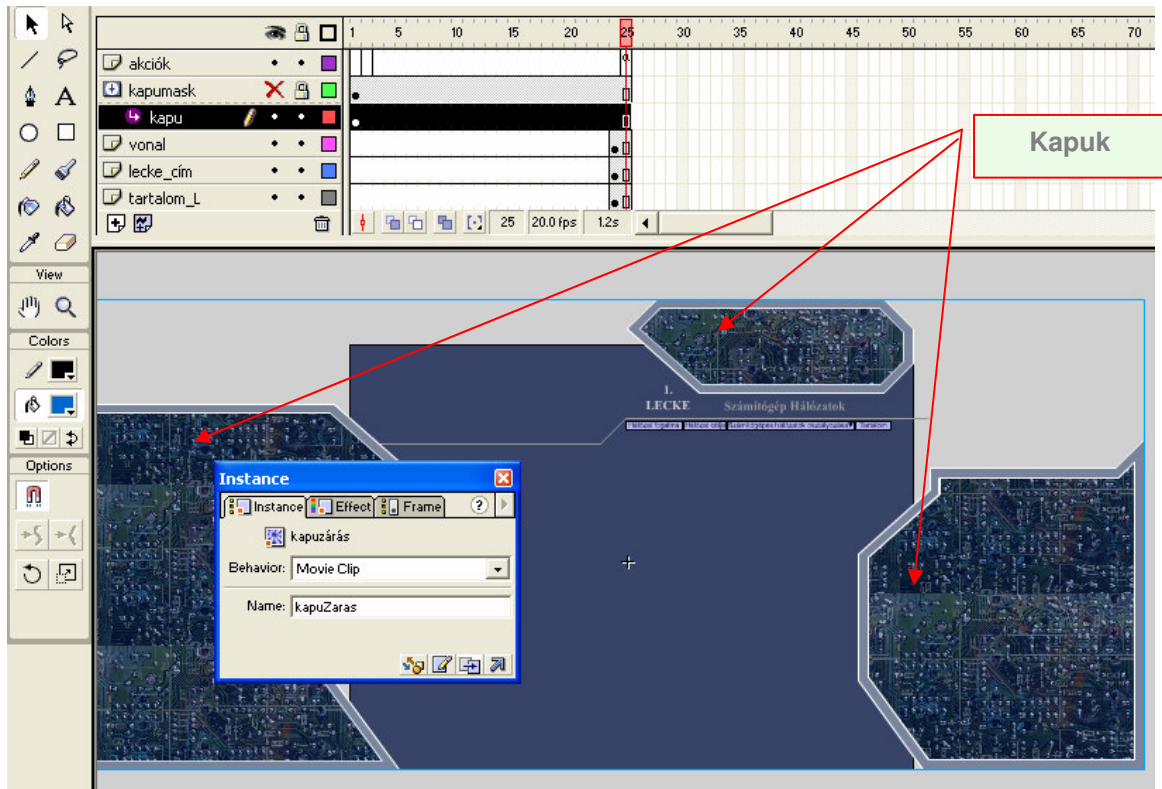
A prezentációs rész is tartalmaz vezérlő gombokat, melyek segítségével elindíthatjuk, megállíthatjuk, újra indíthatjuk a szemléltető animációs klipet, illetve az indító gomb segítségével az egyes bemutatószakaszokból tovább léphetünk. Minden leckéhez tartozik egy szintfelmérő rész, amely felméri a tanuló tudásszintjét és javaslatot tesz a tananyag feldolgozásának további menetére.

6.2 A projekt objektumainak készítéséhez alkalmazott megoldások

A leckék betöltésekor a legszembeütőbb látványosság a képernyő kapuinak mozgása. A kapuk a fő időegyeses első képkockájától megkezdik a betöltődésüket. A 25 képkockából (frame) álló fő idő egyes 20 frame / sec lejátszási sebesség mellett 1,25 másodperc alatt lejátszódik és a vezérlés újra az első képkockára kerül. Ez nem lenne egy szerencsés megoldás, hiszen a film többi része a 23. képkockától kezdődően töltődik be és így egy végtelenített filmet kapnánk amelyben folyamatosan fel-fel villan a 23. frame tartalma. A hiba kiküszöbölése érdekében elhelyeztem egy üres réteget, amelynek az „akciók” nevet adtam. Az „akciók” réteg 25. képkockájában létrehoztam egy üres kulcskockát, melynek a következő ActionScript utasítást adtam: `stop ()`;

Ezt az utasítást célszerű elhelyezni minden film időegyesésének és minden Movie Clip szimbólum időegyesésének utolsó kockájába, kivéve ha az a feltett szándékunk, hogy az adott film végtelenítve legyen.

A „kapuk” fentről a 3. rétegen vannak elhelyezve. Valójában azonban nem a fő időegyesesen zajlik a mozgásuk, hanem a „kapuzárás” nevű film szimbólum egy példánya van elhelyezve a fő időszalagon. (12. ábra) A film szimbólum saját idő egyenessel rendelkezik, melynek végén szintén a `stop ()`; utasítás van elhelyezve.



12.ábra

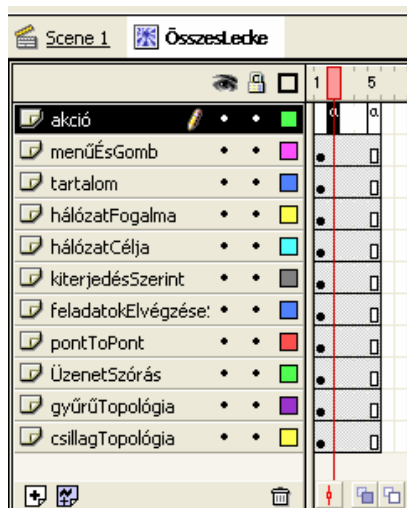
6.3 A lecke témaköreit tartalmazó filmszimbólum szerkezete

A fő időszalag `tartalom_L` nevű rétegre töltődik be az „ÖsszesLecke” film szimbólum `csop` nevű példánya. Ebben a szimbólumba ágyazódik be a `tartalom` többi része: a menürendszer és az egyes témaköröket tartalmazó szimbólumok egy-egy példánya. (13. ábra) A `csop` példány akció rétegének 2. kulcskockájába a következő kód található:

```
_root.csop.v1 = tartalom;
_root.csop.v1.gotoAndPlay(2);
```

A kódból kitűnik, hogy az ActionScript nyelv a JavaScript nyelvhez hasonlóan az objektumorientált programozás szintaksziséát alkalmazza. A `_root` egy foglalt név, mellyel a fő idő egyenesre hivatkozhatunk. Az előző kód a fő idő egyenes `csop` szimbólumpéldányának `v1` változójának a `tartalom` értéket adja. Egy változónak kétféle képen adhatunk értéket, az egyik lehetőség, ha idézőjelek közzé írjuk az értéket. Ekkor szöveg típusú

lesz a változó. Abban az esetben, ha nem használunk idéző jeleket, akkor objektumra hivatkozhatunk a változó segítségével. Így az előző értékadással a változót felhasználva egy filmszimbólum tartalom nevű példányára hivatkozhatunk. Ez a példány az oktatási projektben a lecke tartalmát és a hozzá tartozó grafikus illusztrációt tartalmazza. Az egyes témákat tartalmazó filmszimbólumok minden rétegének első képkockája üres. Erre azért van szükség, mert a színpadra betöltődött példányok egyszerre vannak jelen és a takarási sorrendjük a rétegekben elhelyezett sorrendnek megfelelően rendeződik. Ez esetben csak a legfelső rétegben elhelyezett szimbólumpéldány lenne látható minden esetben. A kód második sorában a `gotoAndPlay(2);` metódus hívásával a tartalom példány első (üres) képkockájáról a másodikra irányítjuk a vezérlést és elindítjuk a lejátszást. Ennek hatására betöltődik a képernyőre a lecke tartalmát megjelenítő filmklip.



13.ábra

Az 1. lecke összes témakörét tartalmazó film szimbólum idő egyenese és rétegei.

Amikor a felhasználó a lecke valamelyik témakörét kiválassza a menüből, rákattint egy témakörre utaló menügombra. Ekkor több egéresemény is

történik. A gombokhoz kapcsolódó eseménykezelők az egérmutató helyét és mozgását figyelik. Ha a mutató kapcsolatba kerül a gomb találati területével, a nyilat formázó mutató kéz szimbólummá alakul át. Ilyenkor azt mondjuk, hogy a mutató a gomb felett van. Ha a mutató nincs a gomb felett, akkor a gombon kívül van. Amikor az egeret az egérgomb lenyomása nélkül mozgatjuk, egyszerűen mozgatásról beszélünk. Amikor az egeret az egérgomb lenyomása mellett mozgatjuk, akkor húzzuk a mutatót. A gombokhoz kapcsolódó eseménykezelőket és utasításokat nem az eredeti gombszimbólum időszalagján, hanem az egyes gombpéldányokon kell elhelyezni. A Flash nem teszi lehetővé, hogy eseménykezelőket helyezzünk a gomb szimbólum időszalagjára.

6.3.1 A Flash gombok eseménykezelői

Press (Lenyomás)

Egy egyszerű kattintás két jól elkülöníthető részre bontható: a gomb lenyomására és felengedésére. A Press esemény akkor következik be, amikor az egérmutató a gomb találati területe felett van és a felhasználó lenyomja az egér gombját.

Release (Felengedés)

A Release esemény akkor következik be, amikor a mutató a gomb találati területe felett van és a felhasználó lenyomja, majd felengedi az egér gombját. Normális esetben ezt az eseménykezelőt a kattintások észleléséhez célszerű használni.

Release Outside (Felengedés kívül)

A Release Outside esemény akkor következik be, amikor a felhasználó az egérmutatót a gomb találati területe fölé mozgatja, lenyomja az egérgombot, az egérmutatót lehúzza a gombról, majd felengedi az egérgombot. A Release Outside eseménykezelővel a meghiúsult kattintásokra reagálhatunk.

Roll Over (Keresztülmozgatás)

A Roll Over esemény akkor következik be, amikor az egérmutatót a gomb találati területe fölé húzzuk az egérgomb lenyomása nélkül.

Roll Out (Kimozgatás)

A Roll Out esemény akkor következik be, amikor az egérmutató elhagyja az egér találati területét és az egérgomb nincs lenyomva.

Drag Over (Keresztülhúzás)

A Drag Over esemény kiváltásához a következőket kell tennünk: nyomjuk le az egér gombját a gomb találati területe felett, húzzuk le a mutatót a gombról (közben ne engedjük fel a gombot), majd újra húzzuk a mutatót a gomb fölé (a gomb továbbra is le van nyomva).

Drag Out (Ki húzás)

A Drag Out eseményt a következő módon válthatjuk ki: nyomjuk le az egér gombját, amikor az egérmutató a gomb találati területe felett van, majd húzzuk le a mutatót a gombról.

A lecke témáit aktiváló menügombokhoz a Release (Felengedés) eseménykezelőt használtam. Például a „Hálózat célja” gombhoz a következő kódot rendeltem:

```
on (release) {  
    if (_root.csop.v1 != celja) {  
        unloadMovieNum (3);  
        _root.csop.v1.gotoAndPlay("vége");  
    }  
}
```

A kód akkor hajtódik végre, ha a gombra rákattintunk(on (release)), aztán egy feltétel vizsgálat következik. Az if ágban akkor hajtódnak végre az utasítások, ha a v1 változó értéke nem egyenlő a betöltendő film

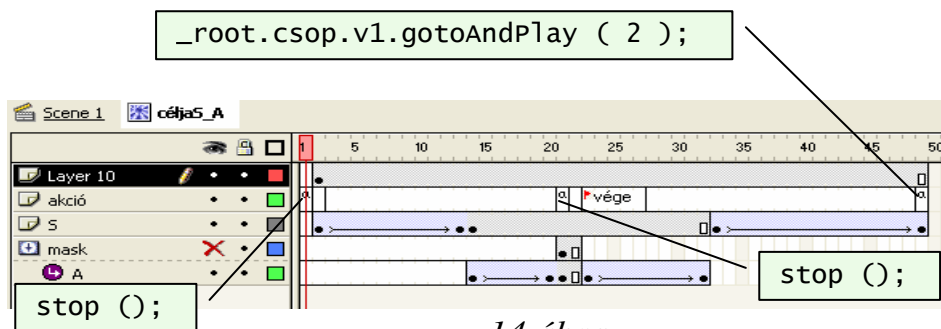
példányának nevével (celja). Ez akkor következik be, ha a celja nevű példány már bevan töltve a képernyő látható területére. Az unloadMovieNum (3); utasítás a lejátszó harmadik szintjéről kitörli az oda töltött .SWF fájlt. Erre azért van szükség, mert a témakörök hanganyagát tartalmazó fájl erre a szintre töltődik. Ha a hang fájl aktiválva van és nem ürítenénk ki ezt a szintet, akkor az akkor is halható lenne, ha már egy másik témakör anyagát töltöttük volna be a színpad látható területére. Ezért ezt az utasítást minden egyes témakör illetve lecke választását lehetővé tevő gomb kódjába el kell helyezni.

A `_root.csop.v1.gotoAndPlay("vége");` sor az aktív témakör film példányának vezérlését arra utasítja, hogy a vezérlés ugorjon a "vége" címke-re és indítsa el a lejátszást. (14.ábra)

Minden témakör utolsó képkockája tartalmazza a következő kódot:

```
_root.csop.v1.gotoAndPlay ( 2 );
```

Mivel a "vége" címke-re ugrást vezérlő gombhoz rendelt kód utolsó utasítása `_root.csop.v1 = celja;` a v1 változó értékét a kiválasztott témakört tartalmazó szimbólumpéldány nevére (celja) változtatta. Ennek segítségével az előző, aktív példány végén elhelyezett gotoAndPlay utasítás betölti a színpad látható részére a kívánt témakört. Példánkban a „Hálózat célja” témakör töltődik be.



14.ábra

A témakörök egyik filmklip példányának idő egyenese.

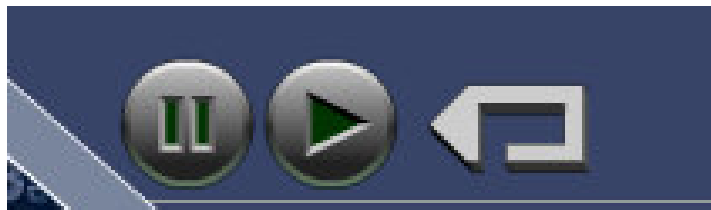
6.4 A leckék témaköreinek szerkezete és működése

A lecke témaköreit tartalmazó példányok is további két szimbólumpéldánnyal bővülnek, amelyek szintén saját idő egyenessel rendelkeznek és azokon is további szimbólumok helyezkednek el: szövegblokk, görgető gombok, hanglejátszást indító, leállító gomb, ellenőrző kérdések gomb, valamint az animációt tartalmazó szimbólumpéldányok.

6.4.1 Az animációs klipet vezérlő gombok

A témákat prezentáló animációk elkészítésében nem lehetett egységes szerkezetet kialakítani. Ugyanis az animációs klipek megvalósítása témájuknál fogva más és más technikai megoldásokat igényeltek: további klipek és grafikus szimbólumok beágyazását, különböző rajz eszközök használatát, raszteres képek használatát, stb.

Néhány hasonlóság azért mégis adódik. Minden témakörhöz tartozó animációs klip szimbólumpéldánya az „anim” példánynevet kapta. Ennek az egységes elnevezésnek a kódok írásában van jelentős szerepe. A prezentációkat vezérlő gombok kialakítása is egységes, ezek a téma filmklip példányán elhelyezett rétegben kaptak helyet. Három gomb tartozik ide (15.ábra): pillanat megállítási, lejátszás, újrajátszás az elejétől



15.ábra

A témakörök prezentációs részét vezérlő gombok

A gombok kódjai a hálózatok fogalma című témakörben a következők:

pillanat megállítás

```
on (release) {  
    _root.csop.fogalom.anim.stop ();  
}
```

lejátszás

```
on (release) {  
    _root.csop.fogalom.anim.play ();  
}
```

újrajátszás az elejétől

```
on (release) {  
    _root.csop.fogalom.anim.gotoAndPlay(2);  
}
```

Az animációs klipek első képkockájához egy stop (); utasítás van rendelve. Ahhoz, hogy a klip újrainduljon, a gotoAndPlay utasítás argumentumában a klip második képkockájának értékét kell megadni. Mivel az animációs szimbólumpéldányoknak egységesen az „anim” nevet adtam, ezért a különböző témák prezentációját vezérlő kódokban csak a témakör példánynevét kellett megváltoztatni. Például a „Hálózat célja” című témakörhöz tartozó újrajátszást vezérlő gombja a következő kódot tartalmazza:

```
on (release) {  
    _root.csop.celja.anim.gotoAndPlay(2);  
}
```

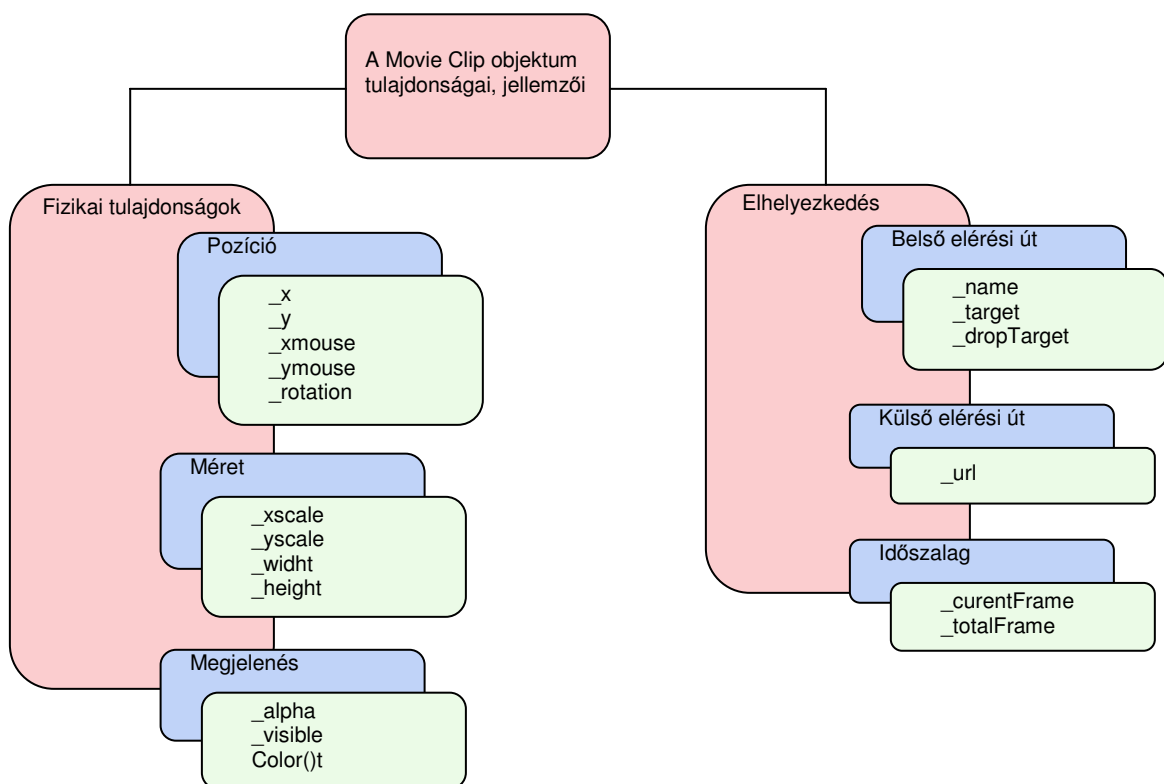
6.5 A szöveges rész működése

A szöveges részt és a hangot tartalmazó szimbólumpéldányokat is egységesen „tAnyag” néven neveztem el. Így minden esetben a következőképpen hivatkozhattam rájuk:

```
_root.csop.<a téma példány neve>.tAnyag.<metódus>;
```

Ennek a filmklipnek a szerkezete roppant egyszerű. Maga a klip egyetlen frame-ből áll és hat rétegből. Az egyes rétegek tartalmazzák magát a szövegdobozt, a szöveget, a szöveget görgető gombokat, a hangbetöltő gombot és az ellenőrző kérdések gombot.

Magát a szöveget egy film példányban helyeztem el és az animációs példányhoz hasonlóan egységes nevet kapott (tBox). A szöveg filmklipben való elhelyezésére azért volt szükség, mert az ActionScript utasítások és metódusok segítségével csak a filmklip példányok érhetőek el és csak is a filmklip példányok tulajdonságai kérdezhetőek le és módosíthatók. A szöveggörgetés problémájának a kulcsa a szöveges tBox nevű példány koordinátáinak módosításában rejlik. Minden filmklip példány rendelkezik olyan tulajdonsággal, amelyek az adott példány kinézetét, méretét, és helyzetét szabályozzák. Ha például megszeretnénk változtatni egy filmklip példány helyzetét a színpadon, elég ha egyszerűen megváltoztatjuk az X és Y koordináták értékét. Ezekre a tulajdonságokra az ActionScript nyelven a `_x` illetve `_y` szimbólumokkal hivatkozhatunk. (16. ábra)



16. ábra

A Movie Clip objektum tulajdonságai

A Flash egyik szolgáltatása az `onClipEvent` eseménykezelő. Ezzel az eseménykezelővel lehetőségünk nyílik arra, hogy kívülről kapcsoljunk utasításokat a filmkliphez. Ezek az utasítások nem jelennek meg a filmklip időszalagján.

Az `onClipEvent enterFrame` eseménykezelőjét használtam a szöveg görgetéséhez. Ez az eseménykezelő a filmklip-példány minden kockájának lejátszásakor végrehajtódik. Az eseménykezelőhöz tartozó utasítások a képkocka többi utasítása után kerülnek végrehajtásra.

A gombokhoz tartozó eseménykezelők közül a `press` (lenyomva) és a `release` (felengedés) eseménykezelőket használtam.

A görgetésért felelős gombokhoz a következő kódok tartoznak:

```
on(press){
    _root.csop.celja.tAnyag.upHold = true;
}
on(release){
    _root.csop.celja.tAnyag.upHold = false;
}
```

illetve

```
on (press) {
    _root.csop.celja.tAnyag.downHold = true;
}
on (release) {
    _root.csop.celja.tAnyag.downHold = false;
}
```

Létrehoztam egy filmszimbólumot és egy példányát elhelyeztem a `tAnyag` példányok színpadon kívül eső részén. A színpadon nem látható példányhoz a következő kódot rendeltem:

```
onClipEvent(enterFrame){
    if(_root.csop.celja.tAnyag.upHold &&
    _root.csop.celja.tAnyag.tBox._y <= 0){
        _root.csop.celja.tAnyag.tBox._y =
        _root.csop.celja.tAnyag.tBox._y + 10;
    }
}
```

```

        if(_root.csop.celja.tAnyag.downHold &&
        _root.csop.celja.tAnyag.tBox._y >= 0-
        _root.csop.celja.tAnyag.tBox._height+350){
            _root.csop.celja.tAnyag.tBox._y =
            _root.csop.celja.tAnyag.tBox._y - 10;
        }
    }
}

```

Az `enterFrame` eseménykezelő figyeli a gombok állapotát. Ha a gombok lenyomott állapotban vannak, az `upHold`, illetve a `downHold` változók a `press` esemény hatására felveszik a `true` logikai értéket. A szöveget tartalmazó film pozíciója a kódokban meghatározott feltételeknek megfelelően 10 pixeles ugrásokkal le illetve felfelé gördül a képernyőn.

Amikor a felhasználó felengedi a gombot a `release` esemény következik be. Az `upHold` vagy a `downHold` változó felveszik a `false` logikai értéket. Az `enterFrame` eseménykezelő `if` ágában megadott feltétel kiértékelése hamis eredményt ad, így a szöveges rész szkrollozása megszakad. A szöveges részek nem minden esetben egyforma terjedelműek, ezért a második `if` ágban az aktuális szövegdoboz méretével kisebb pozíciót kell megadnunk (`0-tBox._height`). Ez esetben a szöveg teljes tartalma kigördül a képernyő látható tartományából, ezért a szöveget megjelenítő képernyőrésznek a kétharmadát (350 pixel) hozzá kell adni a feltételben meghatározott értékhez.

6.6 A szintfelmérő panel működési elve

Miután a „`tAnyag`” szimbólumpéldány valamelyikén a szintfelmérő ikonra kattint a felhasználó (17.ábra), a szintfelmérő panel(`felmer.swf`) betöltődik az `alap.swf` fájl harmadik rétegére. Az imént említett gombhoz az alábbi kód tartozik:

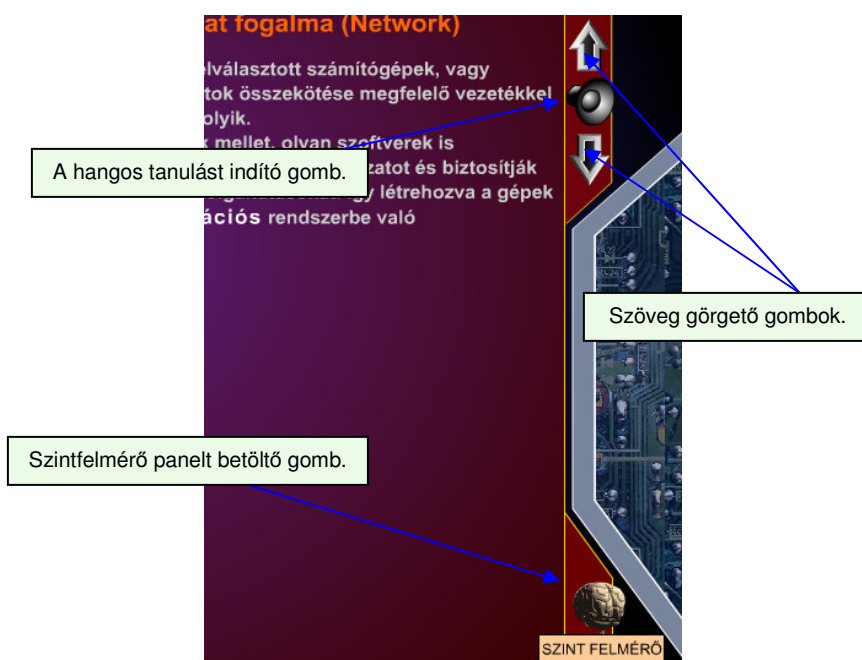
```

on (release) {
    loadMovieNum ("lec1/felmer.swf", 3);
    _root.csop.v1.gotoAndPlay(1);
}

```

A kód harmadik sora az éppen aktuális `tAnyag` szimbólum példány lejátszófejét az első kulcskockára irányítja, ami azt eredményezi, hogy az

aktuális szimbólumpéldány eltűnik a monitorról. Erre azért van szükség, mert a Flash gombok akkor is aktívak, ha azokat egy magasabb rétegen lévő objektum takarja. Ha a tAnyag szimbólum gombjai a szinpadon maradnak, hiába fedí azokat le a harmadik szintre betöltött swf fájl, és ha a felhasználó az aktív gombok területére kattint nem várt események sorozata következhet be. Például a hangos tanulást elősegítő gomb is a harmadik szintre tölti be a leckéhez tartozó aktuális hangkezelő fájlt. Mivel a szintekre való betöltéskor az aktuális szint tartalma felül íródik, a szintfelmérő panel kitörlődne a színpadról.



17.ábra

A „tAnyag” szimbólum példányok gombjai.

Amikor a szintfelmérő panelt a felhasználó bezárja, gondoskodni kell arról, hogy az aktuális tAnyag szimbólum vissza töltődjön a képernyőre. Erre az egységes képernyőkép megőrzése miatt van szükség és a problémát következő Actions Script kód oldja meg:

```
on (release) {
    _level1.csop.v1.gotoAndStop ( 21 );
    unloadMovieNum (3);
}
```

Ez a kód utasítja a fő film _level1 rétegében lévő lejátszófejet, hogy álljon az aktuális tAnyag példány 21. kockájára. A kód harmadik sora kitörli a 3. szintről a felmer.swf fájlt.

A panel szerkezete az időegyenesen szekvenciálisan egymás után elhelyezett kérdéscsoportokból, és az azokat kiértékelő ActionScript kódokból áll össze. (18. ábra)

Az egyes kérdésekhez eldöntendő (igaz-hamis), illetve négy tagú (a, b, c, d) radio gombok tartoznak.

Ha a felhasználó jól válaszol a kérdésekre, kérdésenként 2 pont adható, ha rosszul válaszol 1 pont levonás jár, a válasz kihagyása esetén 0 pont. Ezen szabályok értelmében a radio gombokhoz rendelt ActionScript kódok a következők:

a) Igaz-Hamis gombok esetében, ha a válasz jó

```
on (release) {  
    jovalasz=true;  
    gotoAndStop ("hamisgomb");  
    if (rosszvalasz){  
        _root.osszpont=_root.osszpont+3;  
        rosszvalasz=false;  
    }  
    else{  
        _root.osszpont=_root.osszpont+2;  
    }  
}
```

b) Igaz-Hamis gombok esetében, ha a válasz rossz

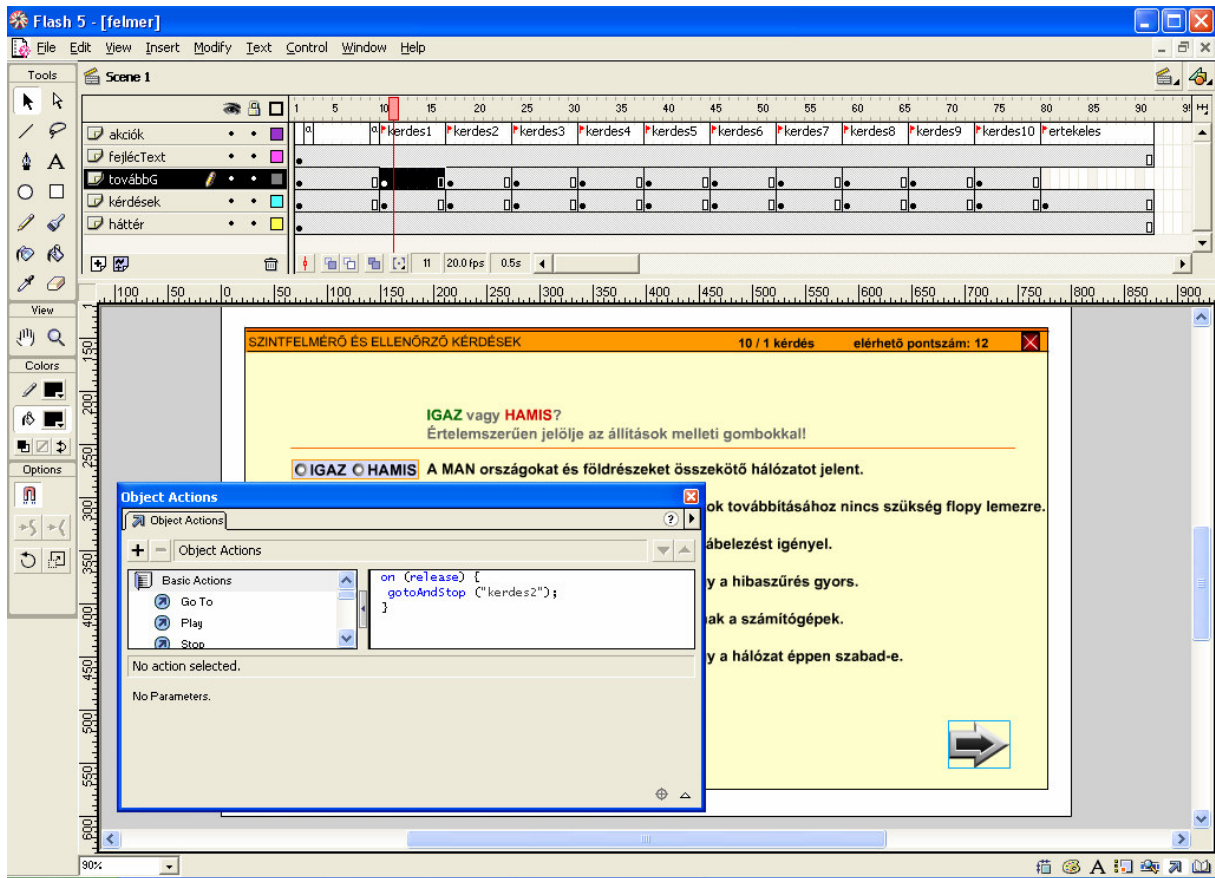
```
on (release) {
    rosszvalasz=true;
    gotoAndStop ("igazgomb");
    if (jovalasz){
        _root.osszpont=_root.osszpont-3;
        jovalasz=false;
    }
    else{
        _root.osszpont=_root.osszpont-1;
    }
}
```

c) Eldöntendő gombsor esetén ha a válasz jó

```
on (release) {
    jovalasz=true;
    gotoAndStop ("bgomb");
    if (rosszvalasz==true){
        _root.osszpont=_root.osszpont+3;
        rosszvalasz=false;
    }
    else{
        _root.osszpont=_root.osszpont+2;
    }
}
```

d) Eldöntendő gombsor esetén ha a válasz rossz

```
on (release) {
    gotoAndStop ("agomb");
    if (!rosszvalasz){
        rosszvalasz=true;
        if (jovalasz==true){
            _root.osszpont=_root.osszpont-3;
            jovalasz=false;
        }
        else{
            _root.osszpont=_root.osszpont-1;
        }
    }
}
```



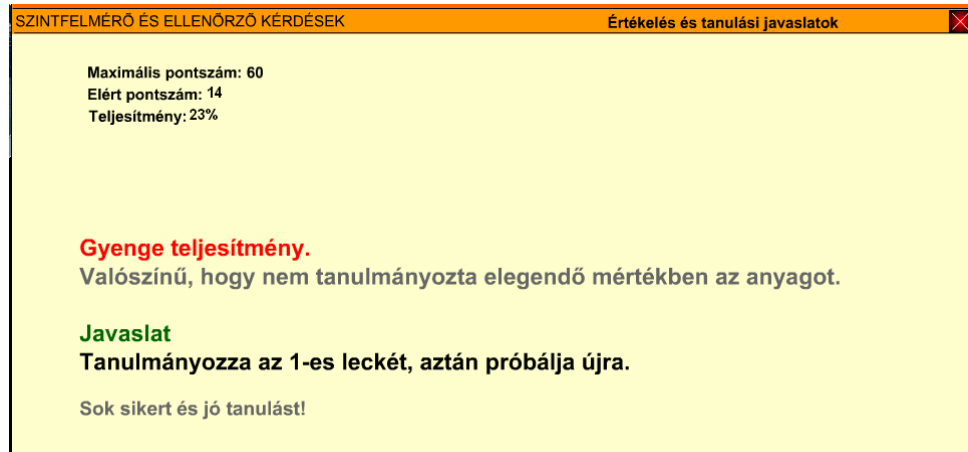
18. ábra

A felmer.fla /szintfelmérő panel/ szerkesztés közben. Jól látható az az ActionScript kód ami a „tovább” gombra kattintva a lejátszófejet a második kérdéscsoport címkéjére navigálja.

A felmérőben elért pontszámokat az osszpont változóban tároljuk. A kiértékelő szakasz színpadán elhelyeztem két dinamikus szövegdobozt az egyiknek az értékének osszpont változót adtam, a másikhoz a szazalek változót rendeltem hozzá. Az összegző panel filmklip példányának első kulcskockájában elhelyeztem a következő kódot:

```
szazalek=Math.round(_root.osszpont/_root.maxpont*100);
teljesit=szazalek+"%";
if (szazalek<=30){ gotoAndStop ("30alatt")}
else if (szazalek<=50) {gotoAndStop ("50alatt");}
else if (szazalek<=70) {gotoAndStop ("70alatt");}
else if (szazalek<=95) {gotoAndStop ("95alatt");}
else if (szazalek<=100) {gotoAndStop ("100alatt");}
```

A kód első két sora a dinamikus szövegdozoknak az elért eredmény szerinti értéket adja. Az if ágakban lévő `gotoAnd Stop("címkék">)` utasítás az elért eredmény szerinti besorolásnak megfelelő időszekvenciára ugrik a kiértékelést tartalmazó filmklip időegyenesén és a lejátszófejet megállítja. Például 30% alatti eredmény elérése esetén a a lecke többszöri átismétlésére kéri a felhasználót a program. (19.ábra)



19.ábra

A szintfelmérő panel kiértékelő táblája a lecke ismétlésére tesz javaslatot.

Annak érdekében, hogy a kérdőív kiértékelése abban az esetben is korrekt legyen, ha a felhasználó nem csak egyszer avatkozik be a gombok kiválasztásakor (például össze-vissza klikkel) szükséges két logikai változót használnunk (`rosszvalasz`, `jovalasz`). Ha a válasz rossz, akkor megvizsgáljuk egy feltételes ágban, hogy választott e korábban a felhasználó egy másik, olyan lehetőséget, ahol a válasz szintén rossz. Ha igen, akkor csak a `gotoAndStop`. ("címké") utasítás fog végrehajtódni. Ellenkező esetben megvizsgáljuk, hogy az utolsó választás egy jó válasz volt e. Ha igen, az összpontszámból 3 pontot kell levonni, hiszen már korábban hozzáadódott 2 pont és a szabályok értelmében 1 pont levonás jár a rossz válaszokért. Ellenkező esetben, ha a felhasználónak ez az első választása és a válasz rossz, akkor 1 pont levonás jár, hiszen az adott feladat panelen még tiszta lappal indul, vagyis a feladatlap értéke 0 pont.

7. FEJEZET

A multimédiás CD elkészítése

7.1 A CD automatikus indítása

Bizonyára mindenki találkozott olyan CD-vel, amely a behelyezése után automatikusan elindított valamilyen programot. Ilyenek például a játékot tartalmazó CD-k, vagy az összes Microsoft szoftvert tartalmazó CD. Windows alatt a titok nyitja első megközelítésben mindössze egy fájl, ami "megmondja" a Windows-nak, hogy melyik fájlt kell indítania. A módszer roppant egyszerű, létre hoztam egy autorun.inf nevű fájlt, melynek tartalm a következő:

```
[autorun]
OPEN=program.exe
```

Ennek hatására a CD-t behelyezve a meghajtóba elindul a CD főkönyvtárában található program.exe. Milyen megoldások alkalmazhatók, ha nem egy futtatható fájlt akarok elindítani, hanem pl. egy HTML oldalt akarok megjeleníteni? Gondolhatunk egy ilyen verzióra:

```
[autorun]
OPEN=iexplore.exe oldal.html
```

Ezzel az a gond, hogy a Windows ilyenkor is a CD főkönyvtárában keres, és mivel nem találja, ezért minden indításnál ki kellene tallózni a C:\Program Files\Internet Explorer\iexplore.exe fájlt. Ez nem egy szép megoldás. Mondhatnánk, hogy akkor írjuk így:

```
[autorun]
OPEN="C:\Program Files\Internet Explorer\iexplore.exe"
oldal.html
```

Ez elvileg működik is, de honnan tudjuk, hogy a C: meghajtón van az Internet Explorer, vagy honnan tudjuk, hogy egyáltalán van-e fent IE a gépen? Jó megoldásnak ígérkezik a start.exe nevű program is, de ugyanaz vele a probléma, hogy különböző Windows alatt, különböző gépeken teljesen más az elérési útvonala.

A megoldás az, hogy egy - a start.exe-hez hasonló - univerzális futtató programot kell elhelyezni a CD-n, ami a nem futtatható fájlokhoz megkeresi

a hozzá tartozó alkalmazást, és azt indítja el a dokumentummal. Erre írhatnánk egy miniprogramot. Visual Basic alatt is lehet ilyet írni, egyetlen hátránya az, hogy a program elindulásához szükséges Runtime moduloknak már fent kell lennie a célgépen. Ez viszont biztos, hogy nincs így minden gépen. Használhatunk olyan programnyelveket is, amelyekből önállóan futtatható (nincs szükség futtató modulra) EXE-ket lehet készíteni. Szerencsére ezt már mások megírták helyettünk. Így nekünk csak használni kell. A CD főkönyvtárában összesen 4 db EXE található és egy autorun.inf. A D:\oktato\start.html fájlt kell megjelenítenünk a CD behelyezése után (ahol D: a CD betűjele), ekkor az autorun.inf így néz ki.

autorun.inf (minta)

```
[autorun]
OPEN=bin\autorun\i386\ShellExec \oktato\start.html
ICON=CD.ico
```

```
[autorun.x86]
OPEN=bin\autorun\i386\ShellExec \ prg\start.html
ICON=CD.ico
```

```
[autorun.alpha]
OPEN=bin\autorun\Alpha\ShellEx32 \ prg\start.html
ICON=CD.ico
```

Az ICON= bejegyzés arra ad lehetőséget, hogy a CD-nek saját ikonja legyen, ami utal a CD tartalmára. Például az Intézőben a szokásos CD-meghajtó ikon helyett a mi saját ikonunk jelenik meg.

7.2 Interneten való publikálás

A Flash-ben készült multimédiás anyagok publikálására a CD mellett kiválóan alkalmas a világháló. Ezért érdemes a CD-n elhelyezett anyagot is úgy kialakítani, hogy az bármikor a nagyközönség elé tárulhasson. Például egy iskolai Intranetes környezetben a tanulók számára hozzáférhetővé tudjuk tenni. Az általam készített CD tartalma egy HTML-be ágyazott Flash projekt. Amennyiben hálózatos elérést szeretnénk biztosítani, egyszerűen fel kell másolni a CD „oktato” könyvtárát és annak tartalmát a kiválasztott Intranet vagy Internet tárhelyre. Ezt követően a „oktato” könyvtárban található start.html fájlra mutató hiperhivatkozást kell készítenünk. Ezt a hivatkozást weblapunkon vagy Intranetes oldalunkon elhelyezve kényelmesen elérhetjük a CD teljes anyagát.

8. FEJEZET

A tanuló értékelés

Az oktatásban felhasznált interaktív multimédia-programoknak fokozottan meg kell felelniük bizonyos kritériumoknak, ilyen például a szakmai pontosság és hitelesség, adekvát médium alkalmazása, amelyeknek természetesen összhangban kell lenniük a formai megjelenéssel is. Tudni kell felismerni és megkülönböztetni a szórakoztató, ismeretterjesztő és a szakmailag igényes, tudományosan megalapozott produktumokat.

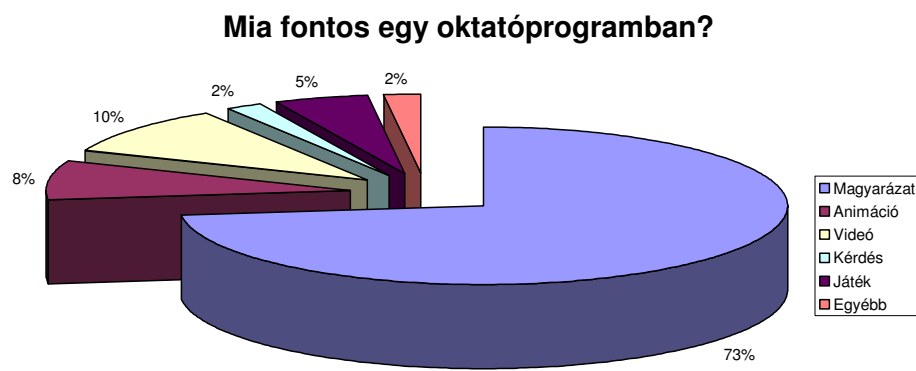
Munkám során igyekeztem ezen kritériumoknak megfelelni és figyelemmel kísértem azon szempontokat, melyek egy oktatóprogram létrejöttéhez szükségesek. Miután létrejött maga a produktum, úgy gondoltam, hogy a legjobb kritikát maguk a felhasználók adnák, ezért a megadott irányelvek alapján egy tesztet állítottam elő. A tesztet már meglévőekből ollóztam össze, melyben a „Multimédia felhasználói programok értékelési szempontjai” című tanulmány volt segítségemre.[9]

A tesztet 15-16 éves diákok között osztottam szét, miután bemutattam az általam készített H.O.P oktatóprogramot. Vizsgálatomat a nyíregyházi Szent Imre Katolikus Gimnáziumban végeztem, 100 tanulóval. A gimnáziumban tanulóknak lehetősége van több ízben számítógép elé ülni, elsősorban a tanórákon, valamint a könyvtárban napi 8 órán keresztül és a kollégiumban a nap nagyobb részében. Az első benyomásokat figyelembe véve és a teszt kiértékelése után levonhattam a legfőbb konzekvenciákat, melyek a következők voltak.

Az első öt kérdésben kíváncsi voltam arra, hogy milyen gyakran használnak számítógépet a tanulók, és milyen szívesen használnak oktatóprogramot. Az öt kérdést illetően több válasz is megadható volt, melyből kiderült, hogy a diákok 92%- a mindennap számítógép elé ül és 8 % -uk hetente használ

számítógépet. Internet elérési lehetőség 88%-ban az iskolában és otthon is lehetséges, 3%-a az iskolát és az ismerősöket jelölte meg 9%-uk pedig csak az iskolában tud internetezni.

A diákok arról is nyilatkoztak, hogy számukra mi a fontos egy oktatóprogramban: a beépített játéklehetőség, a rajzfilmre emlékeztető animált betétek, a szöveges-állóképes magyarázatok, a gondolkodtató kérdések, a rövid filmbetétek vagy más, az oktatási és játékprogramokra is jellemző megoldás. Ennek eredményét tartalmazza a 20. ábra.



20. ábra

Kérdőívem következő kérdése arra kereste a választ, hogy jobb-e oktatóprogrammal tanulni. A 15-16 éves fiatalok 87 %-a felelt igennel 9% nemmel, 4% -uk nem tudom választ jelölte meg.

Kérdőívem következő kérdéséből kiderült, hogy a fiatalok 91%-a más tantárgyakban is szívesen szeretne oktatóprogrammal tanulni. A legtöbb érték öröndetesen pozitív válasz, amely szerint a .számítógépezés. élénkít, elégedetté tesz és ismereteket is közvetít.

A teszt hátralévő részében a multimédia-programom értékelésén volt a fő hangsúly, melyet az 5 fokú skálán lehetett elvégezni, ahol az ötös osztályzat volt a legjobb, 1 a legrosszabb eredmény.

Bár több multimédia-értékelési forma létezik a produktumok minősítésére, dolgozatomban egy kidolgozott – elsősorban pedagógiai, médiaismereti –

értékelési szempontrendszer alkalmaztam, mely nekem, a fejlesztőnek, valamint a pedagógiai felhasználóknak is eligazodást nyújthat. Igyekeztem több szempontú értékelést kialakítani. A számos értékelési szempontokból kiemeltem a leglényegesebbeket valamint azokat a kérdéseket, melyeket egy középiskolai tanulói könnyű szerrel tud pontokkal értékelni.

Három fő szempontot emeltem ki a teszt összeállításánál:

1. Általános és kommunikatív szempontok
2. Általános (pedagógiai, pszichológiai, ergonómiai szempontok)
3. Multimédia komponensek

Ezen szempontok általános szempontrendszer sugalmaznak, és kiválóan lehet segítségükkel gyors, átfogó képet kapni egy multimédia-produkcióról. A három fő vizsgálandó témakör 33 alkotóelemet vizsgál részletesen és pontoz az ötös skálán.

8.1 Kommunikáció, interakció

A kommunikációhoz, interakcióhoz használatos eszközöknek és jelzéseknek (szerszámok és ikonok) szerepelniük kell a programban. Ma már a felhasználók számára is könnyen kezelhető grafikus felhasználói felületeket alkotunk, amelyek egyre inkább emberközelivé, természetessé teszik az ember-gép kapcsolatát. Talán ennek köszönhető, hogy a diákok 95%-a úgy gondolja, hogy a képernyőn megjelenített információ világos, egyértelmű és könnyen olvasható. 5%-a pedig négyesre értékelte ezt a kérdést. A látvány és a működés úgy gondolom mindenkit lenyűgözött, mivel ötösre értékelte minden tanuló. A navigációs gombok kezelése se okozott számukra gondot ötösre értékelték ezt a szempontot.

8.2 Általános (pedagógiai, pszichológiai, ergonómiai) szempontok

Fő célom volt, hogy a program a felhasználó számára tartalmazzon újdonságot, érdekességet, a téma újszerű megoldását igyekeztem feldolgozni, a felhasználó szemszögéből közelíteni a tárgyhoz. A jó multimédia-program biztosítja, hogy a tanuló számára legmegfelelőbb tagolásban és ritmusban történjen a feldolgozás. A tanuló jegyzetelhet közben, ami segíti az elmélyült feldolgozást. A közlés multimediális, egyszerre több érzékszervi csatornára hat.

Ebben a szempontrendszerben 11 kérdést tettem fel. Az itt megjelenő szempontok több helyen bizonytalan választ kaptam, ami elgondolkodtatott, de végül meg kellett állapítanom, hogy a jobb eredmény elérésének a legjobb módja csupán az, hogy a diákoknak adni kell lehetőséget arra, hogy többször megnézzék a programot és, hogy ők maguk is motiváltak legyenek az új információ befogadására.

Többségben közepes eredményt kaptam a „pontos és megbízható, ösztönzi a felhasználót a tanulásra, és a teljesül-e a türelmes várakozás elve?” kérdésekre. Jó értékelést kapott a teljesül-e a megerősítés elve, megvalósul-e az önaktivitás elve, teljesül-e a figyelem fenntartásának elve, valamint az aktuális információkra vonatkozó kérdés. Jeles eredményt érdemelt az információk érthetősége, az anyag logikai felépítése, az érdeklődés fenntartása és a részek azonosíthatósága.

A fennmaradó kérdések ergonómiai szempontokat vizsgáltak. Az ergonómia a munkahelyzet, a hatékonyság és biztonság, az emberi munka minőségi összetevőivel foglalkozik. [9]

Tekintettel arra, hogy a számítógépeket, az interaktív programokat többnyire a hétköznapi emberek számára készítik, úgy tervezik meg, hogy használatukhoz ne legyen szükség számítástechnikai vagy egyéb informatikai előképzettségre.

A pszichológiai és ergonómiai szempontokat sikerült megvalósítanom, mivel szinte mindenki jeles eredményt adott a képernyő felépítésre, a

működésére, a befejezés és kilépés egyszerűségére, a felhasználóbarát kialakításra és a navigációs elemek azonos helyen történő elhelyezésére. Az egyedüli bizonytalanságot a fáradtság és feszültség mérése okozta, mivel 56%-uk nem tudom választ részesítette előnyben. A pontos eredményeket az *V. táblázat* tartalmazza.

Ergonómiai szempontok	3	4	5
Érthetően mutatja be az információt?	3%	9%	88%
Az anyag logikailag jól következik egymás után?	5%	12%	83%
A részek könnyen azonosíthatóak?	2%	5%	93%
Fenntartja az érdeklődést a tananyag iránt		2%	98%
Ösztönzi a felhasználót a tanulásra (interaktivitás)	14%	67%	19%
Teljesül-e a figyelem fenntartásának az elve	2%	71%	27%
Teljesül-e a türelmes várakozás elve	56%	26%	18%
Teljesül-e a megerősítés elve	7%	89%	4%
Megvalósul-e az önaktivitás elve	1%	92%	7%
Képernyőkép-felépítés		2%	98%
Működése és struktúrája egyértelmű a felhasználó szempontjából?		6%	94%
A műveletek befejezése és kilépés (menekülése) egyszerűsége			100%
Felhasználóbarát kialakítás		1%	99%
A navigációs elemek azonos helyre történő helyezése			100%

V. táblázat

8.3 Multimédia komponensek

A multimédiás oktatóprogramokban a kifejezési technikák sokoldalúságából adódóan a tankönyveknél kevesebb a szöveges információ a grafikus és a mozgóképekhez képest. A vizuális elemek szolgálják a legjobban az érdeklődés felkeltését. A tartalom vonzó, adekvát megjelenítése felhívja a tanuló figyelmét. Már a bejelentkező képen érezze az újdonságot, érdekességet, egy tématerület eredeti, látványos megközelítését. Az attraktív képek, háttérgrafika, arculattervezés szintén fokozza a tananyag iránti érdeklődést. [9]

A tanulási stílusnak való megfelelést nehéz biztosítani. Segítséget ad a médiakiválasztás, azaz a vizuális és verbális képességhez igazítás. A lineáris

vonalú tanulási stílust kedvelők számára az ismétlés, míg a csapongó stílust kedvelők számára az elágazások megléte a kedvező. Az átadandó ismeretet olyan formában kell megjeleníteni, amely a legjobban megfelel az adott információnak (állókép, grafika, írott információ, hang, animáció, mozgókép, szimuláció, teszt, gyakorlatok stb.). Fontos, hogy a termék a felhasználó szemszögéből közelítsen a témához, hiszen a tanulás csak akkor lehet hatékony, ha a tanulási folyamat során folyamatosan a tárgyra tudjuk irányítani a tanuló figyelmét. [9]

A szöveges részek egyszerűek, olvashatóak, tagoltak, tömörek voltak, megjelenésük a szemet nem zavarták érzékelhetően. A tudatos színhasználatot „nem tudom” válasszal közepesre értékelték. A szimbólumok, logók használata megfelelt eredményt kapott és viszonylag gyengén osztályozták a tanulók az animációk alkalmazását többségében közepes eredménnyel. Itt valószínűsítem azon feltevésemet, hogy még nem tudnak kritikusan gondolkodni a dinamika, a folyamatos mozgás, és a mozgás valószínűségével kapcsolatban. A pontos eredményeket a VI. táblázat tartalmazza.

	3	4	5
Egyszerűsége		3%	97%
Olvashatósága		3%	97%
Tagolása		6%	94%
Tömörsége		2%	98%
Szembarát megjelenítése		4%	96%
A tudatos színhasználat, a színhasználat elveinek a betartása	52%	43%	5%
A lényegkiemelés mértéke	13%	72%	5%
Az eredeti jelentésére való utalás	18%	63%	19%
Az animáció alkalmazásának indokoltsága	79%	10%	11%
Folyamatosság, egyenletes mozgás	74%	15%	11%
A mozgás valószínűsége	63%	24%	13%
Dinamika (gyors, kiegyensúlyozott, lassú)	56%	33%	11%
Figyelemfelhívó jellege		93%	7%

VI. táblázat

Multimédia-oktatóprogramok használata során a számítógéppel való közvetlen kommunikáció formájában történik az ismeretek elsajátítása, általában minimális számítástechnikai ismeretekre építve. Ahhoz, hogy a felhasználó optimálisan elsajátíthassa az adott tananyagot, nélkülözhetetlen a médiakompetencia egymásra épülő szintjeinek elsajátítása. A rendelkezésre álló szempontrendszerek birtokában lehetségesnek tartom, hogy a kezdő lépéseket azok a pedagógusok is megtegyék, akik még bizonytalanságot éreznek ezen a területen.

A tanulók értékelését figyelembe véve, úgy érzem sikerült egy olyan programot létrehoznom, mely a tanítás- tanulás folyamatát lényegesen megkönnyíti, fenntartva a tanulók érdeklődését, figyelmét. A programban nem várt módon, de nagy sikert aratott ez ellenőrző kérdések teszt szerű feldolgozása, ahol a diákok játékos formában ellenőrizhették tudásukat.

Összefoglalás

Azért választottam a multimédiát a dolgozatom témájának, mert a multimédia az oktatásra nagy hatással lehet a jövőt illetően. Az általános kommunikációs megközelítésből kiindulva a tanulás-tanítás kommunikációs jegyeit vegyük szemügyre. Nem mindegy azonban, hogy ez milyen hatásfokkal tudatosul. A verbális, akár élőszóban, akár írásban kibocsátott üzenetek mellett számos egyéb forma is fontos és nélkülözhetetlen. Ezek egy részét a multimédia és a hálózat rendkívül hatékonyra teszi, de a tapasztalás, nevelés, utánzás, gyakorlás megélt formáit nem lehet kiiktatni.

A képernyő meg növekedett vizualitása, a vizuális interakció korai és túlzott kihasználás esetén a verbális tudás és gondolkodásszerkezet gyengébb egyéni kiépítését okozhatja. Kérdés, hogy ez a kontrolált gondolkodásban pozitív vagy negatív képességeket fog e eredményezni. Igaz-e az, hogy a beszéd szekvenciális szerkezete, determinálja, tudatos megismételhető és kommunikálható gondolkodásunkat? Változtat-e ezen, ha növekszik a vizuális, grafikus kommunikáció a multimédia közvetítésével, és ha szavak, szövegszerkesztés helyett látványszerkesztés is rendelkezésünkre áll üzeneteink kifejezésére?

Természetesen szükség van a szöveges részekre, de ha közben az elmélet a monitoron láthatóvá is válik, az könnyebbé teheti a megértést és így rövidebb idő alatt juthatunk ugyanazon ismeretek birtokába, mintha órákon át ülnénk egy könyv felett. További előnyei közé sorolható, hogy a tanár instrukcióit többször is meg lehet hallgatni, az animációs magyarázatok többször is megtekinthetőek. Ezen túlmenően, nem csak a tanulók élvezhetik előnyeit, hanem lehetőség nyílik az oktatók, tanárok számára is, hogy multimédiás oktatói rendszereket bemutató órákon, egyfajta oktatási segédanyagként használják fel.

Irodalomjegyzék

Irodalomjegyzék

1. Athony F. Iasi: Az Intranet bibliája, Kossuth Kiadó, 1997.
2. Atkinson, R. L., Atkinson, R. C., Smith, E. E.: Pszichológia Budapest, Osiris-Századvég 1994.
3. Bonnie Blake: Macromedia Flash™ 5, Panem Kiadó Budapest, 2001.
4. Ch. Spanik - H. Rügheimer: A multimédia alapjai Budapest, Kossuth Kiadó, 1997. 5
5. Cserjés Á.: A szoftver ergonómia alapvető problémái. Ergonómia. VI. Ergonómiai konferencia Budapest 1994.
6. Cserny László: Számítógépek architektúrája Dunaújváros, 1996.
7. Dave Karlins: CorelDraw 8 Budapest, Panem Kft. 1998.
8. Deke McClland: CorelDraw 8 Budapest, Kossuth Kiadó 1998.
9. Forgó S.: Pedagógiai szemle, 2001.
10. Heico Neumeyer: Photoshop 5, Budapest, Panem Kft. 1999. 11
11. Izsó Lajos: A szoftver ergonómiai ismeretek oktatásának tapasztalatai, Budapest 1997.
12. Jakab Zsolt: Adobe Photoshop, Budapest, ComputerBooks 1998.
13. Journal of Computer Assisted Learning (1996) 12.
14. Kőhalmi Éva: CorelDraw 9, ComputerBooks Kiadó, 2000.
15. Kovács Ilma: Új út az oktatásban? Budapest, 1997.
16. Ralf Steinmetz: Multimédia Bevezetés és alapok, Budapest Springer Kiadó, 1995.
17. Robert Reinhardt, Jon Warren Lentz: Flash 5 Biblia, Kiskapu Kiadó, 2001.
18. Tay Vaughan: Multimédia, Making it work Macromedia, 1996.

Internet források

www.flashplanet.com

Egy magával ragadó, nagyon szépen kivitelezett honlap. Forrásanyagaival, gyakorlópéldáival és hivatkozásaival nagy segítségünkre lehet. A Flash 5 példafájlok felölelik valamennyi Flash 5 témakört, mint amilyenek például a hangokkal kapcsolatos alapismeretek (elindításuk, megállításuk, áthelyezésük, hangerejük, be és kiúsztatásuk), a függvények létrehozása, a találatstesztelő függvények, az egérmutató trükkök (elrejtés, megjelenítés, kicserélés).

www.were-here.com

E honlap több száz Flash 5 segédletet kínál számunkra. Néhány példa: egyedi egérkurzorok készítése, tömbök, többdimenziós tömbök, "fogd és vidd" technika, dinamikus szöveg, moziklipek vezérlése csúszkával.

www.flashguru.co.uk

Ez a honlap letölthető forrásfájllú Flash 5 példákat tartalmaz. Két oktatófilmet különösen megéri megvizsgálnunk: az egyik egy ütközésérzékelő, a másik egy véletlen szám (random) függvény. A honlap Resources menüpontja alatt a letölthető forrásfájlokat találjuk. A More Resources (még több erőforrás) menüpont alatt hangokat találunk.

www.flashkit.com

Egy rövid lista az ezen a honlapon található Flash 5 tananyagokról: az ActionScript használata (akció hívása, változó akciók beállítása, nyomkövetési akciók, felbukkanó menük létrehozása), munka szövegekkel (megtanulhatjuk például, miként állíthatunk elő egy szövegmaszk réteget alatta villogó képekkel), ProjectOr (a Flash ProjectOr fájlokat böngészőtől független médiák esetében használják, mint amilyenek például CD-k, e-mail-ek és a bemutatók).

www.flashkit.com

Más segédanyagok mellett, nyitott forráskódok és hangfájlok is letölthetők innen. Megvizsgálhatunk olyan tanpéldákat, melyekben hangokat importálunk, és videoklip szimbólumokat, gombokat és hangokat hozunk létre. A példák forrásfájljai szintén letölthetők. Mindemellett, hangokat is szerezhetünk innen, melyeket felhasználhatunk Flash-projektjeinkben. Ez a honlap a szerzői védelem alatt nem álló hangok egy hatalmas könyvtárát tartalmazza.

www.were-here.com

Ez a honlap is óriási zenei kínálattal áll rendelkezésünkre. Mind Macintosh AIF formátumú, mind Windows WAV formátumú hangokat és zenéket is letölthetünk.

www.creativepro.com

Fontos információkat találhatunk a Flash 5-ről, többek között annak audio-optimalizálási eljárásait illetően.

www.sitetracks.com

Egy kellemes kinézetű honlap, mely letölthető hangokat kínál.

Melléklet

1. számú melléklet Az oktatóprogram értékelésének kérdőíve

1. Milyen gyakran használasz számítógépet?

1. Naponta
2. Hetente
3. Havonta
4. Ritkábban
5. Soha

2. Hol szoktál internetezni?

1. Otthon
2. Ismerősöknél
3. Iskolában
4. Munkahelyen
5. Közintézményben
6. Egyéb helyen
7. Nem szoktam internetezni

3. Mi fontos egy oktatóprogramban?

- Játék*
- Animáció*
- Magyarázat*
- Kérdés*
- Videó*
- Egyéb*

4. Jobb így oktatóprogrammal tanulni?

Igen Nem Nem tudom

5. Szívesen használnál-e más tantárgyakból is oktatóprogramot?.

Igen Nem Nem tudom

A multimédia-program értékelését az 5 fokú skálán az alábbi skála szerint lehet elvégezni.

Egyáltalán nem tetszett (felelt meg)	Nem tetszett (felelt meg)	Nem tudom eldönteni	Tetszett (megfelelt)	Nagyon tetszett (Nagyon megfelelt)
1	2	3	4	5

I. ÁLTALÁNOS ÉS KOMMUNIKATÍV SZEMPONTOK	
1. A képernyőn megjelenített információ világos, jól szervezett, egyértelmű és könnyen olvasható?	
2. A működés és látvány kompatibilis a felhasználói hagyományokkal és feltételezésekkel?	
3. Tartalmazza a minimális navigációs elemeket	
II. ÁLTALÁNOS (PEDAGÓGIAI, PSZICHOLÓGIAI, ERGONÓMIAI) SZEMPONTOK	
4. Pontos és megbízható?	
5. Aktuális információkat tartalmaz?	
6. Érthetően mutatja be az információt?	
7. Az anyag logikailag jól következik egymás után?	
8. A részek könnyen azonosíthatóak?	
9. Fenntartja az érdeklődést a tananyag iránt	
10. Ösztönzi a felhasználót a tanulásra (interaktivitás)	
11. Teljesül-e a figyelem fenntartásának az elve	
12. Teljesül- e a türelmes várakozás elve	
13. Teljesül-e a megerősítés elve	
14. Megvalósul-e az önaktivitás elve	
III. PSZICHOLÓGIAI, ERGONÓMIAI SZEMPONTOK	
15. Képernyőkép-felépítés	
16. Működése és struktúrája egyértelmű a felhasználó szempontjából?	
17. A műveletek befejezése és kilépés (menekülése) egyszerűsége	
18. Felhasználóbarát kialakítás	
19. A navigációs elemek azonos helyre történő helyezése	
20. A használat közben tapasztalható fáradás, feszültség, frusztráció	
IV. MULTIMÉDIA-KOMPONENSEK	

21. Egyszerűsége	
22. Olvashatósága	
23. Tagolása	
24. Tömörsege	
25. Szembarát megjelenítése	
26. A tudatos színhasználat, a színhasználat elveinek a betartása	
27. A lényegkiemelés mértéke	
28. Az eredeti jelentésére való utalás	
29. Az animáció alkalmazásának indokoltsága	
30. Folyamatosság, egyenletes mozgás	
31. A mozgás valószerűsége	
32. Dinamika (gyors, kiegyensúlyozott, lassú)	
33. Figyelemfelhívó jellege	

Köszönetnyilvánítás

Ezúton szeretném megköszönni konzulensemnek Dr. Nyakóné Dr. Juhász Katalinnak szakmai segítségét, iránymutatását szakdolgozatom elkészítéséhez.