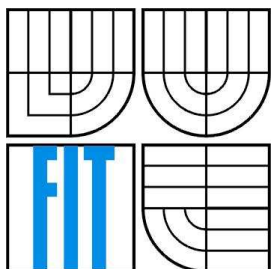


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ  
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY  
DEPARTMENT OF INFORMATION SYSTEMS

# PREZentační software matematiky pro ZŠ

PRESENTATION SOFTWARE FOR MATHEMATICS ON ELEMENTARY

SEMESTRÁLNÍ PRÁCE

TERM PROJECT

AUTOR PRÁCE

AUTHOR

Bc. Tomáš Páník

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Vít Štancl

BRNO 2007

# Prezentační software matematiky pro ZŠ

## Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Víta Štancla

Další informace mi poskytli...

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Jméno Příjmení  
Datum

## Poděkování

Tímto děkuji svému vedoucímu Ing. Vítu Štanclovi za poskytnuté zdroje a pomoc při vypracovávání práce.

© Tomáš Pánik, 2007.

*Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.*

## **Abstrakt**

V dnešní době je matematika jeden ze základních studijních oborů. Téměř neexistuje člověk který by se s ní nestkal. Studenti ovšem k ní často mají odpor a bojí se jí. Cílem této semestrální práce je návrh a analýza softwaru pro výuku matematiky studentů základních a středních škol. V práci je popsána základní motivace pro vytváření tohoto softwaru a principy fungování. Jedná se o software, který je určen výhradně pro samostudium a snaží se, aby se studenti matematiky nebáli a učení je bavilo.

## **Klíčová slova**

.NET Framework, matematika, C#, didaktika, pedagogika, e-learning, analýza

## **Abstract**

Nowadays, mathematics is an elementary study area. A person who has not encountered it is hard to find. However, students often loathe mathematics and are afraid of it. Goal of this term project is to design and analyze software for mathematics education for elementary school, middle school, and high school students. We describe the motivation behind the software and its functionality. The software will be used solely for self-study. We attempt to create software which would ensure that students are not afraid of mathematics and that they enjoy their studies.

## **Keywords**

.NET Framework, mathematics, C#, didactics, pedagogy, e-learning, analysis

## **Citace**

Tomáš Pánik: Prezentační software matematiky pro ZŠ. Brno, 2007, semestrální práce, FIT VUT v Brně

# Obsah

Obsah .....	1
1 Úvod.....	2
2 Metodika vyučování matematiky.....	2
2.1 Didaktika .....	2
2.2 Druhy vyučování .....	3
2.2.1 Transmittivní vyučování .....	3
2.2.2 Konstruktivní vyučování.....	3
2.2.3 Porovnání.....	3
2.2.4 Postupy vyučování.....	4
3 Výuka matematiky s využitím e-learningu .....	4
4 Návrh.....	5
4.1 Abstraktní model .....	5
4.1.1 Architect .....	7
4.1.2 Reader.....	8
4.2 Modularita.....	8
4.2.1 Plugin.....	9
4.3 Šablona výuky .....	12
4.4 Architect detailní specifikace.....	14
4.4.1 Okno kapitol .....	14
4.4.2 Okno objektů .....	15
4.4.3 Okno vlastností .....	15
4.4.4 Hlavní okno .....	15
4.5 Příklad fungování softwaru.....	16
5 Závěr .....	17
Literatura .....	18

# 1 Úvod

Tato práce se zabývá návrhem softwaru pro výuku matematiky. Tento návrh jsem se snažil udělat co nejrozsáhlejší a co nejvíce do detailů. Z vlastních zkušeností vím, že při implementaci vznikají různé změny, ale snažil jsem se, aby tyto změny později byly co nejmenší. Snažil jsem se pokrýt problematiku co možná nejvíce obecně a kladl jsem důraz na modulovatelnost tzn. aby jednotlivé bloky šli použít i v jiné implementaci. Nejdříve popíši metodiku výuky, základy didaktiky a elearningu a následně samotný návrh.

## 2 Metodika vyučování matematiky

### 2.1 Didaktika

Didaktika matematiky je vědecká disciplína zkoumající zákonitosti vyučování matematice v souladu s cíli vyučování určenými společností. Didaktika matematiky se zabývá studiem procesu vyučování matematiky počínaje dětmi předškolního věku až po studenty vysokých škol.

Problémy spadající do didaktiky matematiky jsou dvojího druhu:

- Problémy obsahu (Co učit?)
- Problémy vyučovacích metod (Jak učit?)

Hlavní podmínkou úspěchu je získat zájem žáků. Důležitá je osobnost učitele a jeho přístup k žákům. Obecně platí, jaký učitel taková škola. Učitel musí vzbuzovat důvěru.

Učitel zpracovává informaci obdrženou z osnov, vědecké, učební a metodické literatury, resp. obdrženou od důvěryhodných osob. Tuto informaci zpracuje podle úrovně myšlenkové činnosti žáka a předá jí žákovi.

Žák přijímá a zpracovává informaci obdrženou od učitele, z učebnice a z jiných zdrojů a na požádání učitele mu poskytuje informaci o kvalitě a osvojení si učební látky a dosaženém rozvoji myšlenkové činnosti.

Co je tedy cílem učitele matematiky? Zdá se, že je to: naučit se matematiku a přenést jí na žáka. Udělat tedy určitou transmisí neboli přenost. Odtud pochází i pojem **transmisivní vyučování**. Dále pak rozlišujeme pojem **didaktický konstruktivismus**.

## 2.2 Druhy vyučování

### 2.2.1 Transmisivní vyučování

Při transmisivním chápání vyučování matematiky se klade důraz na to, jak naučit žáka matematiku, tedy cílem je především samotná matematika, její školská forma, množství poznatků a především její kvalita. Žák v tomto vztahu není dominantní. Jako by při malování byla důležitější technika malby a výběr motivu, než sám samotný malíř. Učitel se hlavně podřizuje učivu a úloze zprostředkovatele.

### 2.2.2 Konstruktivní vyučování

Při konstruktivním vyučování matematiky se důraz klade především na žáka, přičemž matematika je chápána jako nenahraditelný nástroj na formování psychiky žáka a rozvoje jeho osobnosti prostřednictvím matematiky.

### 2.2.3 Porovnání

Představíme-li si konstruktivistické vyučování jako jeden pól spektra, na opačné straně mluvíme o transmisivním vyučování. (Transmise-přenos). Učitel v transmisivně vedené výuce se snaží předat žákům již hotové znalosti v dobré víře, že toto je nejlepší nejrychlejší cesta k poznání. Žák je viděn v roli pasivního příjemce a ukladatele vědomostí do paměti, aniž by se kladl důraz na jejich vzájemné propojení. (Podobné vyučování se také nazývá behavioristické nebo direktivní-přímé). V transmisivním pojetí jako by vyučování bylo podobné přidávání zboží (znalostí) do skladu (žákovy mysli), kde příliš nezáleží, co už je v sousedních odděleních skladiště.

Pozor-transmisivní vyučování nelze zatratit, může vyučovací proces vhodně doplňovat.

		Konstruktivistické vyučování	Transmisivní vyučování
1	hodnota poznání	Kvalita	Kvantita
2	motivace	Vnitřní	Vnější
3	trvanlivost poznání	Dlouhodobá	Krátkodobá
4	vztah učitel žák	Partnerský	Submisivní (oddaný)
5	klima	Důvěry	Strachu
6	nositel aktivity	Žák	Učitel
7	činnost žáka	Tvořivá	Imitativní

8	poznatek žáka	Produktivní	Reproduktivní
9	nosná otázka	Co? a Proč?	Jak?

## 2.2.4 Postupy vyučování

V současné době se v didaktice matematiky uvádějí čtyři postupy k výuce matematiky:

1. **Mechanický** - Mechanický přístup k učení odpovídá chápání učení jako systému reakcí. Žáka můžeme, podobně jako počítač, naprogramovat pomocí drilu k provádění aritmetických, algebraických a geometrických operací a řešení problémů, které lze podle určitých znaků klasifikovat a dále řešit podle určitých vzorců.
2. **Strukturalistický** – Strukturalistický přístup k učení lze doložit dvěma příklady: tradiční geometrií uspořádanou na základě axiomatické konstrukce a tzv. moderní matematikou založenou na teorii množin a logice. Pro žáky byl vytvořen strukturovaný svět množin a relací.
3. **Empirický** – Empirie je v překladu zkušenost. Empirický značí-vycházející ze zkušeností. Empirický přístup vychází z potřeb praxe a potřebám praxe má sloužit. Při vyučování jsou využívány zkušenosti žáků, nejsou však vedeni k systematickému a racionálnímu zpracování těchto zkušeností.
4. **Realistický** – Opět se vychází z reálných podnětů, z neustále se rozšiřujícího žákova světa. Žák se stává znovuobjevitelem matematiky, což podněcuje a rozvíjí jeho schopnosti. Realistické vyučování matematiky vychází z principu, že učení matematiky znamená konstruování matematiky vlastními postupy žáků od neformálních přístupů spjatých s realitou k něčemu, co je přijatelné jako formální matematika.

Nelze podceňovat nebo ztracovat ani jeden z těchto přístupů. Je na učiteli, aby našel vhodnou míru přístupu.

## 3 Výuka matematiky s využitím e-learningu

Jednou možností jak přiblížit matematiku studentovi je využití potenciálů výpočetní techniky. Studium s pomocí počítačové techniky bývá pro studenta atraktivnější nežli klasické metody a navíc s sebou přináší i další pozitivita.

Pod pojmem e-learning si však nesmíme představovat pouze studijní materiály umístěné na internetu, hlavní podstatou je práce s těmito materiály. Jedním z předpokladů je však i vzájemná komunikace mezi pedagogy a studenty. Ovšem při výuce matematiky s elektronickou podporou se pedagogové většinou staví skepticky. Mají názor že je nutný tzv. „face to face“ čili přímý kontakt studenta s učitelem, který nelze nahradit žádnou elektronickou učební pomůckou. Je pravda že výuka matematiky elektronickou formou vyžaduje náročnější přístup než některé klasické výkladové předměty, ale ukázalo se, že právě moderní elektronické nástroje umožní výuku matematiky zpestřit a více ji přiblížit studentům než jen pouze suchopárné knížky či učebnice. Ukazuje se že nové technologie pomáhají lépe přiblížit matematiku studentům, a to z těchto důvodů:

- **Individuální tempo výuky** - díky elektronické podpoře může student individualizovat tempo studia i rozsah látky, což umožňuje po určitém čase srovnat znalosti studentů s různými startovacími znalostmi.
- **Atraktivnost studia** - již samotné využití počítačů v výuce přidává na atraktivnosti. Navíc oproti knížkám resp. tabuli přináší další možnosti, jako jsou multimediální záznamy (videa, animace, simulace, zvukový popis apod.)
- **Průběžná kontrola studia** - vhodně zvolené testy umožňují, aby si student sám ověřil, do jaké míry probranou látku pochopil. Při těchto testech navíc není stresován prostředím, přítomností pedagoga ani hrozbou sankcí v případě neúspěchu.

Troufám si tvrdit, že studenti kteří studují s elektronickou podporou, mají o studiu větší zájem než studenti, kteří studují klasicky. Nesměřují tolik k mechanickému učení látky („biflování“) a raději se snaží probranou látku lépe pochopit. Lépe taky překonají obavu a ostych, které studium matematiky provází, a není výjimkou že se počáteční nedůvěra změní na pozitivní vztah k předmětu.

## 4 Návrh

### 4.1 Abstraktní model

Jedná se o software pro výuku matematiky, který je určen primárně pro samostudium. Není určen například pro přímou výuku ve školních hodinách ani není řešená žádná síťová komunikace. Základní cíl softwaru je aby učitelé měli možnost vytvořit si vlastní plán výuky a všechno co k němu patří. Základní podporované vlastnosti jsou:

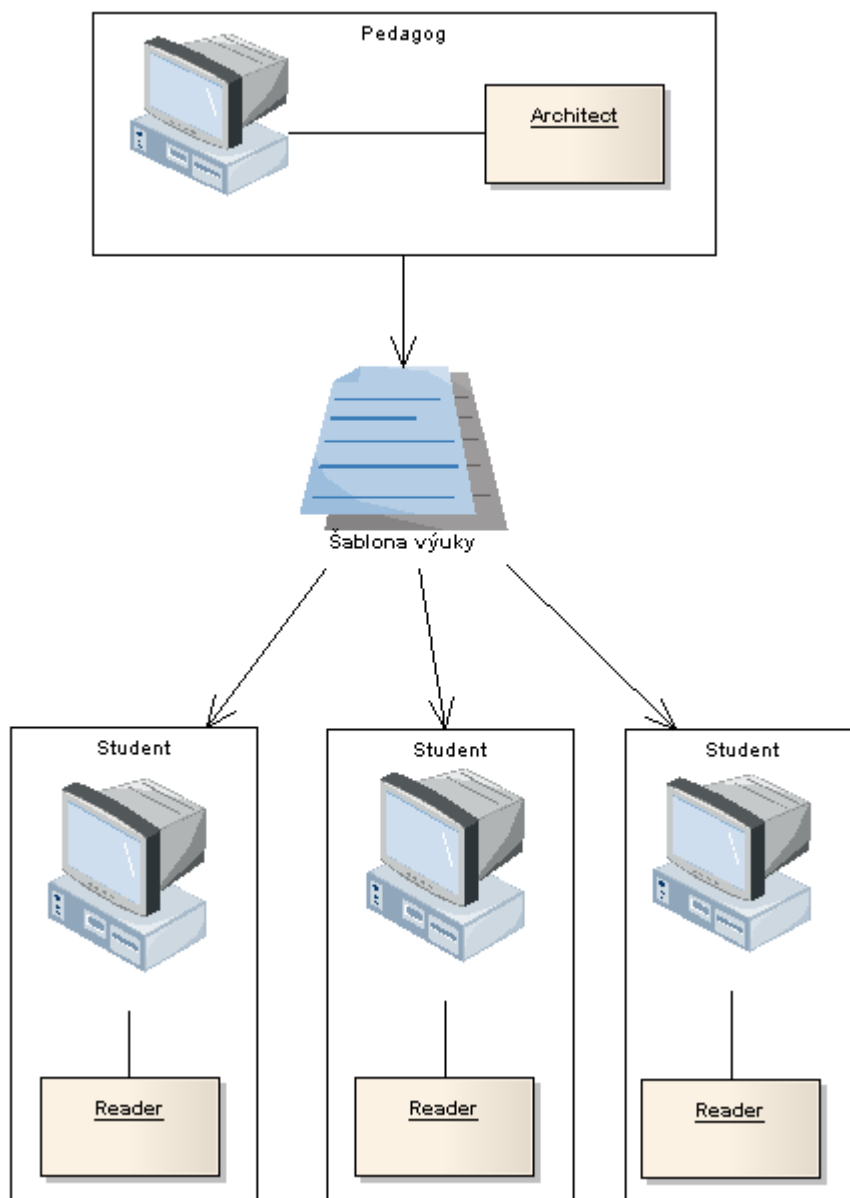


- **Výklad** - program studentovi podrobně popisuje vybranou látku, včetně názorných ukázek a příklady postupů řešení
- **Test** - student si sám vyzkouší řešit příklady, kde na výběr má určitý počet odpovědí. Na konci mu program test vyhodnotí a příslušně oznámkuje

Podrobněji je postup výuky popsán dále.

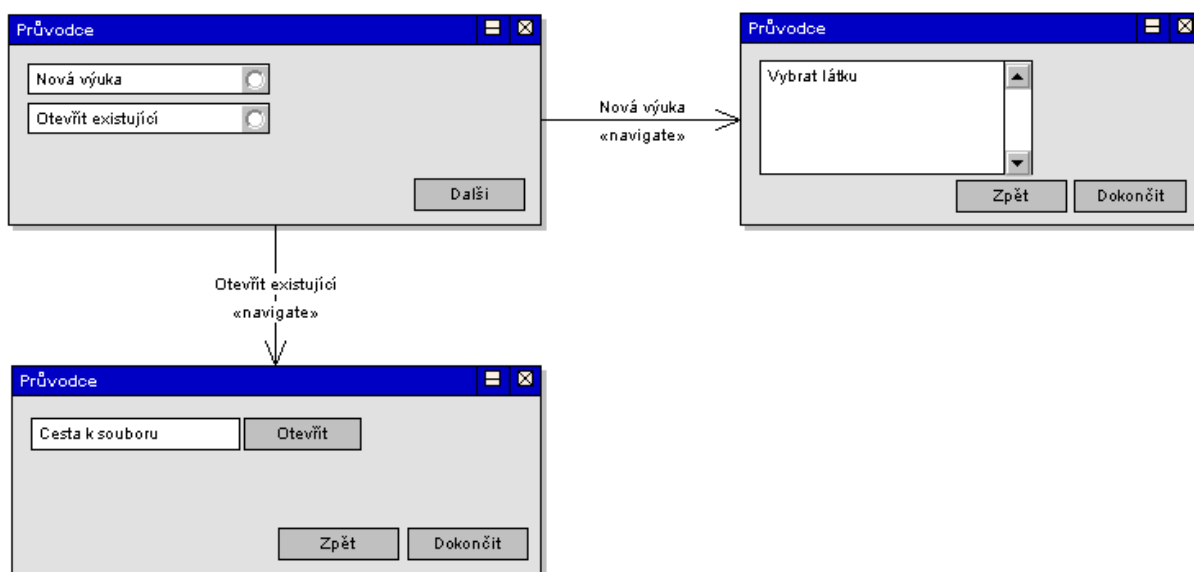
Software je rozdělen na 2 samostatné části:

- **Architect** - Tímto programem se výuka vytváří. Určeno pro pedagogy
- **Reader** - Interpretuje samotnou výuku z vytvořené šablony vytvořené pomocí Architect modulem.

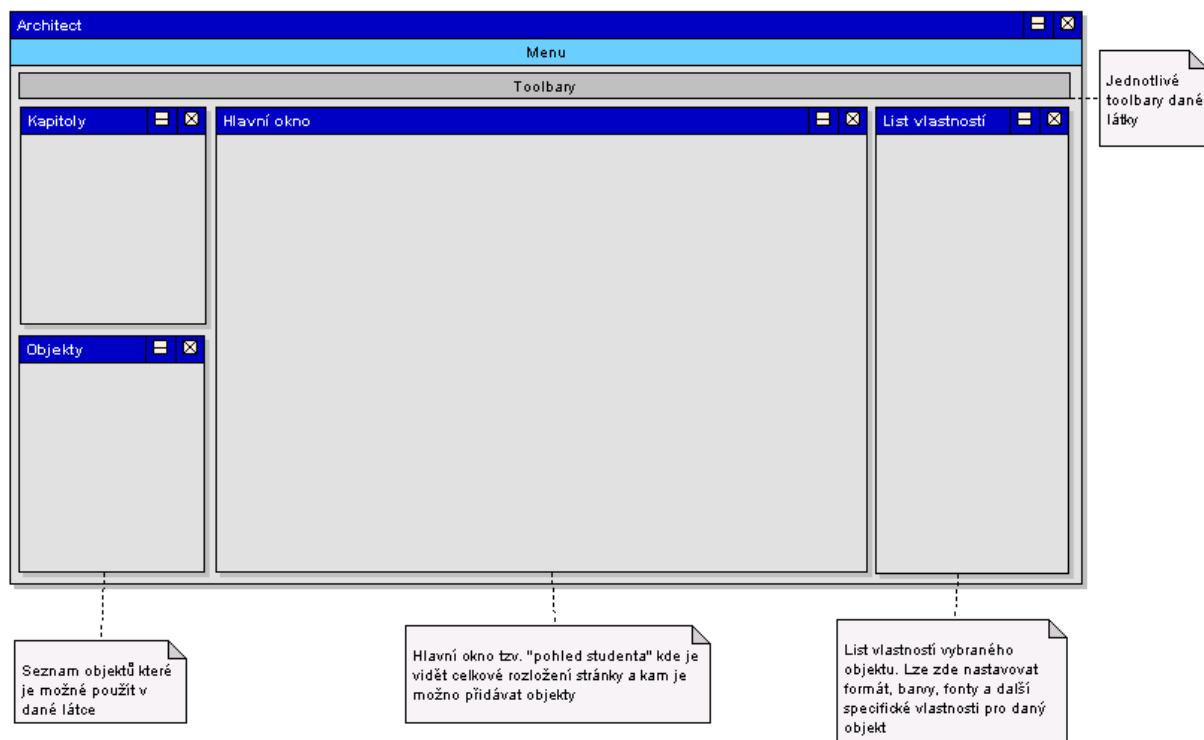


## 4.1.1 Architect

Základní stavební kámen softwaru. Zde učitel vytvoří výuku pomocí prostředků které má software k dispozici. Při samotném spuštění se zobrazí úvodní průvodce, pomocí kterého se vybere zda se má založit tvorba výuky nové, či otevřít stávající. Při výběru tvorby nové se následně dotáže na látku (pluginy) které budou ve výuce použity. To se bude dát rozšířit i později, ale určují se podle toho zobrazené toolbary, objekty a menu.



Při stisknutí tlačítka dokončit se otevře hlavní editor výuky. Je rozdělen na 4 části. V levé části je stromový seznam kapitol a stromový seznam objektů, uprostřed hlavní okno a vpravo list vlastností vybraného objektu. Detailní specifikace těchto částí bude popsána dále.



## 4.1.2 Reader

Je to jakýsi interpret, který umožní studentovi projít vytvořenou výuku. Jeho základní vlastnosti:

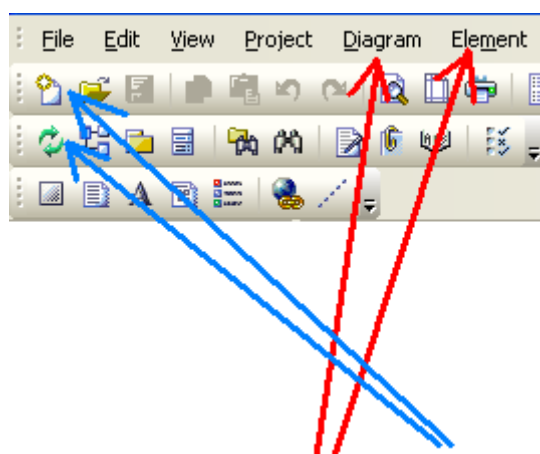
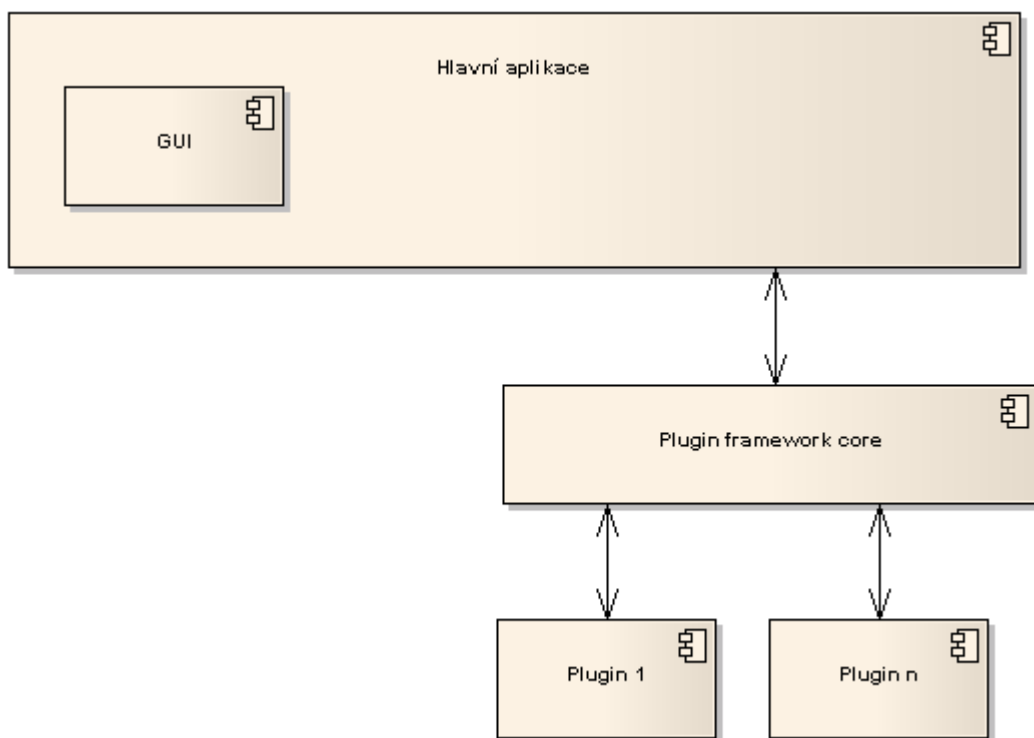
- **Fullscreen** - možnost pracovat ve fullscreen módu, tzn. zatemnit krajové rušivé elementy.
- **Výběr kapitoly** - výběr ze seznamu kapitol, nebo testů
- **Znovu přehrání přiložených multimédií** - jako zvuků, videí apod. pokud jsou přiloženy

## 4.2 Modularita

Základní požadavek na software je jeho plná modulovatelnost. Jak již bylo řečeno dříve v software si lze vybrat z nějaké látky a podle ní je možné používat příslušné objekty.

Pro tento účel byl vyvinut framework pro řízení pluginů. Díky němu aplikace obsahuje pouze základní potřebné operace a funkce, jako je stylování a formátování textu, práce s multimediálními daty a další. Přidáním modulu, např. pro výuku geometrie či logaritmů, bude moci s touto látkou plně pracovat.

Framework byl navržen tak aby ho bylo možné rychle přilinkovat k libovolnému jinému projektu. Jeho návrh je vidět na obrázku níže.



Pluginy přidané **menu** a **toolbar**

## 4.2.1 Plugin

Pro konkrétní vysvětlení fungování pluginů musíme znát pojem *reflexe*.

Reflexe je jedna z vlastností platformy .NET a čím větší aplikaci vyvíjíme tím víc je tato vlastnost důležitá. Ve stručnosti je to silný způsob sbírání a manipulování informací shromážděných v aplikacích a jejich metadat. Metadata obsahují všechny typové informace použité v aplikaci. Schopnost získat tyto informace „za běhu“ to dělá ještě více prospěšné. Reflexe může být použita pro efektivní hledání všech druhů typů nebo dynamicky volat metody podle jejich názvu. Můžeme také vytvářet instance typů, svázat typ k existujícímu objektu, nebo získat typ z existujícího objektu a poté

volat jeho metody nebo zpřístupnit jeho pole a vlastnosti. Právě díky této vlastnosti je možné jednoduše v prostředí .NET plugin framework.

Každý plugin obsahuje XML soubor pro výměnu informací s menu, toolbary a objekty. Struktura je následující:

```
<FrameworkAddin>
  <AppVer>1</AppVer>
  <AddinName>Jméno pluginu</AddinName>
  <ToolBarButtonCount>Počet tlačítek toolbaru</ToolBarButtonCount>
  <MainMenu>
    <Name>Název menu</Name>
    <ShortcutKeyIndex>1</ShortcutKeyIndex>
    <SubMenu>
      <Name>Název podmenu</Name>
      <ShortcutKeyIndex>2</ShortcutKeyIndex>
      <LeafMenu>
        <Name>Název spustitelného menu</Name>
        <FunctionName>Název funkce v pluginu</FunctionName>
        <HelpString>Nápověda</HelpString>
        <ToolTip>Tooltip nápověda</ToolTip>
        <ToolBarIndex>Addin1.ico</ToolBarIndex>
        <ShortCutKey>Ctrl+B</ShortCutKey>
      </LeafMenu>
      <LeafMenu>
        .....
      </LeafMenu>
    </SubMenu>
    <SubMenu>
      .....
      .....
    </SubMenu>
  </MainMenu>
  <MainMenu>
    .....
    .....
  </MainMenu>
  <Objects>
    <Object>
      <Name>Název</Name>
      .....
    </Object>
  </Objects>
</FrameworkAddin>
```

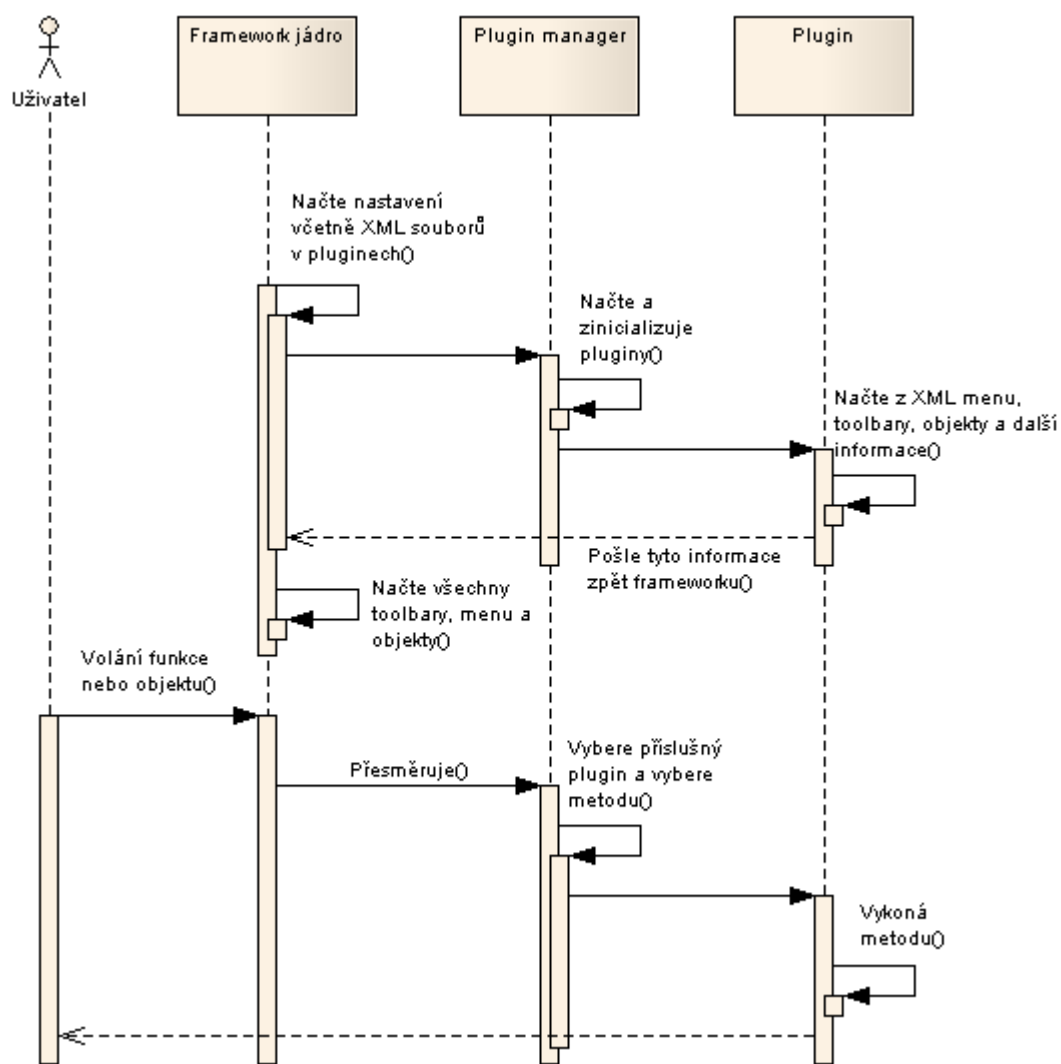
```

        .....
    </Object>
    <Object>
        .....
        .....
    </Object>

</Objects>
</FrameworkAddin>

```

Sekvenční diagram popisující činnost frameworku



#### 4.2.1.1 Některé pluginy

- **Plugin animací** - Někdy je potřeba zobrazovat data postupně. Výklad je většinou lepší prezentovat touto formou, a ne na ně „vybalit“ najednou všechna data. Nastaví se bloky, které se budou zobrazovat a čas a plugin animací se už o všechno ostatní postará. Pro studenta to pak bude vypadat jak prezentace vystřižená z PowerPointu. A to je účel tohoto pluginu.
- **Editor rovnic** - téměř v každém oboru matematiky je potřeba nějakým způsobem zapisovat rovnice a vzorce. Tzn. je potřeba, aby bylo možné zadat do listů rovnice, které bude následně editovat. To jsou objekty které když se editují, tak vyskočí formulář kde je možné tyto vzorce zadávat. Jisté obdoby používají i MSWord a OpenOffice, tento je trochu zjednodušený a určitě je ještě spousta prostoru na rozšíření.

## 4.3 Šablona výuky

Je předem jasné, že souborů výuky bude víc. Jedná se o multimediální soubory, jako je zvuk, obrázky, video ale taky samotná výuka, která bude mít formát XML, ale o tom níže. Soubor s výukou který je předáván studentovi musí být kompaktní. Je sice možné mu předat složku se soubory ale výrazně to zvyšuje možnost že některá data budou chybět a snižuje to bezpečnost. Pro tento účel nebudu vymýšlet speciální formát pro data. Uchýlil jsem se jednoduchou ale velice efektivní cestou. Všechny soubory budou zabaleny pomocí zip. Díky tomu se soubor i znatelně zmenší a není problém do šablony ukládat tzv. hrubá data, jako obrázky ve formátu BMP nebo XML soubory.

Adresářová struktura je následující:

```

├── [multimedia]
│   ├── [sound]    -- všechny zvukové soubory
│   ├── [video]   -- všechny video soubory
│   └── [pictures] -- všechny obrázky
└── template.xml

```

Hodně důležitý je pak formát souboru template.xml. Byl zvolen následovně

```

<Document>
  <Appver></Appver>
  <Name>Jméno dokumentu</Name>
  <Plugins> --potřebné pluginy
    <Plugin>
      <Name>Jméno pluginu</Name>
      <Config>

```

```

Volitelná konfigurace pluginu
.....
.....
</Config>
</Plugin>
.....
.....
</Plugins>
<Body>
Grafický styl dokumentu, zde je barva pozadí, defaultní fonty atd..
.....
.....
</Body>
<Sections>
  <Section Name="Jméno sekce" Type=" Typ sekce (Výuka, Test)">
    <Chapters>
      <Chapter Name="Jméno kapitoly">
        <Slides>
          <Slide Name="Jméno stránky (slouží pouze jako identifikátor)">
            <Objects>
              <Object>
                Vlastnosti a nastavení objektu (styly, fonty, barva,
                pozice, atd.)
              </Object>
              .....
              .....
            </Objects>
            <Sequence>
              Definice sekvence zobrazování objektu

            </Sequence>
          </Slide>
          .....
          .....
        </Slides>
        <Questions> --testové otázky (pouze pro section typu Test)
          <Question>
            <Answers>
              <Answer>
            </Answer>
            .....

```



```

.....
    <Answers>
  </Question>
.....
.....
  </Questions>
</Chapter>
.....
.....
  </Chapters>
</Section>
.....
.....
</Sections>
</Document>

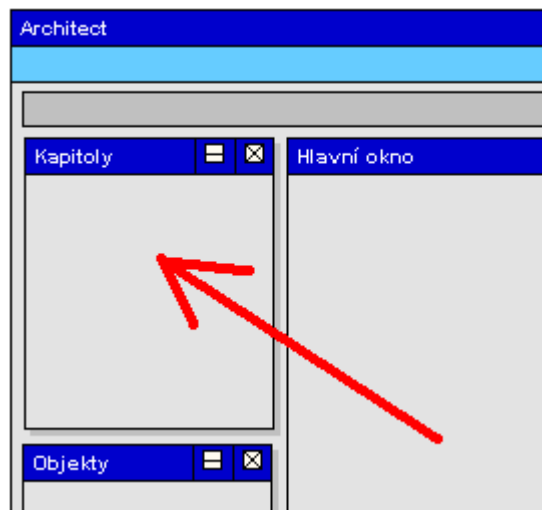
```

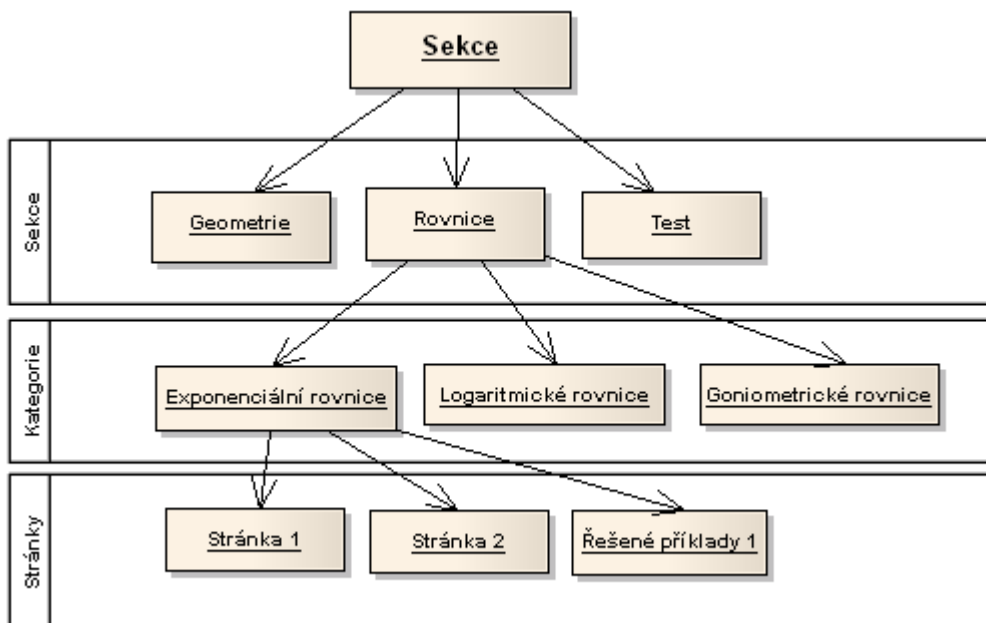
## 4.4 Architect detailní specifikace

Zde si popíšeme jednotlivé části modulu architect, jejich detailnější popis a specifikace.

### 4.4.1 Okno kapitol

Nachází se na levé straně a obsahuje stromovou strukturu jednotlivých sekcí, kapitol a stránek. Jak je vidět ve formátu šablony, výuka se v první fázi rozděluje na sekce. Sekce můžou být 2 a více typů. Typ **výklad** a typ **test**. Sekce test je pouze sbírka otázek a odpovědí k procvičování. Výklad je již složitější. Rozděluje se na kapitoly a každá tato kapitola ještě na stránky. Příklad použití těchto sekcí je znázorněn na obrázku.





#### 4.4.2 Okno objektů

Okno objektů je seznam všech objektů, které je možné použít v aktuálním listu. Každý list bude moci používat jiné, takže při přepnutí listu v okně s kapitoly se změní i tento seznam. Objekty je možné vkládat do hlavního okna pomocí tahem myši, neboli „Drag and Drop“.



#### 4.4.3 Okno vlastností

Vpravo je okno s listem vlastností objektu. V tomto okně lze nastavovat vlastnosti objektů. Jako je font, barva a další vzhledové vlastnosti, ale i další speciální vlastnosti podle vybraného objektu.

#### 4.4.4 Hlavní okno

Toto okno je vlastní jádro aplikace. Tak jak jsou zde data zobrazeny, by měl vidět i student při výuce. Přiblížím zde jen základní prvky tohoto okna. Zobrazuje se zde vždy aktuálně vybraný list v oknu kategorií. Tento list je zde možné editovat. Na spodní části jsou tlačítka na přechod mezi listy, tak jak jimi bude listovat student. Ještě pod touto lištou bude další lišta, kterou student už nevidí. Zde se budou nacházet speciální objekty, které budou neviditelné a budou určovat chování daného listu, jako např. multimedia(komentář..), časovač na animace a další.

## 4.5 Příklad fungování softwaru

Popíše zde stručně jak taková výuka proběhne.

Profesor se rozhodne, že se začne probírat nová látka geometrie. Spustí si architekta, vytvoří novou šablonu výuky s názvem „lekce 21“. Vytvoří 2 sekce, „geometrie“ a „test“. Sekce geometrie bude klasický výklad nové látky. Zde vytvoří 5 kapitol kde v každé se bude vysvětlovat jiná látka. Jedna bude nazvaná „konstrukce trojúhelníku“, další „průsečíky“ a další. V každé z kapitol udělá několik stránek, kde látku doslovně popíše a ke každé stránce namluví i mluvní popis. Takto vytvořenou šablonu předá studentům.

Student dostane tuto látku jako přílohu k právě probrané látce ve škole. Spustí si modul reader a načte šablonu. Zvolí si sekci „geometrie“ a zvolí první kapitolu. Spustí si slovní audio popis a bude si postupně látku pročítat. Po projití probrané látky si spustí test kde si látku vyzkouší. V případě nejasností si potřebné stránky může projít znovu, nebo se v příští vyučovací hodině zeptá profesora.

## **5 Závěr**

Je spousta výukových softwarů v oblasti matematiky, vytvářejí je ale specializované společnosti a rozšiřování takových materiálů lektory, kteří jsou přímo v kontaktu se studenty, je problematický. V tomto projektu jsem se pokusil určitým způsobem vyřešit tuto problematiku.

# Literatura

- [1] E-Matematika : *články matematiky* [online] <http://www.e-matematika.cz>
- [2] Wikipedia : *e-learning* [online] <http://cs.wikipedia.org/wiki/ELearning>
- [3] Hejný M., Kuřina F: *Konstruktivní přístupy k vyučování matematice*, Matematika, Fyzika, Informatika, 1998.
- [4] eucebnice.cz : *osnovy matematiky* [online] <http://www.eucebnice.cz/matematika/osnovy2.htm>