

**Česká zemědělská univerzita v Praze**

**Provozně ekonomická fakulta**

**Katedra informačních technologií**



**Bakalářská práce**

**Technologie Flash a její využití v praxi**

Zpracoval: Tomáš Kabourek

Vedoucí práce: Ing. Vladimír Očenášek

© 2011 ČZU v Praze

Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Akademický rok 2009/2010

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

**Tomáš Kabourek**

obor Provoz a ekonomika

Vedoucí katedry Vám ve smyslu Studijního a zkušebního řádu ČZU v Praze  
čl. 16 určuje tuto bakalářskou práci.

Název práce: **Technologie Flash a její využití v praxi**

### Osnova bakalářské práce:

1. Úvod
2. Cíl práce a metodika
3. Teoretická východiska
4. Použití Flashe k vytváření webů
5. Použití ActionScriptu pro akce a události
6. Závěr
7. Seznam použitých zdrojů
8. Přílohy

Rozsah hlavní textové části: 30 - 40 stran

Doporučené zdroje:

1. Shupe, Rich, Roster, Zevan. Leasing ActionScript 3.0. vyd. O'REILLY, 2008. ISBN-10: 0-596-52787-X, ISBN-13: 978-0-596-52787-7.
2. Lott, Joey, Schall, Darron, Petersl, Keith. ActionScript 3.0 Cookbook. vyd. O'REILLY, říjen 2006. ISBN: 0-596-52695-4.
3. Adobe Creative Team. Adobe Flash CS4 Professional. Oficiální výukový kurz, duben 2009. ISBN 978-80-251-2334-8.

Vedoucí bakalářské práce: **Ing. Vladimír Očenášek,**

Termín odevzdání bakalářské práce: duben 2011

  
.....  
Vedoucí katedry



  
.....  
Děkan

V Praze dne: 19. 2. 2010

## Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Technologie Flash a její využití v praxi" jsem vypracoval samostatně, pod vedením pana Ing. Vladimíra Očenáška, s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu literatury na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne .....

\_\_\_\_\_

## **Poděkování**

Rád bych touto cestou poděkoval panu Ing. Vladimíru Očenáškoví za vedení bakalářské práce.

## **Souhrn**

Bakalářská práce je zaměřená na problematiku technologie Flash, zejména možnosti jejího využití v praxi.

Práce poskytuje informace o vývoji této technologie v závislosti na trendech vývoje informačních technologií, možnosti využití nejaktuálnější verze Adobe Flash CS5 Professional pro různé projekty za podpory skriptu ActionScript 3.0.

V praktické části je využito poznatků ze studia dané problematiky a je zde popsán postup tvorby konkrétní webové stránky, kde jsou demonstrovány možnosti této technologie.

## **Klíčová slova**

Flash

ActionScript

Animace

Design

Prezentace

Vektor

Bitmap

## Summary

These bachelor theses are focused on the Flash technology, mainly the ways of its use in practice.

This work provides information about the development of the technology depending on the trends of information technology, the possibilities of using the latest version of Adobe Flash CS5 Professional for strange projects with ActionScript 3.0.

In the practical part, there were used knowledge of the study of the problem. There was described the procedure of creation a web page, where are demonstrated the possibilities of this technology.

## Keywords

Flash

ActionScript

Design

Presentation

Vector

Bitmap

1. Úvod.....	10
2. Cíl a metodika.....	11
2.1. Cíl práce .....	11
2.2. Metodika .....	11
3. Teoretická východiska .....	12
3.1. Vznik a vývoj .....	12
3.2. Intellidraw .....	13
3.3. FutureWave a SmartSketch.....	14
3.3.1. Microsoft, Disney a Macromedia Flash 1.0.....	15
3.3.2. Jednotlivé verze Macromedia a později Adobe Flash .....	16
3.4. Adobe Flash CS5 Professional (10.1).....	17
3.5. Nové funkce v CS5 .....	18
3.6. Budoucnost Flashe .....	19
4. Použití Flashe k vytváření webů.....	20
4.1. Jednotlivá odvětví využití .....	20
4.2. Animace .....	20
4.3. Prezentace .....	20
4.3.1. Tvorba grafiky .....	20
4.3.2. Hry .....	20
4.3.3. Webové stránky .....	21
4.4. Prostředí Adobe Flash CS5 .....	21
4.4.1. Časová osa .....	22
4.4.2. Inspektor Vlastností .....	22
4.4.3. Panel nástrojů.....	23
4.5. Publikování projektu .....	23



5.	Použití ActionScriptu pro akce a události.....	24
5.1.	ActionScript 1.0 a 2.0.....	24
5.2.	ActionScript 3.0 .....	26
5.2.1.	Kód umístění.....	26
5.2.2.	Používání datových typů.....	27
5.2.3.	Změny v použití proměnných.....	28
5.3.	Vytvoření webové stránky pomocí Flashe za použití ActionScriptu 3.0.....	29
5.3.1.	Menu .....	30
5.3.2.	Animace .....	31
5.3.3.	Video.....	35
5.3.4.	Fotogalerie .....	37
5.4.	Zhodnocení.....	39
6.	Závěr .....	41
7.	Seznam použitých zdrojů.....	42
8.	Přílohy.....	43

## 1. Úvod

V dnešní době si většina z nás neumí život bez internetu vůbec představit, určitá část si dokonce dokáže své stránky vytvořit, technologií pro tvorbu existuje celá řada. Jednu z nich si v této práci představíme. Jedná se o produkt společnosti Adobe, konkrétně software Adobe Flash.

Tato technologie patří do kategorie „Rich Internet Application“, mohla by tedy být vhodnou volbou pro společnost, která se chce na internetu prezentovat na co nejvyšší úrovni.

Technologie Flash už není zdaleka tak neobvyklá, jak tomu bývalo v minulosti. Nyní je Flash Player nainstalován na 98% počítačích připojených k internetu, už proto, že 75% videí na internetu potřebuje tento přehrávač. O možnostech Flashe svědčí i fakt, že 85 z top 100 webových stránek hodnocených prestižním serverem alexa.com využívá tuto technologii. Pokud vás ani toto nepřesvědčilo o významnosti této technologie a potřebou se jí zabývat, je možné doplnit ještě fakt, že Flash používají 3 000 000 vývojářů.

Je velmi nepravděpodobné, že se vás tato technologie nějakým způsobem nedotkla, ale zřejmě málokterý uživatel vnímá rozdíly ve využitých technologiích na webu až do chvíle, kdy něco nefunguje.

## 2. Cíl a metodika

### 2.1.Cíl práce

Dílním cílem této práce je prezentovat možnosti využití technologie Flash, charakterizovat základní funkce a jejich uplatnění. Práce by měla poukázat na schopnost Flashe nahradit celou řadu zavedených technologií pro tvorbu projektů, a především použitelnost pro důstojné prezentování firmy na internetu.

Dalším dílním cílem práce je vytvoření jednoduché Flash aplikace, ve které bude využita vektorová grafika, bitmapová grafika, animace, video a text z externího souboru.

### 2.2.Metodika

První část práce - literární rešerše - bude zpracována na základě analýzy literárních zdrojů. Využito bude odborných publikací světových autorů. Dojde k nastínění vývoje technologie Flash, jeho pronikání do různorodých odvětví grafiky, rozšiřování jeho možností a předpoklad dalšího vývoje.

V druhé části se zaměříme na to, čím je Flash atraktivní a čím si získává tolik příznivců.

Jako jeden ze zdrojů budou využity i oficiální výukové kurzy společnosti Adobe.

Třetí část literární rešerše se bude zaměřovat na to, co uživatel nevidí, ale přesto je to bezpodmínečně nutné pro fungování jakéhokoliv projektu ve Flashi, tedy ActionScript.

Praktická část této práce bude vytvořena na základě získaných informací ze studia dané problematiky. Projekt bude realizován v programu Adobe Flash CS5 Professional.

### 3. Teoretická východiska

#### 3.1. Vznik a vývoj

První Macromedia Flash je spjata s rokem 1996 a jménem Jonatan Gay. Uvedení Flashe na trh předcházela dlouhý vývoj, z programu podobného Malování, který je součástí Windows, až k profesionálnímu animačnímu prostředí aplikace Macromedia Flash. Sám Jonatan Gay popisuje celý vývoj na své oblíbené stavebnici Lego, podle jeho slov totiž vše začalo několika barevnými kousky plastu: „Jako dítě jsem vyrostl na hraní si s Legem, v době kdy ještě neexistovaly Lego panáčky nebo velryby nebo složité balíčky doplňků, pouze obyčejné kostky a pár kol. Tyto kousky barevného plastu mě naučily základy funkčního designu, řešení problémů a proces neustálého zdokonalování. Skvěle mi pomohl vyjádřit mé rané nadšení pro budování věcí.

Můj oblíbený projekt byl stavba Lego lodi se spoustou ramp, které mohou převážet má autíčka. To mě naučilo, že je třeba si vybrat problém, který mě inspiruje a vyzývá mě k hledání řešení s omezenými možnostmi.

Lidská mysl také limitována, nedokáže zachytit celou problematiku komplexně. S Legem můžete začít celkem a postupně dopracovávat detaily, což je víceméně stejný proces, jaký byl použit k vytvoření Flash.“ [7]

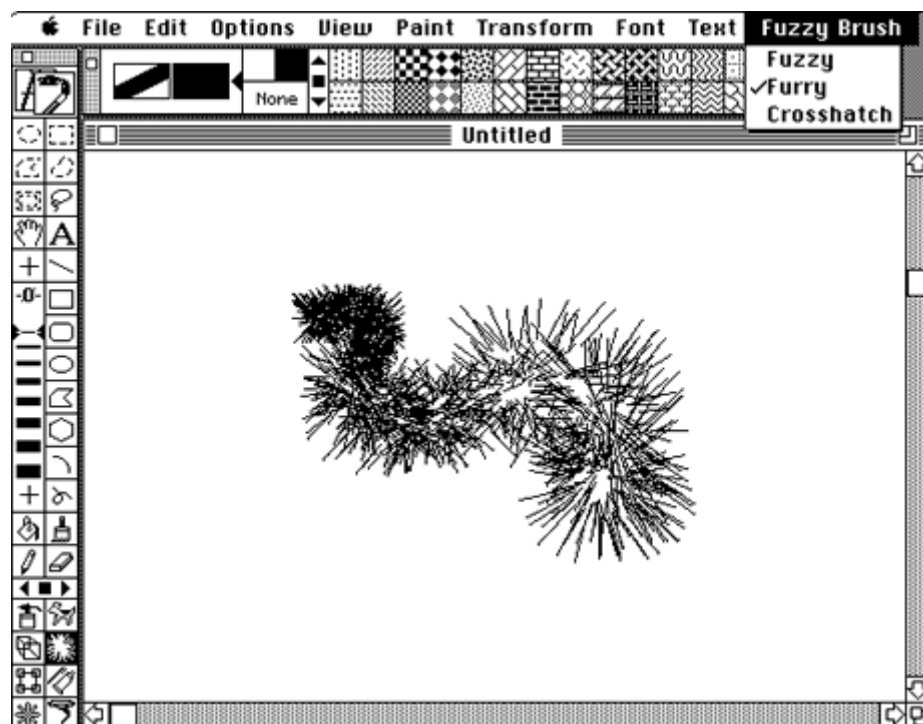
- Výběr problematiky: Stavba Lego lodi.
- Rozvoj myšlenek: Jaký druh lodi by to měl být? Jak velká by měla být? Co bude jejím úkolem?
- Stavba: Sestavení základního skeletu lodi.
- Doplnění detailů: Navržení rampy, dveří a dalších detailů.
- Test: Testování v praxi.
- Doladění: Změna částí lodi a snaha upravit je pro lepší funkčnost.
- Závěr: Získání nových poznatků pro tvorbu dokonalejší příští lodi.

Postupem času se mladý Jonatan Gay dostal k počítačům a k programování a zjistil, že se to od jeho oblíbené stavebnice zase tolik neliší. Začal psát první hry v Basicu, pak se ale naučil s Pascalem a napsal svůj první grafický editor pro Apple II. Na střední škole se seznámil Charlie Jacksonem, jehož snem bylo založit softwarovou firmu, ale neměl dostatek kapitálu na zaplacení profesionálních programátorů. Slovo dalo slovo a mladý

Jonatan Gay se dostal ke kvalitní počítačové technice a Jackson získal dobrého programátora, kterému nemusel platit dříve, než byl program uveden na trh.

Tvorbou her pro Silicon Beach Software Jonatan pronikl do oblasti animací, digitálního zvuku a jejich vzájemné provázanosti. [7]

Obrázek 1 – Gayův grafický editor



Zdroj: [http://www.flashmagazine.com/news/detail/the\\_flash\\_history/](http://www.flashmagazine.com/news/detail/the_flash_history/)

### 3.2. Intellidraw

Po řadě her se Gay vrátil k tvorbě grafického editoru. Intellidraw pro Aldus vznikl po ukončení vysoké školy a vedl k založení Jonatanovy vlastní firmy. Intellidraw byl nejenom kreslicím programem, bylo v něm možné i objektům přidávat jednotlivé funkce, bohužel se, pro svou komplikovanost, nedočkal předpokládané popularity. [7]

Obrázek 2 - Intellidraw



Zdroj: <http://www.ifanr.com/17526>

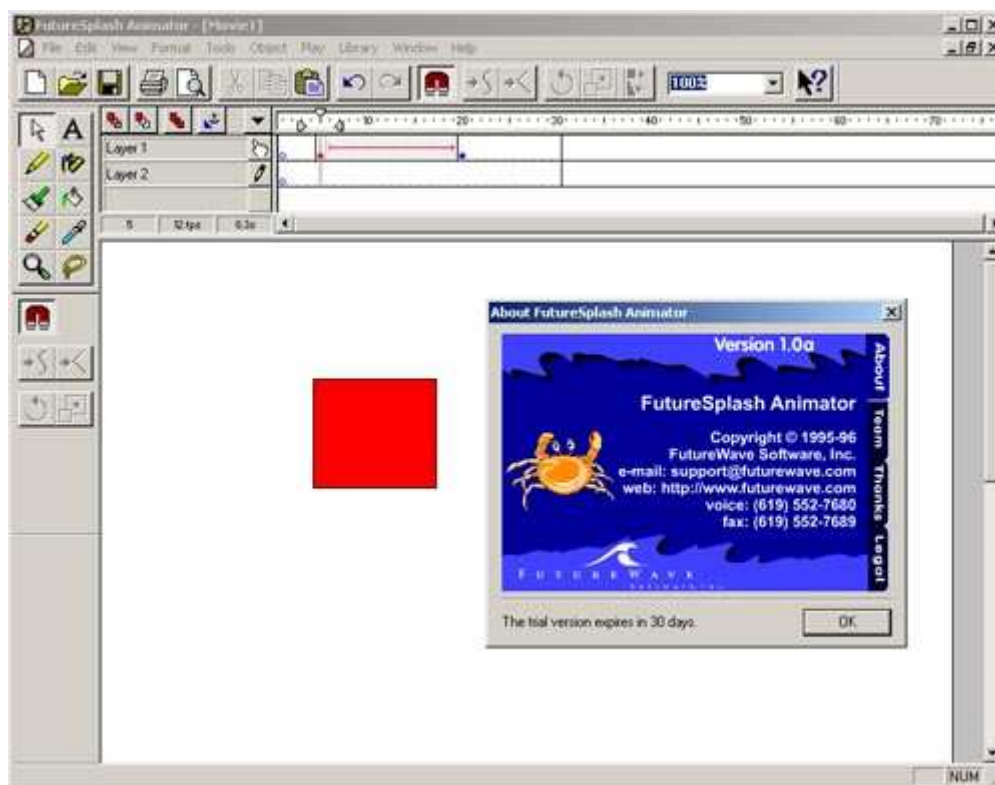
### 3.3.FutureWave a SmartSketch

Novým impulsem pro tvorbu byly tzv. pencomputers, neboli počítače, kde se píše přímo na obrazovku speciálním perem. Jonatan Gay vytvořil kreslicí program, v němž bylo kreslení stejně jednoduché a stejně rychlé, jako tužkou na papír, což bylo v porovnání s kreslením na prvním touchpadu od společnosti Macintosh skutečně velkým krokem kupředu. Společnost SiliconBeach představila program FutureWave v roce 1993. Zajímavostí bylo, že si ho pořídil i architekt, který v té době pracoval na domě Billa Gatese. Slávu FutureWave ukončil zánik operačního systému Go, pro který byl napsán. Bylo tedy třeba opět usednout ke klávesnici a posunout grafické programy o kousek dál. SmartSketch nebyl jen FutureWave předělaný pro Windows a Macintosh. V konkurenci programů, jako je Adobe Illustrator, jen těžko uspěl. Tvůrci se rozhodli, že mu přidají funkce pro tvorbu a úpravu animací. Jedním z důvodů, proč tvůrci zvolili animace, byl rozvoj internetu a předpoklad, že dojde k nárůstu popularity posílání animací a grafiky.

První animace přehrával jednoduchý program, který používal Javu, ale byl neúnosně pomalý. Časem byl nahrazen plug-inem API v prohlížeči Netscape, který se dále stal inspirací pro vytvoření Macromedia Flash Playeru.

V roce 1996 byl SmartSketch rozšířen o nové funkce a prodáván pod názvem FutureSplash Animator. [7]

Obrázek 3 – FutureSplash Animator



zdroj: <http://www.search.com/reference/FutureSplash>

### 3.3.1. Microsoft, Disney a Macromedia Flash 1.0

O dalším postupu společnosti Macromedia hovoří Jonatan Gay takto: „Náš velký úspěch přišel v srpnu 1996. Microsoft pracoval na MSN, snažili se vytvořit službu co nejpodobnější televiznímu vysílání. Tvůrci si velice oblíbili FutureSplash, a proto tuto technologii použili. Jsem stále ohromen, že použili v té době novou technologii, kterou vytvořila šestičlenná společnost! Naším dalším klientem byl Disney Online, ti použili FutureSplash pro tvorbu animace a uživatelského rozhraní pro Disney Daily Blast. Disney také pracoval s Macromedia Shockwave.

Macromedia o nás slyšela díky jejich vztahu s Disney, a tak jsme se časem dostali k spolupráci. FutureWave fungoval s celkovou investicí ve výši \$ 500 000. Možnost

většího kapitálu pro rozvoj FutureSplash se zdála výhodná, takže v prosinci roku 1996 jsme prodali FutureWave Software společnosti Macromedia a FutureSplash Animator se přejmenoval na Macromedia Flash 1.0.“[7]

### 3.3.2. Jednotlivé verze Macromedia a později Adobe Flash

Následující tabulka podává ucelený přehled o změnách ve vývoji jednotlivých verzí Macromedia Flash a později Adobe Flash. Některé změny, jsou zásadní, jiné méně důležité, ale každá z nich přispěla k současné podobě Adobe Flash.

Tabulka 1 - Změny Flashe během jeho vývoje

Název	Rok	Změny oproti předchozím verzím vydání
<b>Macromedia Flash 1</b>	1996	FutureSplash Animator pod křídly Macromedia
<b>Macromedia Flash 2</b>	1997	knihovna, které umožňuje ukládání objektů a jejich pozdější použití v různých projektech
<b>Macromedia Flash 3</b>	1998	MovieClip element, JavaScript plug-in, průhlednosti a externí samostatný přehrávač
<b>Macromedia Flash 4</b>	1999	interní proměnné, streamování MP3
<b>Macromedia Flash 5</b>	2000	ActionScript 1.0, podpora XML, Smartclips, HTML formátování textu přidáný pro dynamický text
<b>Macromedia Flash MX (6)</b>	2002	video kodek (Sorenson Spark), Unicode , v1 UI komponenty, komprese, vektorové kreslení ActionScript API
<b>Macromedia Flash MX 2004 (7)</b>	2003	ActionScript 2.0, chování, rozšiřitelnost vrstvu ( JSAPI), alias text podpora, efekty časové osy. Macromedia Flash MX Professional 2004 obsahuje všechny funkce aplikace Flash MX 2004, Screens (formuláře pro nelineární statické prvky a slidy pro uspořádání snímků, podobné jako je



		PowerPoint), integrace webových služeb, rozhraní pro import videa, přehrávání médiích komponentů (který je zjednodušeným přehrávačem MP3 a FLV, která může být umístěna v SWF), Datvé komponenty (DataSet, XMLConnector, WebServicesConnector, XUpdateResolver, atd.)
<b>Macromedia Flash 8</b>	2005	Rozdělena na Flash 8 Basic, který má pouze základní animační a grafické prvky, a Professional, který má oproti MX 2004 filtry, režimy prolnutí, zjednodušení kontroly animace, rozšíření možností grafiky (krytí a spojování), objektově založené kreslení, nově řešené vkládání textu, On2 VP6 video kodek, podpora průhlednosti ve videu, samostatný encoder a pokročilé vkládání videa, pokročilé přehrávání videa a interaktivní mobilní emulátor.
<b>Adobe Flash CS3 (9) Professional</b>	2007	Flash CS3 je první verze vydaná pod názvem Adobe. CS3 nabízí plnou podporu pro ActionScript 3.0, umožňuje celé aplikace převést do ActionScriptu, přidává lepší integraci s dalšími produkty Adobe, jako je Adobe Photoshop, poskytuje lepší vektorové kreslení, více se podobá Adobe Illustrator a Adobe Fireworks .
<b>Adobe Flash CS4 (10) Professional</b>	2008	Obsahuje inverzní kinematiky (kosti), základní prvky pro práci s 3D objekty, objektově-založené animace, textový engine, a další rozšiřování ActionScript 3.0. CS4 umožňuje vývojářům vytvářet animace s mnoha funkcemi chybějícími v předchozích verzích.

Zdroj: [8]

### 3.4. Adobe Flash CS5 Professional (10.1)

Tvůrci oficiálního kurzu píší o Adobe Flash toto: „*Aplikace Adobe Flash CS5 nabízí komplexní autorizační prostředí pro tvorbu počítačových animací a interaktivních webových stránek. Flash je široce rozšířená technologie používaná především k tvorbě*

*poutavých aplikací bohatě vyšperkovaných videosekvencemi, zvukem, grafikou a animacemi. Obsah lze vytvářet přímo ve Flashi, nebo jej importovat z jiných aplikací společnosti Adobe, jako jsou Photoshop či Illustrator, můžeme rychle navrhnout jednoduché animace a pomocí jazyka Adobe ActionSkript 3.0 vyvinout důmyslná interaktivní díla.*

*Adobe Flash Professional je jeden z produktů balíku nástrojů Flash CS5. Kromě Flash Professional nabízí Adobe také aplikace Flash Catalyst a Fash Builder. Flash Catalyst je designový nástroj pro tvorbu působivých rozhraní a interaktivního obsahu bez nutnosti psaní zdrojového kódu. Flash builder je prostředí pro psaní zdrojového kódu zaměřené spíše na vývojáře než grafiky či designéry, které slouží ke tvorbě interaktivního obsahu. Ačkoliv jsou tyto vývojové platformy odlišené, všechny tři nástroje nakonec vygenerují stejný produkt – flashový obsah (soubor SWF). Flashový soubor SWF poté spustíte v přehrávači Flash Player ve svém počítači, v prostředí AIR (Adobe Integrated Runtime) na své ploše vně prohlížeče nebo na svém mobilním telefonu.“[3]*

Flash CS5 byl dokončen 12. dubna 2010 a do prodeje nasazen 30. dubna 2010.

### 3.5. Nové funkce v CS5

- Nástroj Deco s rozšířenými výrazovými možnostmi, které umožňují jednoduše a automaticky vytvářet složité vzory a dekorace.
- Nástroj Text byl kompletně přepracován, aby podporoval pokročilejší prvky rozložení jako vícenásobného sloupce či zalamování textu.
- Funkce Pružina umožňující fyzikální simulace pro animace s inverzní kinematikou.
- Fragmenty kódu: nový panel nabízející předpřipravený zdrojový kód ActionScript pro jednotlivé projekty a způsob, jak ukládat a sdílet svůj zdrojový kód s ostatními.
- Živý náhled pro video nahrávané z externích zdrojů.
- Nový souborový formát XLF, který otevírá obsah flashových souborů a usnadňuje práci teamu vývojářů na jednom souboru. [3]

### **3.6. Budoucnost Flashe**

O budoucnosti Flashe se debatuje od chvíle, kdy Apple odmítl spolupráci s Adobe a novější přístroje od Applu nedostaly v základu žádný program pro přehrávání Flashe. Dalším vykřičníkem, je prohlášení Deana Hachamovitche, generálního ředitele divize Internet Explorer, že budoucnost vidí v HTML5, už kvůli tomu, že Flash má určité problémy, zejména v oblasti spolehlivosti, bezpečnosti a výkonu a kvůli problematické použitelnosti v mobilních zařízeních.[9]

## 4. Použití Flashe k vytváření webů

Pro představu můžeme ve Flashi vytvářet například fascinující internetové prezentace, účinné reklamní kampaně, obchodní aplikace, výukové kurzy, on-line hry či interaktivní video. Toto je pouze nastínění několika využití, celkové možnosti Adobe Flash jsou však mnohem širší.

### 4.1. Jednotlivá odvětví využití

Následující výčet možností by měl podat celkový přehled o tom, na co všechno lze Flash využít. Vzhledem k rozsahu a rozmanitosti možností využití se dá říci, že Flash patří softwaru s širokým spektrem použitelnosti.

### 4.2. Animace

Zřejmě nejčastěji se s animacemi setkáváme ve formě reklamních bannerů a videí např. na Youtube

### 4.3. Prezentace

Ve Flashi je možné vytvořit prezentace obdobné PowerPointu, ale i mnohem propracovanější a interaktivnější. Je například možné vytvářet rozcestí, což je vhodné pro výukové kurzy, kdy se oslovovaní mohou například rozhodovat pro různé alternativy vývoje prezentace, například výběr tématu.

#### 4.3.1. Tvorba grafiky

Adobe Flash umožňuje využití nejen vektorové a bitmapové grafiky, ale i jejich kombinací, například bitmapová výplň vektorového objektu. Adobe Flash umí i převod bitmapu na vektor a obráceně, bohužel ne v takové kvalitě, jaké by bylo třeba. Jedinou výhodou tak zůstává snazší upravitelnost bitmapové grafiky převedené do vektorové grafiky.

#### 4.3.2. Hry

Flashové hry jsou většinou velice jednoduché, nalezneme je například na [www.hry.cz](http://www.hry.cz), jejich výhodou je, že není třeba nic instalovat a je možné je rovnou hrát.

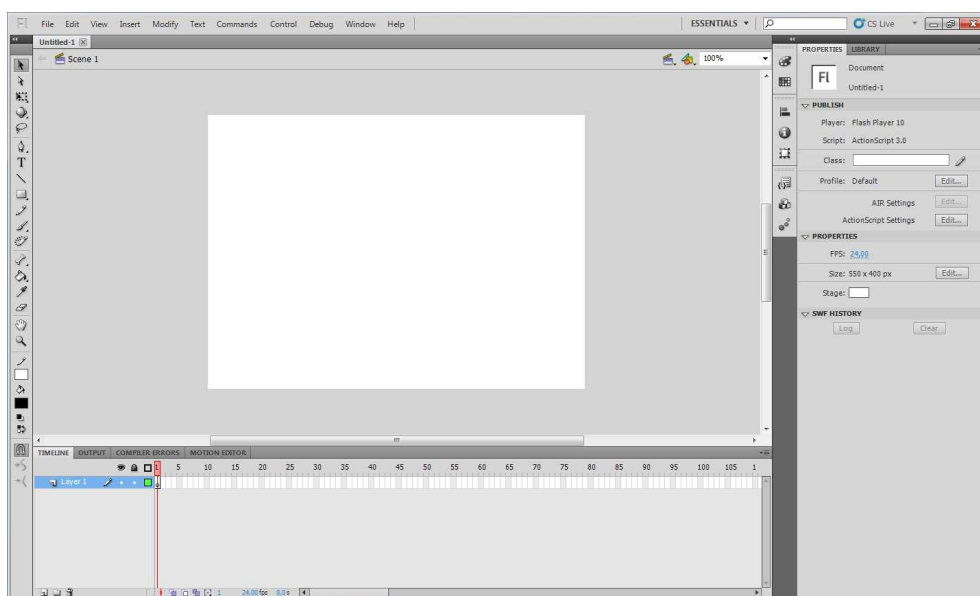
### 4.3.3. Webové stránky

Webové stránky ve Flashi vypadají profesionálně a čistě, i proto vyhrávají soutěže o nejlepší webový design. Flash se používá pro portfoliové nebo imagové stránky. Méně vhodný je však pro velké firemní stránky. Argumenty pro a proti Flashi se různí. Například v otázce bezpečnosti odborníci tvrdí, že Flash je velmi snadno odposlouchávatelný, a proto je nevhodný například pro e-shop, na druhé straně informace ohledně internetového bankovníctví Tatra banky hovoří jinak: „V rámci ďalšieho zlepšovania poskytovaného servisu sa banka rozhodla, že nová verzia internetového bankovníctva bude vybudovaná na novej technológii, ktorá poskytne priestor na ďalšie inovatívne riešenia určené klientom. Touto technológiou bude Adobe Flex®, ktorá je súčasťou Adobe®Flash® platformy. Ak vidíte pod týmto textom animáciu, znamená to, že máte správnu verziu softvéru Adobe Flash Player“, jak je zveřejněno na oficiální stránce <https://moja.tatrabanka.sk>.

### 4.4. Prostředí Adobe Flash CS5

Pro lepší představu o tvorbě webových stránek pomocí Flashe se seznámíme s prostředím programu Adobe Flash CS5. V horní části se nachází příkazové nabídky a pestrá škála nástrojů pro editaci a přidávání prvků filmu.

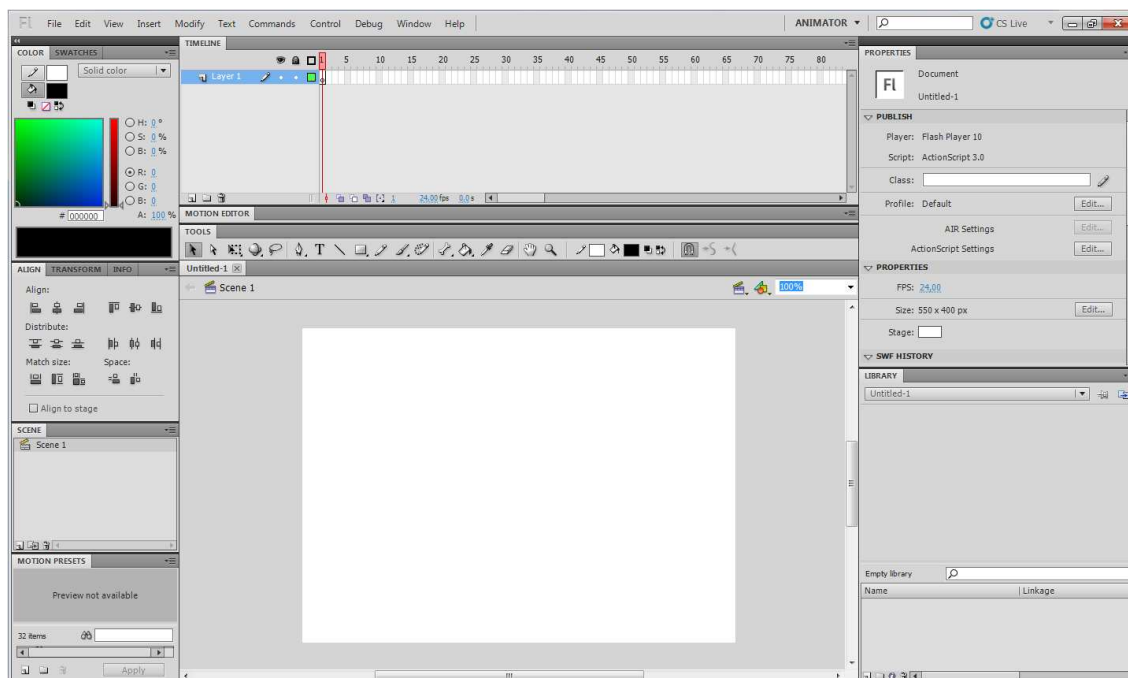
Obrázek 4 – Prostředí Adobe Flash CS5



Zdroj: vlastní

V konkrétním nastavení (viz obrázek 5) zobrazuje program panel nabídek, panel nástrojů, časovou osu, vymezenou plochu a inspektor Vlastnosti. Uživatel má možnost si panely poskládat podle svého uvážení a podle své vlastní potřeby, například rozložení pro animátora může vypadat jako na obrázku 5.

Obrázek 5 – Variace panelů nástrojů



Zdroj: vlastní

### 4.4.1. Časová osa

Stejně jako u filmu se i v dokumentaci Flash měří čas ve snímcích. Na panelu Časová osa můžeme měnit obsah jednotlivých snímků, ale i přidávat jednotlivé vrstvy. Výhodou používání vrstev je možnost kreslení a jiných úprav jedné vrstvy bez narušení jiné, která může být v tu dobu zamčena, nebo skryta (nezobrazená na vymezené ploše).[3]

### 4.4.2. Inspektor Vlastnosti

Inspektor Vlastnosti nabízí snadný přístup k nejčastěji používaným atributům. Jeho obsah závisí na typu zvoleného prvku. Pokud například není zvolen žádný prvek, nabídne inspektor Vlastnosti volby pro obecný dokument Flash, včetně změny barvy či rozměrů vymezené plochy. Po zvolení určitého objektu, zobrazí inspektor Vlastnosti

mimo jiné výšku, šířku, osovou souřadnici x a y daného objektu, které zde můžeme změnit zadáním přesné číselné hodnoty. [3]

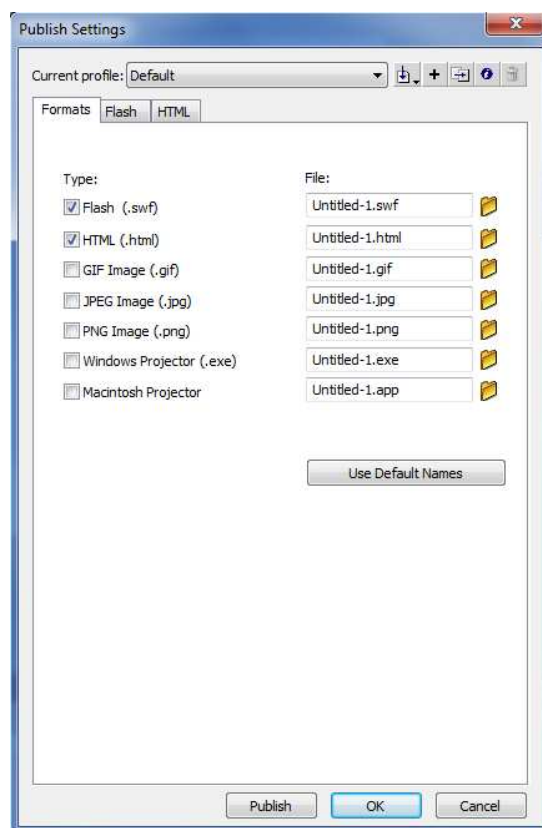
#### 4.4.3. Panel nástrojů

Panel nástrojů obsahuje nástroje pro výběr, kreslení, psaní, malování, úpravy, navigaci a volby nástroje. Pro každý nástroj existuje celé řada dílčích nastavení, například pro štětec nevybíráme pouze barvu, ale i tvar štětce nebo jeho velikost. [3]

#### 4.5. Publikování projektu

Jakmile je projekt hotový, můžeme jej publikovat, Adobe Flash nám k tomu nabízí celou řadu možností, stačí si vybrat z publikování formu. Pro webové stránky potřebujeme vygenerovat kód .html a flashový soubor .swf, máme ale na výběr i z bitmapových .gif, .jpg, .png, souboru pro Windows s příponou .exe, nebo souboru pro Macintosh s příponou .app. [3]

Obrázek 6 – Publish Settings



Zdroj: vlastní

## 5. Použití ActionScriptu pro akce a události

Aplikace Adobe Flash používá pro rozšíření funkcionality technologie Flash jazyk ActionScript, což je robustní skriptovací jazyk. Díky ActionScriptu, který je podobný jazyku JavaScript, je možné do projektů přidávat více interaktivních prvků, umožňuje přiřadit objektům, nebo časové ose jednotlivé vlastnosti, funkce nebo omezení. Pomáhá zvýšit interaktivnost výsledného Flashe.[4]

Pánové Joey Lott, Darron Schall a Keith Peters ve své knize ActionScript 3.0 Cookbook zdůrazňují, že společnost Macromedia byla první, kdo vytvořil metodu pro tvorbu webových stránek z kategorie „Rich Internet Application“. Podle těchto autorů se v ActionScriptu dají vytvořit jakékoliv aplikace, které si člověk dokáže představit. Výhodou je, že ActionScript používá stejný logický syntax, strukturu i koncepci, jako ostatní zavedené jazyky (C++, JavaScript), proto se tento jazyk dá zvládnout velmi rychle.[2]

### 5.1. ActionScript 1.0 a 2.0

ActionScript 1.0 se objevil v roce 2000 v Macromedia Flash 5. Protože je syntaxí shodný s ActionScriptem 2.0, pouze nemá tolik funkcí, přejdeme rovnou k vývojově mladšímu z této dvojice.

ActionScript 2.0 byl součástí Flashe od Macromedia Flash MX 2004. Tento skript používá několik základních prvků:

**Klíčová slova** jsou slova, která Flash chápe jako určitý příkaz či funkci, nesmí se tedy používat nikde jinde (například jako názvy proměnných).

**Složené závorky {}** Všechny příkazy, které jsou podmnožinou vyššího příkazu, jsou uzavřeny v závorkách. Česky řečeno *Když se stane toto {proved' tohle}*

**Oblé závorky ()** se používají k vyjádření určité hodnoty metodě nebo funkci. Například `gotoAndStop(2)` přesune časovou osu na druhý snímek.

**Středníky ;** se používají pro oddělení jednotlivých akcí.



**Uvozovky ""** se používají pro přidání textového řetězce nějakému parametru.

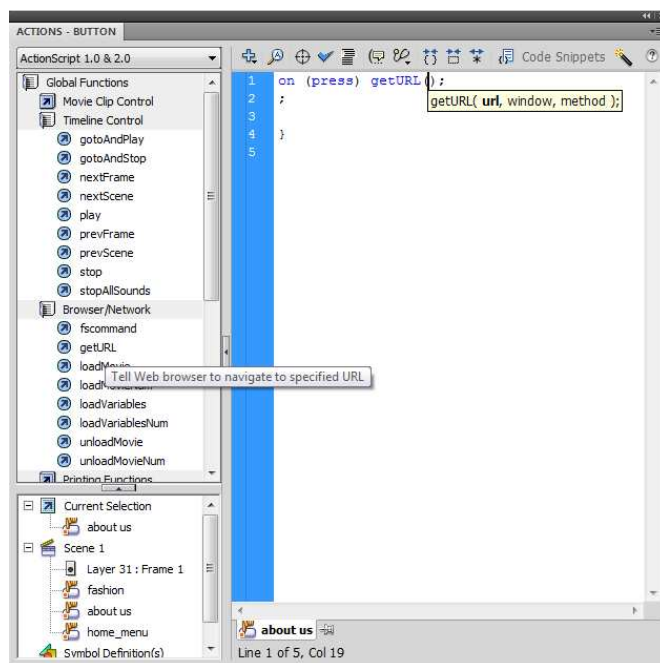
Je třeba si dát pozor na rozlišování textových hodnot a textových řetězců, protože "5" a 5 se v konečném zobrazení projeví úplně jinak.

**Komentáře //** se v programu Flash oddělují od ostatního textu dvěma lomítky.

Používají se pro poznámky, například pokud na projektu pracuje více tvůrců, nebo je kód složitý a je třeba se v něm rychle orientovat.[1]

Většina tvůrců webu si vystačí s několika málo příkazy, jako jsou zastavení na určitém snímku, přechod na jiný snímek po kliknutí na určité tlačítko nebo otevření stránky s jinou internetovou adresou. K takovýmto příkazům stačí základní znalost angličtiny, a to bez nadsázky. Pomocníkem je zde totiž výběr z jednotlivých příkazů v okně *Actions*, kde stačí jednotlivé položky vyhledat a dvojitým kliknutím přesunout do okna, případně doplnit jednotlivé parametry podle nápovědy. Zkušenější tvůrci mohou knihovnu schovat a psát buď s našeptávačem, nebo i ten vypnout. Pro kontrolu kódu je zde funkce *Check syntax*, která se ale může zdát trochu nadbytečnou vzhledem k barevnému rozdělení jednotlivých prvků a to následovně: klíčová slova modře, textové řetězce zeleně, neúplné příkazy červeně a ostatní černě.

Obrázek 7 – Okno ActionScriptu (Actions)



Zdroj: vlastní

ActionScript, podobně jako CSS nabízí několik možností, jak nadefinovat jednotlivé parametry, první možností je napsat jednotlivé parametry přímo do kódu ActionScript:

```
-default-size 800 600  
-default-background-color 0xffffffff  
-default-frame-rate 31
```

Druhou možností je potom využití tzv. metadat, které nejsou přímo interpretovány v ActionScript kódu, ale také se používají pro determinování. Konečné zobrazení bude zcela identické:

```
[SWF(width="800", height="600", backgroundColor="#ffffff",  
frameRate="31")]
```

## 5.2.ActionScript 3.0

Zřejmě nejvýstižnější komentář k ActionScriptu 3.0 vyslovil Dan Carr, vývojář a designer: *„Když jsem poprvé začal pracovat s ActionScript, nejtěžší pro mě bylo zvyknout si na změny v jednoduchých příkazech, na které jsem byl po léta zvyklý. Kam se poděla metoda attachMovie? Kde se podělo všechno z MovieClipu? Je to pocit, jako bych se musel učit znovu chodit těsně před velkým závodem.*

*V letech po vydání Adobe Flash CS3, jsem pracoval na plný úvazek na produkci SWF ActionScript 3 a Adobe AIR aplikace. Smál jsem se, plakal, narazil jsem do zdi (obrazně), několikrát. V závěru musím říct, že je těžké si představit návrat k ActionScript 2.0. V obecné rovině, skvělá věc na ActionScriptu 3.0 je, že všechno je standardizované: je tu jeden způsob, jak vytvořit projekt, jeden způsob, jak pracovat s událostmi, a tak dále. Pokud jste začátečník v kódování ActionScript, může tam toho pro vás být, co se množství týká, více, ale z hlediska náročnosti je to jednodušší na naučení a nakonec i vhodnější pro start.“*[6]

Následující body popisují zásadní změny v syntaxi jazyka v základní ActionScript 3.

### 5.2.1. Kód umístění

Jedna z nejvíce viditelných změn od předchozí verze ActionScriptu je, že můžete kód umístit přímo na instance. ActionScript 3 vyžaduje, aby byly všechny kódy umístěny

na klíčových snímcích na časové ose, nebo ve třídách ActionScript spojené s časovou osou. To znamená, že již nelze umístit kód přímo na tlačítko nebo filmový klip tak, jak tomu bylo v minulosti.

I když se na tuto změnu může hůř zvykat, jsou tyto normy výhodnější pro vytváření projektů, kde je kód jednodušší. V praxi to znamená, že projekt by měl mít jednu vrstvu na časové ose, ve které by byly všechny příkazy. Pokud je to možné, je dobré přidat kód do snímku 1, aby bylo snadnější jej najít. Samozřejmě můžete umístit akci „stop“ všude tam, kde danou akci budete potřebovat. Pokud jste se pracovali s kódem v souboru FLA před použitím tohoto konceptu, nebude to pro vás ničím novým, s rozdílem toho, že přidávání kódu do časové osy, je nyní přímo požadováno.[2]

### 5.2.2. Používání datových typů

ActionScript 3 využívá nový ActionScript Virtual Machine 2 (AVM2) v aplikaci Flash Player 9 a novější optimalizovaný kód. Jedním z klíčových prvků vztahujících se k optimalizaci je použití datového typu syntaxe, kdekoli jsou deklarovány proměnné, funkce, či prvky v časové ose. To umožňuje kompilátoru Flashe kontrolovat váš kód a identifikovat chyby, stejně jako pracovat rychleji a efektivněji za běhu.

Rozdíl mezi zápisem v ActionScriptu 2.0 a 3.0, lze to posoudit na příkladu dvou kódů:

#### ActionScript 2.0

```
_root.createEmptyMovieClip("kruh",1);
vykresli=function(){
  with(_root.kruh){
    x=Math.sin(++a*Math.PI/180)*50+100;
    y=Math.cos(++a*Math.PI/180)*50+100;
    lineStyle(2,0xff0000);
    moveTo(x-1,y-1);
    lineTo(x,y);
    if(a==360){
      clearInterval(int);
    }
  }
}
int=setInterval(vykresli,50);
```

#### ActionScript 3.0

```
sampleSprite.graphics.lineStyle( );
sampleSprite.graphics.beginFill(0xFF0000);
sampleSprite.graphics.drawCircle(100, 100, 50);
sampleSprite.graphics.endFill( );
```

### 5.2.3. Změny v použití proměnných

Proměnné jsou v podstatě stejné v ActionScript 3 jako v předchozích verzích s těmito výjimkami: je třeba používat klíčové slovo „var“ k deklarování proměnné a psát typ proměnné při deklaraci. Vynecháním klíčového slova „var“ vzniká error. Vynechání datového typu bude program hlásit varování nebo error v závislosti na prostředí, ve kterém píšete kód. [6]

Vzhledem k tomu, že všech příkazů je skoro tisíc, vybral jsem pro demonstraci pouze ty pro ovládání Flashe myší.

Tabulka2 - Rozdíly mezi ActionScriptem 2.0 a 3.0

ActionScript 2.0	ActionSkript 3.0	Rozdíl
Mouse (třída)	<a href="#">flash.ui.Mouse</a>	
<b>addListener()</b> Metoda	<a href="#">flash.events.EventDispatcher.addListener()</a>	V novém modelu událostí jazyku ActionScript 3.0 již není potřeba zvláštní metoda <code>addListener()</code> pro každou třídu, protože všechny objekty zobrazení dědí metodu <code>addEventListener()</code> ze třídy <code>EventDispatcher</code> .
<b>hide()</b> Metoda	<a href="#">flash.ui.Mouse.hide()</a>	Vrácená hodnota byla změněna na <code>void</code> .
<b>removeListener()</b> Metoda	<a href="#">flash.events.EventDispatcher.removeListener()</a>	V novém modelu událostí jazyku ActionScript 3.0 již není potřeba zvláštní metoda <code>removeListener()</code> pro každou třídu, protože všechny objekty zobrazení dědí metodu <code>removeEventListener()</code> ze třídy <code>EventDispatcher</code> .
<b>show()</b> Metoda	<a href="#">flash.ui.Mouse.show()</a>	Vrácená hodnota byla změněna na <code>void</code> .
<b>onMouseDown</b> Naslouchání	<a href="#">flash.display.InteractiveObject dispatches event: mouseDown</a>	V novém modelu událostí nahrazeno událostí <code>mouseDown</code> .
<b>onMouseMove</b> Naslouchání	<a href="#">flash.display.InteractiveObject dispatches event: mouseMove</a>	V novém modelu událostí nahrazeno událostí <code>mouseMove</code> .
<b>onMouseUp</b> Naslouchání	<a href="#">flash.display.InteractiveObject dispatches event: mouseUp</a>	V novém modelu událostí nahrazeno událostí <code>mouseUp</code> .
<b>onMouseWheel</b> Naslouchání	<a href="#">flash.display.InteractiveObject dispatches event: mouseWheel</a>	V novém modelu událostí nahrazeno událostí <code>mouseWheel</code> .

Zdroj: [5]

### 5.3. Vytvoření webové stránky pomocí Flashe za použití ActionScriptu 3.0

Jak bylo vytýčeno v cílech této práce, vyzkoušíme prakticky využití technologie Flash na jednoduché webové stránce za použití ActionScriptu 3.0, stránka bude obsahovat prvky vektorové i bitmapové grafiky, video a text.

V prvním kroku si nastavíme projekt v ActionScriptu 3.0 a rozložení stránky, které bude 800 x 600 pixelů, barva pozadí bude standardně bílá. Nyní si vytvoříme vrstvy, do kterých rozdělíme jednotlivé objekty. Celkem budeme pracovat s 6 vrstvami, první nazveme „Pozadí“. Tato vrstva bude zařazena úplně dole a ostatní vrstvy jí budou překrývat. Do vrstvy „Pozadí“ vložíme logo České zemědělské univerzity v Praze a převedeme jej do vektorové grafiky pomocí funkce „Trace Bitmap“. Na obrázku je vidět, že při větším zvětšení je logo nedokonalé, pro porovnání vpravo je logo převedené pomocí programu CorelDRAW X3 a zkopírované do Flashe, je evidentní, že Corel je lepší pro převod grafiky, ale projevil se nedostatek v převodu textu.

Obrázek 8 – Srovnání převodu grafiky



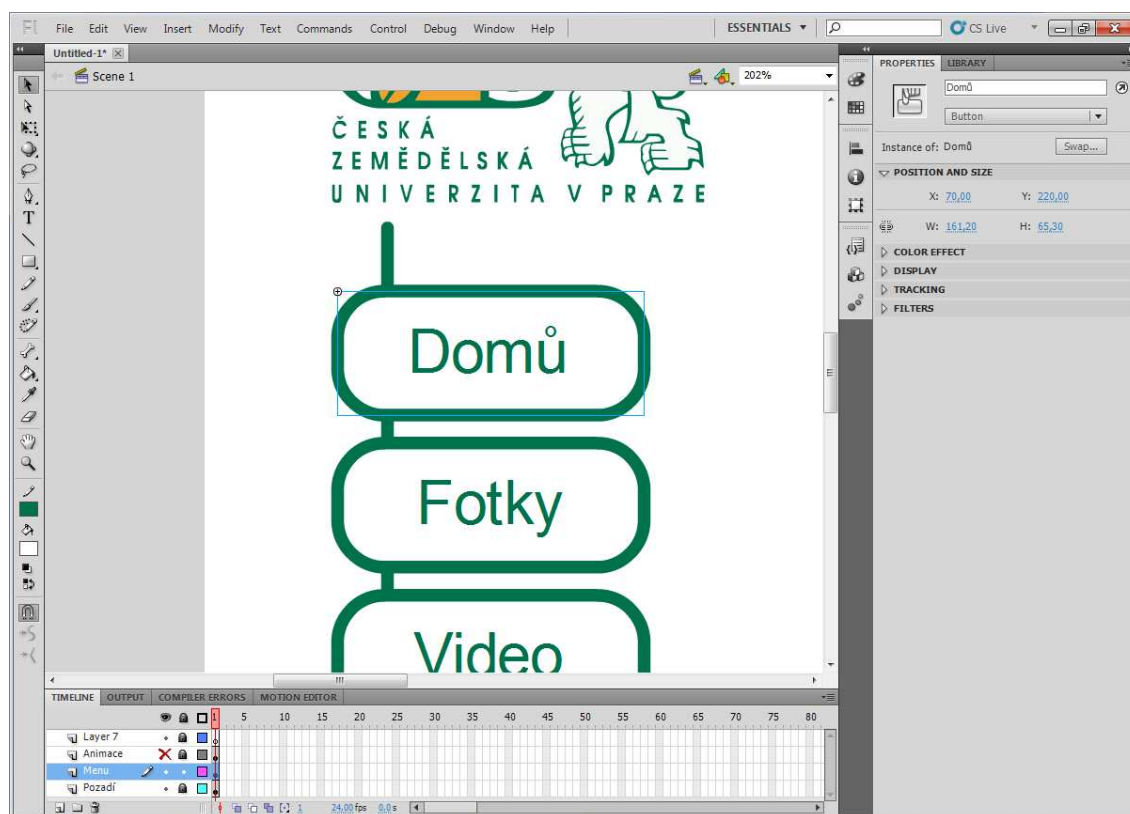
Zdroj: vlastní

Do vrstvy pozadí přidáme ještě jednoduchý rámeček, konkrétně obdélník bez výplně se zakřivením rohů 25, který vyřízneme v levém horním rohu pro umístění loga ČZU.

### 5.3.1. Menu

Druhá vrstva bude určena pro menu, pojmenujeme jí tedy „Menu“. Menu se bude skládat z 3 tlačítek a to „Domů“, „Fotky“ a „Video“. Každé tlačítko bude obdélník se stejným zakřivením a stejným zvýrazněným obrysem, jako obdélník v pozadí. Nejdůležitějším krokem je nyní deklarování prvku, do kolonky „Instance name“ napíšeme jméno prvku a vybereme z nabídky „Button“, nyní můžeme pomocí ActionScriptu přidávat tomuto prvku vlastnosti a funkce.

Obrázek 9 - Editace tlačítka



Zdroj: vlastní

Flash nabízí celou řadu efektů pro tlačítka, my využijeme změny tlačítka při kontaktu s kurzorem a přidáme zvuk. Vše se dělá pomocí editace tlačítka, můžeme zde nastavit vlastnosti pro „Up“, „Over“, „Down“ a „Hit“, tedy normální zobrazení, zobrazení při kontaktu s kurzorem, při stisknutí tlačítka a pro tlačítko, které již bylo stisknuto, jediné,

co je třeba udělat, je přidat klíčové okno do daného okna na časové ose a upravit tlačítko podle svého uvážení, popřípadě i přidat zvuk jednoduchým přetažením.

Zvuk tlačítka můžeme i vyvolat pomocí ActionScriptu, výsledek ale není ideální, zvuk se neozve vždy a pokud ano, má poměrně velkou prodlevu, kód by vypadal takto:

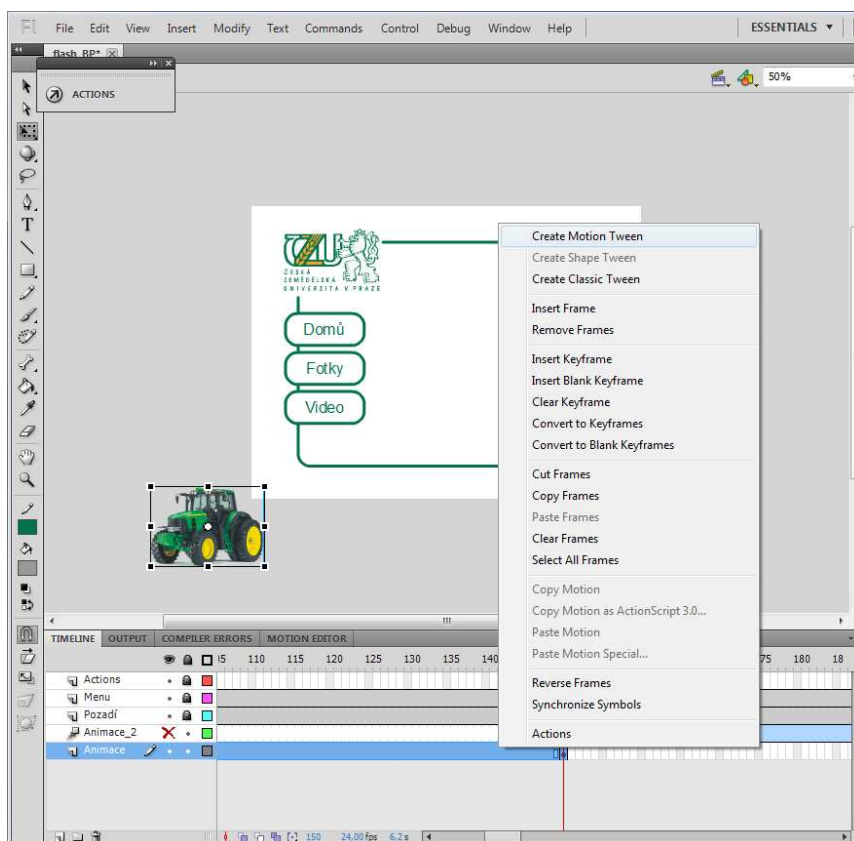
```
Domů.addEventListener(MouseEvent.CLICK, zvuk_tlacitka);
var prehrani:SoundChannel;
function zvuk_tlacitka(evt:MouseEvent):void
{
    var s:Sound = new Sound(new URLRequest("umístění zvuku na síti"));
    prehrani = s.play();
}
```

### 5.3.2. Animace

Protože chceme, aby naše flashová stránka byla co nejzajímavější, přidáme do ní animace. Vložíme je do samostatné vrstvy „Animace“ a „Animace\_2“, aby se nám s nimi lépe pracovalo. Pro tento projekt použijeme 2 obrázky, které budou střídavě projíždět z jedné strany obrazovky na druhou, zároveň budou měnit svou velikost. Obrázky převedeme do vektorové grafiky, podobně jako logo na pozadí, využijeme zde toho, že vektorová grafika se při změně velikosti nedeformuje.

Animaci musíme vložit jako „Movie Clip“, protože je nutné, aby se animace pohybovala nezávisle na hlavní časové ose. K animaci využijeme funkci „Create Motion Tween“, která nám velmi snadno vytvoří z počátečního a koncového stavu objektu lineární animaci, a to jak velikostní, tak pohybovou.

Obrázek 10 – Tvorba animace



Zdroj: vlastní

Na časovou osu vložíme další 3 vrstvy, bude se jednat o „Úvod“, „Fotky“ a „Video“, to budou jednotlivé stránky, které se zobrazí po kliknutí na dané tlačítko menu. Do ActionScriptu přidáme tyto funkce:

*stop();* pro zastavení flashe na prvním okně

```
Domů.addEventListener(MouseEvent.CLICK, fl_ClickToGoToAndStopAtFrame_2);  
function fl_ClickToGoToAndStopAtFrame_2(event:MouseEvent):void  
{  
    gotoAndStop(1);  
}
```

Pro přesunutí a zastavení na prvním okně flashe po kliknutí na tlačítko „Domů“



```
Fotky.addEventListener(MouseEvent.CLICK, fl_ClickToGoToAndStopAtFrame);  
function fl_ClickToGoToAndStopAtFrame(event:MouseEvent):void  
{  
    gotoAndStop(2);  
}
```

Pro přesunutí a zastavení na druhém okně flashe po kliknutí na tlačítko „Fotky“

```
Video.addEventListener(MouseEvent.CLICK, fl_ClickToGoToAndStopAtFrame_3);  
function fl_ClickToGoToAndStopAtFrame_3(event:MouseEvent):void  
{  
    gotoAndStop(3);  
}
```

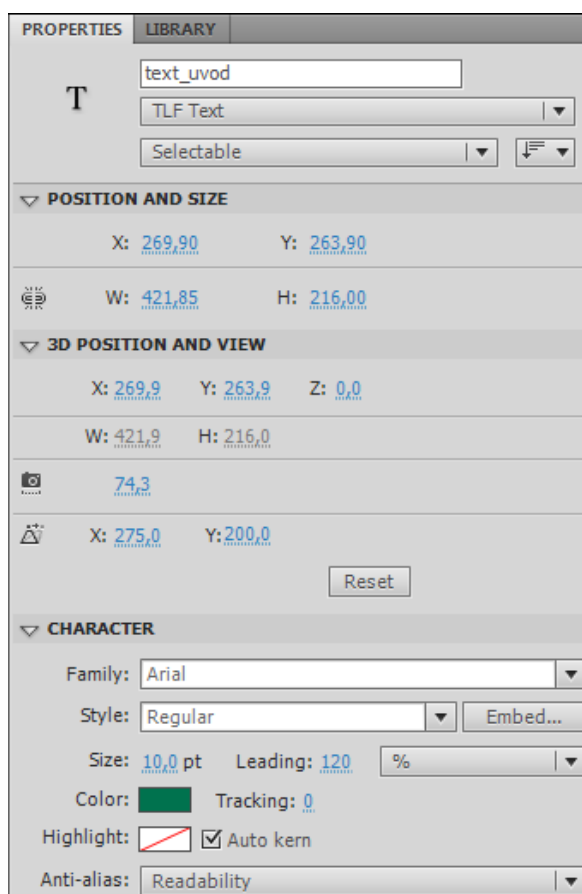
Pro přesunutí a zastavení na třetím okně flashe po kliknutí na tlačítko „Video“.

Funkce pro posun má název „*fl\_ClickToGoToAndStopAtFrame\_3*“ mohla být ale nazvána „*vezmitoaposunjinam*“ to záleží pouze na tvůrci.

Do prvního okna vrstvy úvod vložíme text v následujícím znění: „Vítejte na stránkách České zemědělské univerzity“. Můžeme vybrat barvu, velikost, font a další vlastnosti. Problém ale nastává při použití jiných možností vkládání textu. Jedním takovým příkladem je textové pole, které používá text z externího souboru, to se dá využít například pro editaci aktualit, aniž bychom nějakým způsobem zasahovali do samostatného flashového projektu.

Prvním krokem je vytvoření textového pole. Toto textové pole musí mít svůj název a musí mít přiřazené vlastnosti. V našem případě je názvem „*text\_uvod*“, font „*Arial*“, velikost 10 a barva zelená.

Obrázek 11 – Vlastnosti textu



Zdroj: vlastní

Pro vyvolání textu použijeme ActionScript. Tento skript nám načte „text.txt“ a zobrazí v textovém poli „text\_uvod“:

```

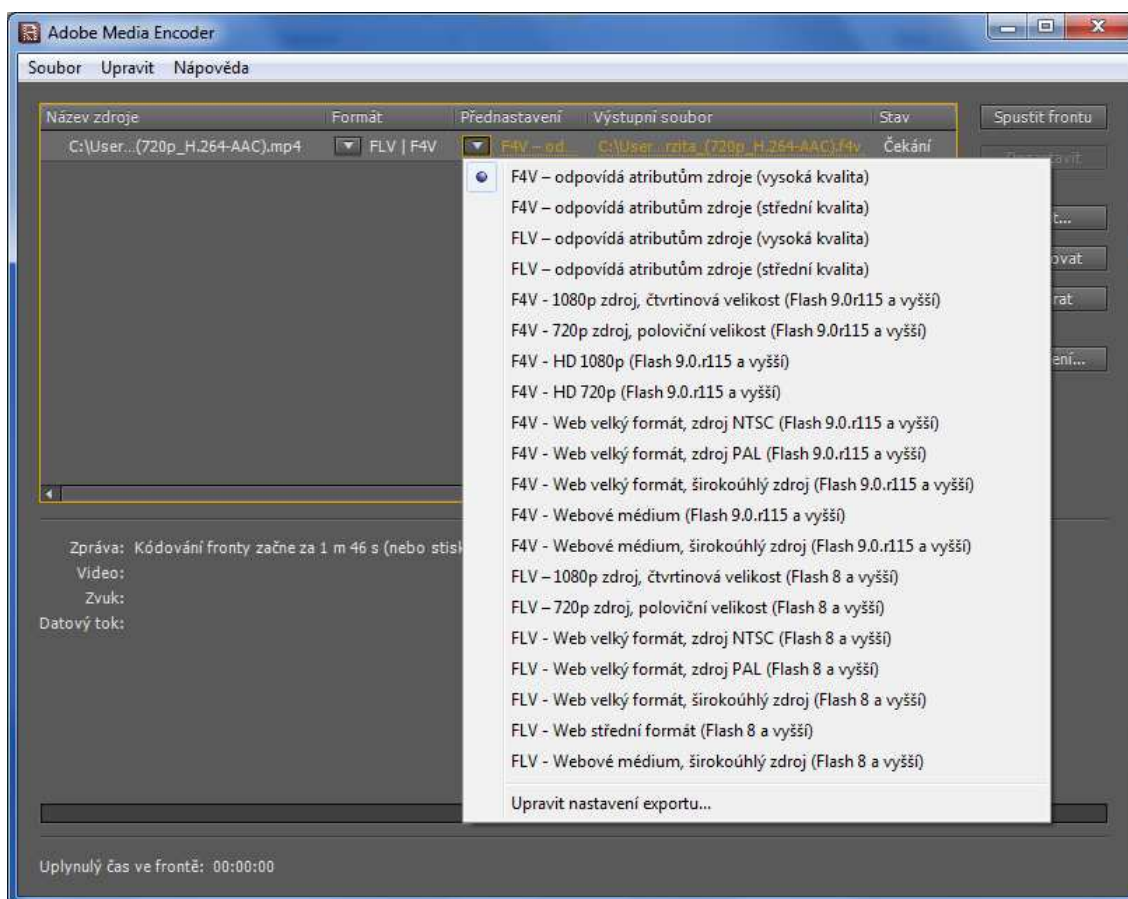
var fl_TextLoader:URLLoader = new URLLoader();
var fl_TextURLRequest:URLRequest = new URLRequest("text.txt");
fl_TextLoader.addEventListener(Event.COMPLETE, fl_CompleteHandler);
function fl_CompleteHandler(event:Event):void
{
    var textData:String = new String(fl_TextLoader.data);
    text_uvod.text = textData;
}
fl_TextLoader.load(fl_TextURLRequest);
    
```

### 5.3.3. Video

Video se dá ve Flashi zobrazovat dvěma způsoby. První možností je ponechání videa v samostatném souboru, který se přehraje pomocí komponentů pro přehrání videa. Druhou možností je vložit video přímo do projektu.

Prvním krokem je převedení videa do správného formátu, k tomu použijeme Adobe Media Encoder, který je součástí sady Adobe Flash Professional CS5. Jako výstupní vybereme F4V – Webové médium (Flash 9.0r115 a vyšší) z důvodu menší velikosti souboru, byť na úkor kvality.

Obrázek 12 – Převod videa

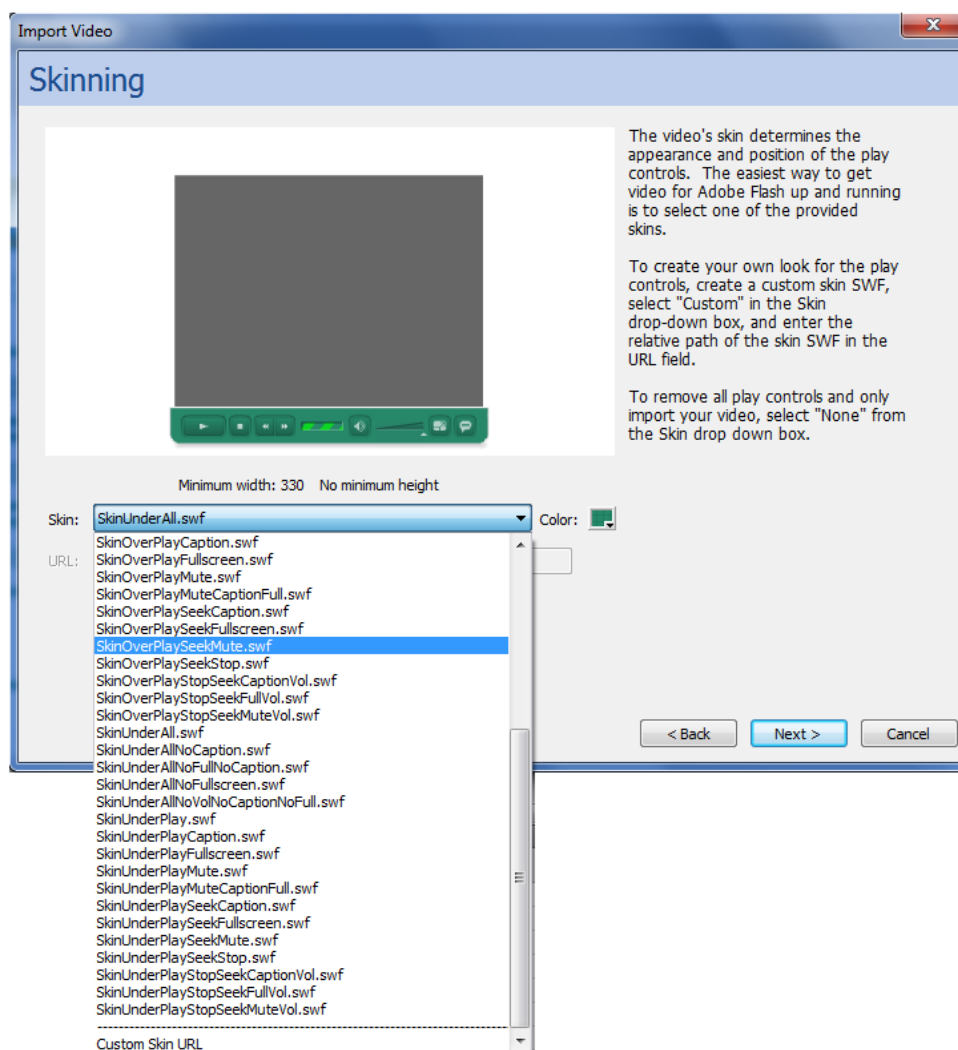


Zdroj: vlastní

Pro importování již přeformátovaného videa použijeme funkci „Import Video“, poté nám Adobe nabídne celou řadu možností volby vzhledů přehrávače, každý si může každý vybrat podle svého uvážení. Pro náš projekt zvolíme skin „SkinUnderAll.swf“

a barvu ovladačů stejnou, jako jsme použili na pozadí. Tu vybereme buď nastavením kódu barvy „00724E“, nebo pomocí „kapátka“, které po kliknutí na jakýkoliv bod na ploše přesně namíchá danou barvu.

Obrázek 13 – Import videa



Zdroj: vlastní

Video umístíme do třetího okna do vrstvy „Video“, tam již máme nastavený cíl v ActionScriptu, proto po kliknutí na tlačítko „Video“ bude mít prohlížeč možnost shlédnout video, které se začne neprodleně přehrávat.

### 5.3.4. Fotogalerie

Posledním krokem před exportováním celého projektu bude vytvoření fotogalerie. Možností je celá řada, dá se například vytvořit galerie rotující kolem určitého bodu, jak bylo použito na webu [www.unicate.cz](http://www.unicate.cz).

Obrázek 14 – 3D fotogalerie



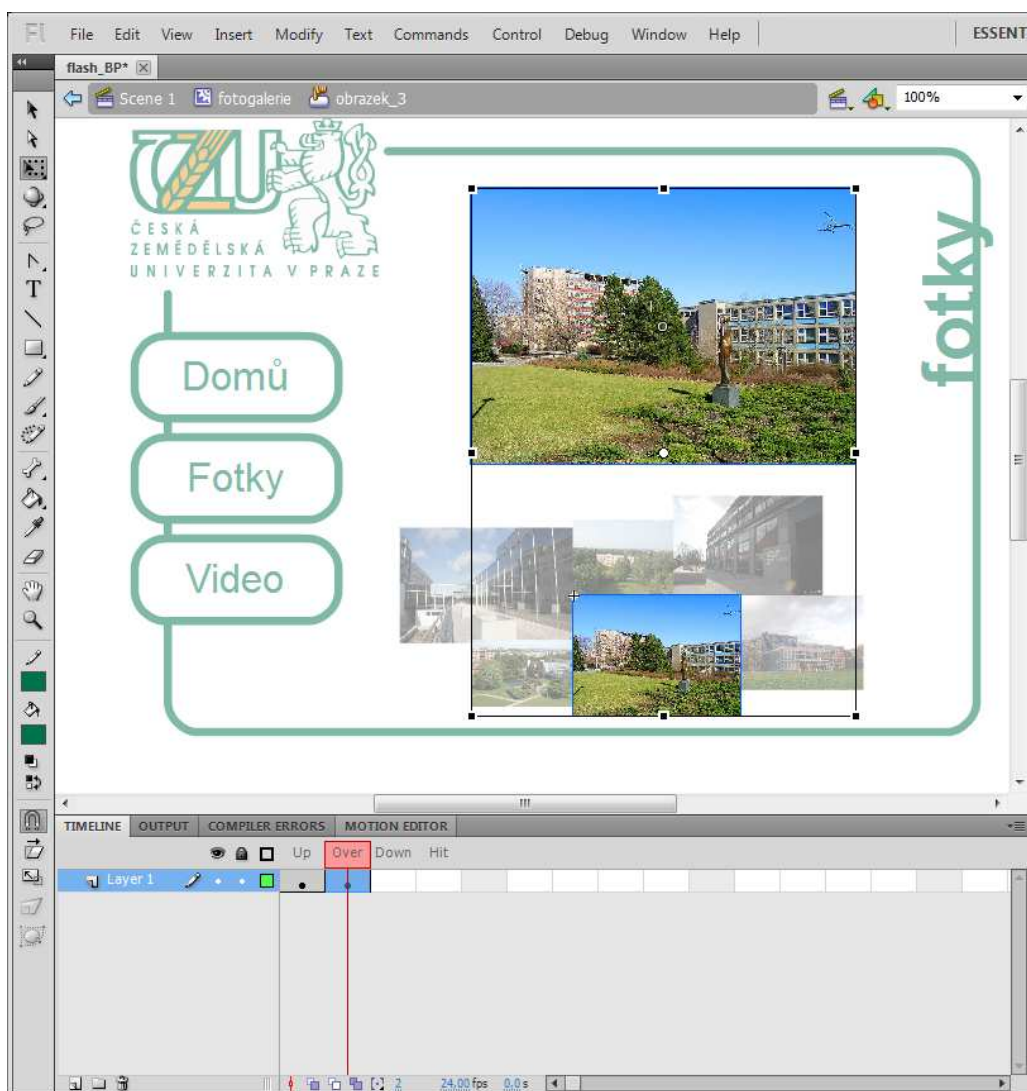
Zdroj: [www.unicate.cz/fashion](http://www.unicate.cz/fashion)

Pro náš projekt ale použijeme fotogalerii v ploše. Celá galerie bude vložena do projektu jako „Movie Clip“, to je z důvodu snazší orientace při úpravách. Naše galerie se bude skládat z šesti fotek, které můžeme umístit do řady ale také nemusíme, protože se Adobe Flash chová jako WYSIWYG, tedy výsledný formát .swf bude vypadat totožně s pracovním formátem .fla. Fotky zmenšíme, aby se vešly na zhruba polovinu akčního pole.

Každou z fotek změňme na tlačítko a pojmenujeme je analogicky „obrazek\_1“ až „obrazek\_6“. Naším cílem je, aby se po kontaktu myši s miniaturou objevil obrázek v horní polovině. Toho docílíme tak, že do okna „Over“ vložíme klíčové okno a vložíme do něj velký obrázek na místo, kde chce, aby se nám zobrazoval. Jeho polohu můžeme poupravit v panelu „Properties“, kde lze doladit polohu a velikost tak, aby

měly všechny objekty shodnou velikost a polohu, to umí Adobe Flash s přesností na dvě desetiny pixelu. Celý Movie Clip vložíme do 3. okna časové osy do vrstvy „Fotky“.

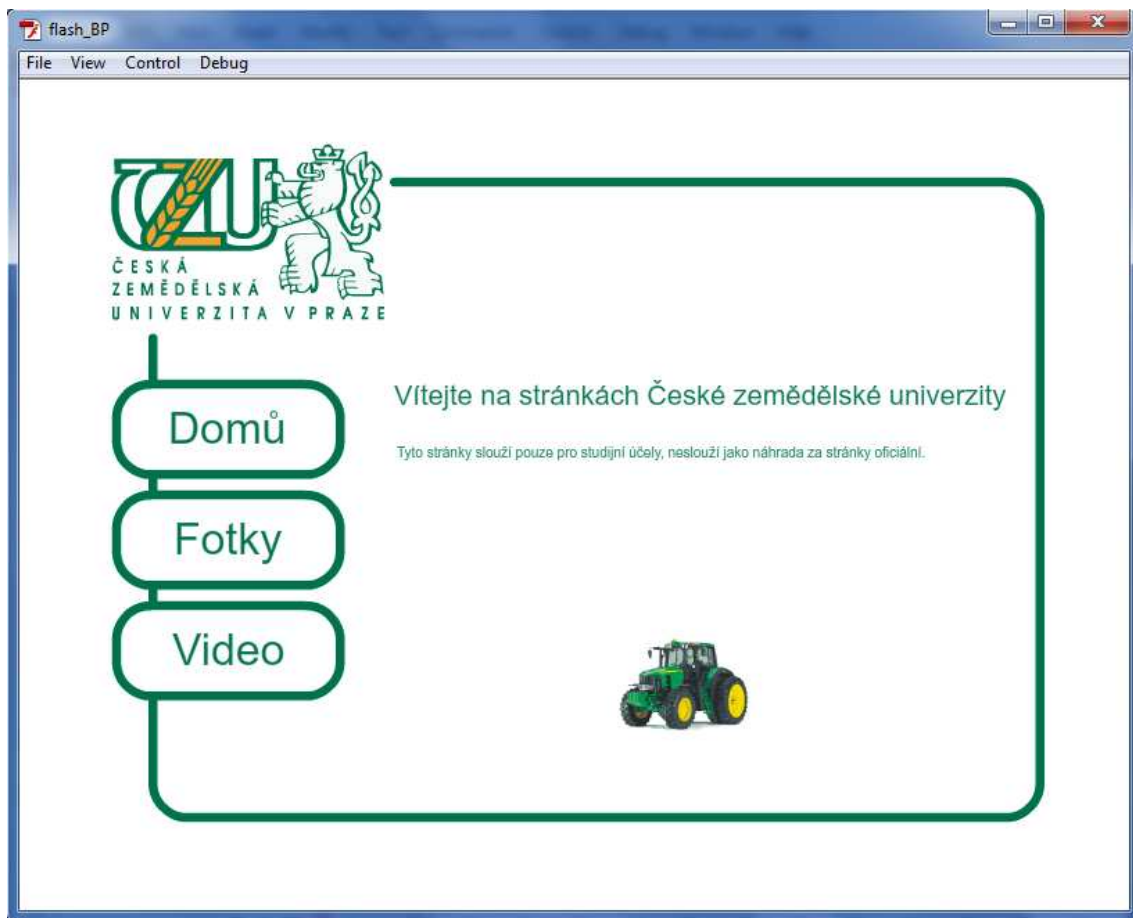
Obrázek 15 – Řešení fotogalerie



Zdroj: vlastní

Celý projekt si můžeme nyní prohlédnout pomocí funkce „Test Movie“ a vyzkoušet, jestli fungují všechny funkce. Pokud bychom měli některý ze skriptů napsaný špatně, dostaneme přesnou informaci o chybě v panelu časové osy, konkrétně v „COMPILER ERRORS“, kromě toho se nám nezobrazí žádné objekty, kterým je předepsán jakýkoliv skript. Chyby si můžeme kontrolovat i na nejspodnější liště programu Adobe Flash, kde je v ideálním případě hláška: „0 Error(s) 0 Warning(s)“.

Obrázek 16 – Dokončený projekt



Zdroj: vlastní

Nyní již je na našem uvážení, jak tento projekt budeme prezentovat, původně byl koncipován jako internetová stránka, proto v nabídce „Publish Settings“ vybereme pro publikování položky „HTML (.html)“ a „Flash (.swf)“, které nyní stačí nahrát na server a webová stránka je hotova.

#### 5.4. Zhodnocení

Při realizaci projektu se neobjevil jediný problém týkající se omezenosti možností Flashe. Vše je i v konečném zobrazení na internetu naprosto totožné s původními myšlenkami tvorby.

Celý projekt samozřejmě mohl vzniknout v alternativních programech, které umí generovat soubory s příponou .swf, například KoolMoves, SWiSHMax, nebo i

CorelDraw, ale všechny tyto programy jsou pouze náhražkou a původní Adobe Flash plnohodnotně nenahrazují, protože nenabízí tolik funkcí.

Navíc pokud bychom chtěli tento projekt vytvořit pomocí jiné, podobné technologie, budeme se pohybovat mezi poměrně mladými technologiemi. Konkurenci trvalo dlouhou dobu, než dokázala na Flash adekvátně odpovědět. Například Microsoft Silverlight byl představen v roce 2007, Flash má oproti Silverlightu výhodu v léty prověřené historii, naproti tomu Silverlight je použitelný i ve Windows Mobile nebo Symbian S60.

Alternativa od Sun Microsystems - JavaFX , uvedená ve stejném roce jako Silverlight, se řídí heslem „Write once, run anywhere“. To znamená, že aplikace naprogramované na platformě JavaFX lze spouštět ve kterémkoliv prostředí, kde běží JRE (Java Runtime Environment).

Použit by se jistě daly i jiné programovací jazyky, ale bylo by to mnohem náročnější na tvorbu a výsledek by byl nejistý.



## 6. Závěr

Jak bylo stanoveno v cílech, do této práce byl zaznamenán široký výčet možností této technologie, každá oblast možností Flashe by se dala zkoumat dopodrobna a jistě by bylo možné nalézt i další schopnosti, které ani nebyly vývojáři původně zamýšleny.

Autoři knihy *ActionScript 3.0 Cookbook* se zřejmě nemýlili, Flash skutečně dokáže naplnit veškeré představy autorů nejrůznějších projektů pro nejrůznější účely a s různými způsoby prezentace. Uživatel nemusí ActionScript aktivně znát, může využívat tzv. Code Snippets, a přesto dokáže vytvořit profesionálně vyhlížející projekt. Na tom je vidět, jak je program uživatelsky přívětivý a tvořit v něm může prakticky kdokoli i bez větších znalostí skriptovacích jazyků.

Tato práce dokazuje, že Flash si zaslouží své místo ve světě internetových prezentací, nejen svými možnostmi, ale i aktuálním rozšířením. Na druhé straně je pochopitelný konkurenční boj počítačových gigantů. Každý z nich by rád distribuoval vlastní technologii, proto je otázkou, jak se povede společnosti Adobe ve vyjednávání ohledně podpory ze strany producentů operačních systémů.

Závěrem by bylo dobré zmínit, že informacemi zveřejněnými na internetu se řídí stále více lidí, podle ČSOB dokonce až 60%, z toho vyplývá, že velké množství zákazníků lze získat na základě vhodné prezentace firmy na internetu. Předpokladem pro vhodnou technologii tvorby úspěšného webu, bude publikovatelnost informací, zobrazitelnost stránky na co největším množství zařízení a celková atraktivita. Díky možnostem textových polí, rozmanitosti v oblasti grafiky a designu a rozšíření příslušného přehrávače v počítačích se technologie Flash zdá být jedním z nejvhodnějších prostředků.

## 7. Seznam použitých zdrojů

1. SHUPE, Rich; ROSSER, Zevan. Learning ActionScript 3.0 : A begginer's guide to the flash language. first edition. Canada : O'Reilly, 2007. 384 s. ISBN 0-596-52787-X.
2. LOTT, Joey; SCHALL, Darron; PETERS, Keith. ActionScript 3.0 Cookbook. O'Reilly, 2006. 592 s. ISBN 0-596-52695-4.
3. Adobe Creative Team. Adobe Flash CS5 Professional. Oficiální výukový kurz. první vydání. Brno : Computer Press, 2010. 392 s. ISBN 978-80-251-3224-1.
4. Adobe Creative Team. Adobe Flash CS4 Professional. Oficiální výukový kurz, duben 2009. ISBN 978-80-251-2334-8.
5. Adobe. help.adobe.com. 16. 2. 2009 [cit. 2011-03-03]. Referenční příručka jazyka ActionScript 3.0 a jeho komponent. Dostupné z WWW: <[http://help.adobe.com/cs\\_CZ/AS3LCR/Flash\\_10.0/migration.html](http://help.adobe.com/cs_CZ/AS3LCR/Flash_10.0/migration.html)>.
6. CARR, Dan. Migrating from ActionScript 2 to ActionScript 3 : Key concepts and changes. 17.1.2011 [cit. 2011-03-03]. Dostupné z WWW: <[http://www.adobe.com/devnet/flash/articles/first\\_as3\\_application.html](http://www.adobe.com/devnet/flash/articles/first_as3_application.html)>.
7. GAY, Jonatan. The History of Flash. 14.7.2009 [cit. 2011-02-24]. Dostupné z WWW: <[http://www.adobe.com/macromedia/events/john\\_gay/index.html](http://www.adobe.com/macromedia/events/john_gay/index.html)>.
8. Flash Designer. 2009 [cit. 2011-03-10]. IFlashdesigner.co.uk. Dostupné z WWW: <<http://www.iflashdesigner.co.uk/>>.
9. HACHAMOVICH, Dean. Msdn.com. 29.4.2010 [cit. 2011-03-10]. IEBlog. Dostupné z WWW: <<http://blogs.msdn.com/b/ie/archive/2010/04/29/html5-video.aspx>>.

## **8. Přílohy**

Příloha 1: Vytvoření projekt (CD)