

UNIVERZÁLNÍ TESTOVACÍ PROSTŘEDÍ

Petr Voborník

Abstrakt

V příspěvku představíme nově vyvíjený on-line nástroj pro vytváření, provoz a správu testů. Ty mohou být provozovány nejen v režimu plně zabezpečeného hodnoceného zkoušení, ale i jako efektivní auto testy. Nástroj je vytvářen v technologii Silverlight, která dovoluje použití bohatých a komfortních prvků pro interakci s uživatelem, jako je např. metoda drag&drop a další, které klasické HTML rozhraní neumožňuje. Díky podpoře API mohou být tyto testy integrovány nejen do různých LMS, ale i do libovolné webové aplikace. Pro vytváření testových otázek byl navržen i nový jazyk na bázi XML, který autorům testů dovoluje plně uplatnit svůj tvůrčí potenciál, včetně podpory sofistikovaných elementů pro generování náhodných prvků.

Klíčová slova

testování, online, silverlight, API, drag&drop

1. Úvod

Testovacích systémů existuje velmi mnoho (viz [4]), avšak jen mizivé procento z nich využívá bohatých možností, které nabízí současné technologie. Mezi stále preferovanější aplikační architekturu patří *cloud computing* a tzv. RIA¹. Tyto aplikace zahrnují veškeré výhody webových aplikací (není nutné je instalovat, potřeba je pouze webový prohlížeč, jsou dostupné odkudkoli, mají centralizované datové úložiště apod.) ovšem dosahují komfortu aplikací desktopových (rychlá odezva, použití pokročilejších programových technologií, částečná offline funkčnost apod.). Nové trendy v oblasti uživatelských aplikací také směřují k propracovanější grafice, plynulosti ovládání, interaktivitě a multimediálním prvkům. Těchto zásad se drží i zde popisované univerzální testovací prostředí.

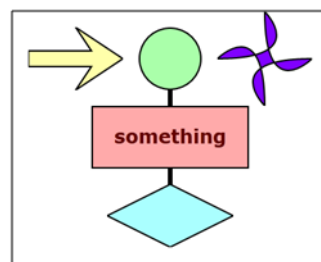
2. Jazyk pro tvorbu otázek

Jak je u testů běžné, skládají se z jednotlivých otázek. Zde je pak ke každé z nich přístupováno jako k samostatnému dílu. Aby jejich autorům bylo umožněno plně využít svůj tvůrčí potenciál a neomezovat se pouze na několik předdefinovaných typů otázek, byl pro jejich zápis vytvořen vlastní jazyk, pracovně nazývaný **QML** (Questions Markup Language), založený na klasickém XML. Ten dovoluje do otázky zapracovat libovolnou grafiku, animace, aktivní prvky i náhodně generované objekty. Každá otázka je tak samostatným grafickým i funkčním celkem, jež při vhodné aplikaci náhodných prvků může pokrývat i celý tematický okruh učiva.

¹ RIA –Ritch Internet Application – bohatá internetová aplikace

2.1. Grafika

Pro tvorbu grafického vzhledu otázky byl v QML vytvořen mini-jazyk (QML-graphics), který umožňuje používat základní grafické objekty a to jak vektorové (čáry, elipsy, křivky apod., viz Obr. 1) tak i rastrové (obrázky). Inspirací pro něho byly již existující standardy SVG a XAML, z nichž bylo převzato to nejlepší. Možné je také použít lineární a radiální dvou i vícebarevné barevné přechody (viz Obr. 5).



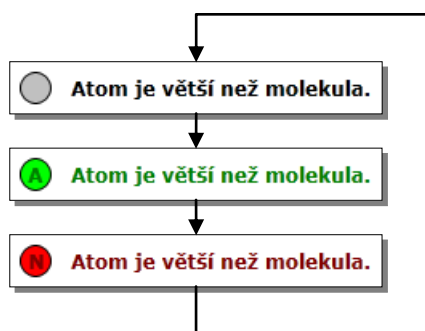
Obr. 1: Ukázka grafiky

2.2. Transformace a animace

Transformace umožňují objektům i jejich skupinám, umístěným na pozicích prvcích, změnu jejich měřítka nebo otočení. Pouze díky nim lze vytvořit např. pootočenou elipsu. Animace jsou pak jednou z výhod, kterými papírový test nikdy disponovat nemůže, i proto v QML nechybí jejich podpora. Aktuálně je možné animovat měřítko, otočení, pohyb, průhlednost a barvy.

2.3. Aktivní prvky

Aktivní prvky poskytují zkušenému interaktivitu, tedy možnost do předložené otázky zasáhnout, něco vykonat a tím na ni odpovědět.



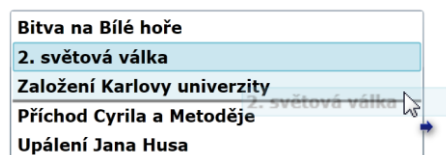
Obr. 2: Ukázka přechodů třístavového přepínače

Jednou z nejobvyklejších typů testových otázek je otázka s výběrem jedné či více z nabízených odpovědí. Volbu předpokládané odpovědi zkušený obvykle provádí jejím vyznačením (zaškrtnutím, zakroužkováním, podtržením apod.). Pro tyto účely zde slouží **přepínače**. Ty umožňují definovat takové grafické objekty, které se po kliknutí na ně změny v jiné (např. prázdný kružek v zaškrtnutý). Původní myšlenku však překračují, neboť mohou mít v podstatě neomezené množství stavů (tedy nejen dva), přičemž grafický obsah každého

z nich, kromě vymezené polohy a rozměrů, není nijak závislý na ostatních. Lze tak například snadno vytvořit sérii výroků, u kterých má zkušený rozhodnout, jsou-li pravdivé či nikoli, včetně možnosti „neodpovídat“, tj. třemi stavy (viz Obr. 2).

Otázka, nebo její část také může po zkušeném vyžadovat, aby seřadil nějaké položky do správného pořadí („**uspořádací úloha**“ dle [1 str. 49]). To je zde realizováno metodou drag&drop, kdy se jednotlivé položky mezi sebou přetahují tak, že se některá z nich umístí mezi jiné dvě a vše se okamžitě posune tak, aby nikde nezbyla mezera. Těchto řadících jednotek může být v otázce i více a lze je nastavit tak, že položky bude možné přetahovat i mezi nimi navzájem.

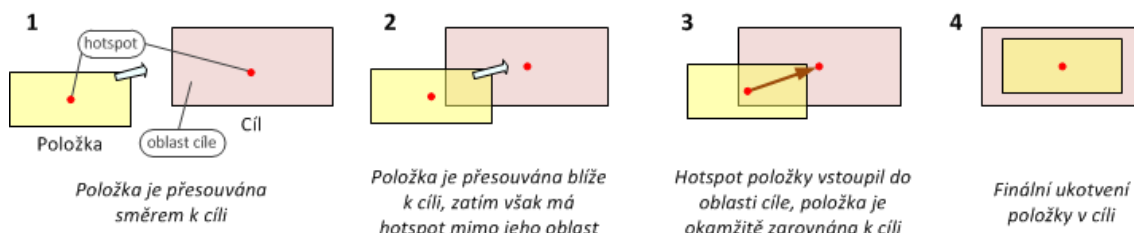
Seřad'te vzestupně historické události podle data, kdy k nim došlo.



Obr. 3: Ukázka uspořádací úlohy

Součástí otázky mohou být i **položky a cíle**. Jedná se o velmi efektivní typ úlohy, který se v klasickém HTML realizuje jen velice těžko. V této podúloze jsou samostatně a nezávisle na sobě stanoveny cíle a položky. Položky jsou plovoucí objekty s vlastní grafikou, které lze myší přesouvat po celé ploše otázky metodou drag&drop. Cíle jsou stanovené oblasti, do kterých se mají položky umísťovat.

Cíle zároveň mají tu vlastnost, že pokud se do jejich vymezené blízkosti během tažení dostane položka, mohou ji „přitáhnout“ k sobě a ta tak jakoby zapadne na své místo (viz Obr. 4).

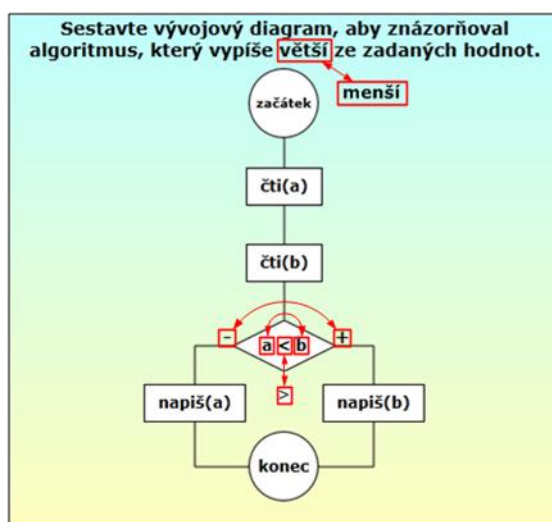


Obr. 4: Postup umísťování položek do cílů

Každá položka přitom zapadá do každého cíle stejně, takže pouhým zkoušením přítomnosti této „magnetizace“ různých položek v různých cílech nelze odhadnout, která položka kam patří. Pokud je cíl obsazen, další položky již nepřitahuje. Ukotvenou položku lze z cíle uvolnit „tažením“ myší mimo jeho oblast.

2.4. Náhodné prvky

QML podporuje také elementy pro generování náhodných čísel, znaků, slov a barev. Mimo to i určité části kódu mohou být podmíněny pro jeho použití při testu náhodným výběrem z několika variant. Kód může, podobně jako je tomu v XSLT, používat podmínky, větvení, cykly a interní šablony. Dokáže též vyhodnocovat matematické i logické výrazy. Jedna otázka tak může mít obrovské množství variant, což umožňuje její opakované použití, aniž by jí hrozilo „opotřebení“ (mechanické odpovídání typu „správná odpověď je B“ apod.).



Obr. 5: Ukázka řešení otázky s vyznačením náhodně se měnících částí

2.5. Automatické vyhodnocování

Úspěšnost každé otázky je hodnocena procentuálně, resp. v intervalu $<0, 1>$. V otázce však může být mnoho různých možností, jak toto skóre zvýšit či snížit a bylo by velmi složité, v některých případech dokonce nemožné, definovat jednotlivým případům takový podíl na skóre, aby v součtu všech nejlépe hodnocených voleb dával právě 1 (100%). Z tohoto důvodu je uvnitř každé otázky stanoven bodový subsystém, kdy se jednotlivým případům přidělují určité bodové

váhy. Jejich rozsah a stupnice může být v každé otázce zvolena libovolně. Systém totiž při generování otázky sám vypočítá maximální možný bodový zisk a jím dělí počet bodů získaných, čímž je výsledné skóre převedeno zpět do jednotného rozsahu $<0, 1>$. Obdobně funguje i celkové hodnocení testu, kdy je výsledek váženým průměrem výsledků jednotlivých otázek.

2.6. Zpětná vazba

Díky automatickému vyhodnocování systém dokáže ihned poznat, je-li odpověď zkoušeného zcela správná, zcela chybná, případně správná jen z části a z jaké. Na základě tohoto rozdělení nejen přidělí procentuální hodnocení, ale i může v rámci zpětné vazby vyznačit jednotlivé části (aktivní prvky) v otázce dle tohoto stavu.

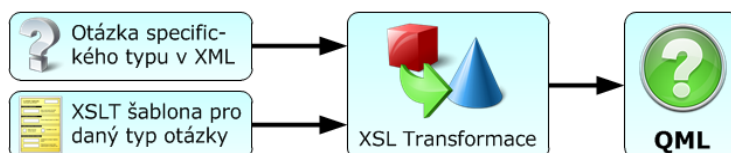
Další možností při následném rozboru testu je kromě vyznačení chybných a správných odpovědí v každé otázce i funkce, která zkoušenému vypíše, jak to mělo být správně, což dokáže systém generovat automaticky, a proč, což již samozřejmě musí zadat tvůrce otázky.



Obr. 6: Ukázka automaticky generované zpětné vazby

2.7. Transformace pomocí XSLT šablon

Způsob zápisu QML umožňuje vytvářet graficky efektní a přitom funkčně rozsáhlé otázky, ovšem daní za tuto univerzálnost je někdy až zbytečně zdoluhavý QML kód. Aby stejný kód u opakujícího se typu otázek nebylo nutné psát (kopírovat) neustále znovu, byla přidána možnost definovat si libovolnou transformační šablonu (XSLT¹), která ze stručného zápisu v několika málo triviálních XML elementech vygeneruje propracovanou a plně funkční otázku v QML.



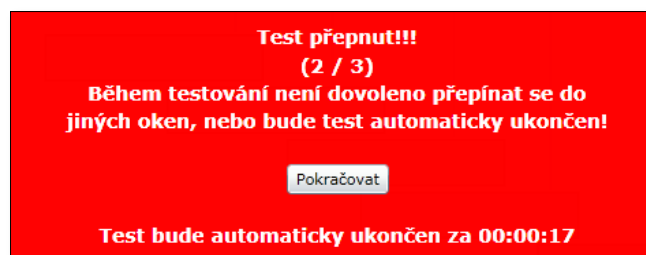
Obr. 7: Schéma transformace zjednodušeného XML zápisu otázky do QML pomocí XSLT

3. Rozhraní aplikace

Zcela funkční rozhraní zatím bylo vytvořeno pouze pro testovací část. Obsahuje „stránky“ pro registraci, přihlášení, přehled organizací, přehled testů i s podpřehledy dosažených výsledků, testovací část (viz Obr. 9) a rozbor testu.

Během testování je také k dispozici ochrana proti opisování z internetu. Je-li totiž okno prohlížeče nebo jeho aktuální záložka během testování přepnuta, test je překryt červeným panelem a je zahájen odpočet jeho automatického ukončení, které lze zastavit pouze kliknutím na tlačítko „Pokračovat“ (viz Obr. 8). Čas pro tento odpočet i počet tolerovaných přepnutí lze nastavit pro každý test zvlášť.

¹ XSLT (eXtensible Stylesheet Language Transformations) je jazyk založený na XML pro transformování XML dokumentů do jiných textových formátů. Nejčastěji se XSLT používá pro transformaci jednoho typu XML dokumentu do jiného typu XML dokumentu, což pomáhá zmírnit nekompatibilitu. [2 str. 85]



Obr. 8: Ochrana proti opisování z internetu či jiných aplikací

Co se týče komunikace během testování, nezbytné je stáhnout data pouze při zahájení testu a odeslat výsledek po jeho ukončení. Testování je tedy odolné proti případným výpadkům spojení v jeho průběhu. Při nepovedeném odeslání výsledků jej lze opakovat, dokud se nezdaří, případně výsledky uložit do zašifrovaného souboru a z něj je do systému vložit později ručně. [4]



Obr. 9: Ukázka testovací části rozhraní

3.1. API

Aplikace kromě přímého uživatelského použití disponuje i rozhraním API, tedy možnosti jejího částečného ovládnutí prostřednictvím aplikací třetích stran. Tyto aplikace tak mohou například pomocí parametrizovaných URL spouštět přímo konkrétní testy pro předdefinované uživatele a po jejich ukončení přijímat výsledky z těchto testů přes návratovou či skrytou URL, JavaScriptem nebo přes administrační API. Veškerá komunikace mezi aplikacemi samozřejmě podléhá přísnému zabezpečení proti neoprávněným zásahům, s využitím jednorázově platných podpisů obou stran (více o této metodě např. viz [3]).

Testy tak mohou být spouštěny nejen přes integrované rozhraní, ale i přímo přes odkazy či tlačítka jiné webové aplikace a to jak na samostatné stránce prohlížeče, tak i v rámci (iFrame) obsahu jiné stránky. Testy tak lze přímo propojit s obsahem nejen libovolného LMS, ale i kterékoli jiné aplikace. Pro jednodušší integraci do LMS se pak připravuje SCORM rozhraní.

Na server aplikace přistupující k testům přitom není třeba instalovat jakýkoli další software, k testovacímu prostředí se přistupuje výhradně jejím voláním na centrálním serveru, podobně jako fungují video objekty na YouTube.

4. Závěr

Představené univerzální testovací prostředí používá vlastní jazyk pro tvorbu otázek, umožňující jak jejich bohaté grafické a interaktivní zpracování, tak i širokou variabilitu pro vnesení náhodné složky přímo do jednotlivých otázek. Ty díky tomu mohou být používány opakovaně bez rizika automatizované interpretace řešení zkoušeným, aniž by musel látce v otázce zahrnuté rozumět. Systém je zatím stále ve vývoji, již nyní však zvládá zkušební provoz v reálném prostředí a jeho výsledky jsou zjevné.

Aplikace byla ve své první verzi nasazena při výuce programování na střední škole Podorlické vzdělávací centrum v Dobrušce od října 2010, jako hodnotící nástroj a to, díky své automatizaci při opravování, pro krátký opakovací test v téměř každé hodině. Celkem jejím prostřednictvím bylo jen za první pololetí 2010/11 se studenty napsáno 1 041 známkových testů [4] a 873 v pololetí druhém.

Ukázkový test si je možné vyzkoušet na adrese www.alltest.eu/demo, pro jeho spuštění je třeba mít nainstalován Silverlight 4.

Literatura

- [1] SCHINDLER, R. *Rukověť autora testových úloh*. Praha : Centrum pro zjišťování výsledků vzdělávání, 2006. ISBN 80-239-7111-5.
- [2] SKONNARD, A., GUDGIN, M. *Essential XML Quick Reference: A Programmer's Reference to XML, XPath, XSLT, XML Schema, SOAP, and More*. Boston : Addison-Wesley Professional, 2001. ISBN 978-0201740950.
- [3] VOBORNÍK, P. *Bezpečná autentizace aplikace klient-server v internetu pomocí povinně unikátních saltů*. In: Internet, bezpečnost a konkurenceschopnost organizací 2011. Zlín : Univerzita Tomáše Bati ve Zlíně, 16.–17. 3. 2011. s. 347–354. ISBN 978-80-7454-012-7.
- [4] VOBORNÍK, P. *Počítačové testovací systémy*. In: Sborník příspěvků z konference Alternativní metody výuky 2011. Praha : Univerzita Karlova, 28. 4. 2011. ISBN 978-80-7435-104-4.

Ing. Petr Voborník

Univerzita Hradec Králové, Fakulta informatiky a managementu
Rokitanského 62, 500 03 Hradec Králové, Česká republika
e-mail: petr.vobornik@uhk.cz