



Vysoká škola
polytechnická
Jihlava

Informatika 2026

Sborník příspěvků z konference

Vysoká škola polytechnická Jihlava

13.–14. ledna 2026
Jihlava, Česká republika

Konference Informatika 2026 se konala pod záštitou EUNIS.



Informatika 2026

Sborník příspěvků z konference

Vydavatel: Vysoká škola polytechnická Jihlava
Tolstého 1556/16, 586 01 Jihlava

Editor: Hana Vojáčková

Publikace neprošla jazykovou kontrolou.

Příspěvky jsou řazeny abecedně podle příjmení autora.

Všechny příspěvky prošly recenzním řízením.

Technické zpracování a výroba:

Vysoká škola polytechnická Jihlava

Tolstého 1556/16, 586 01 Jihlava

První vydání

2026

© Vysoká škola polytechnická Jihlava

© Autoři příspěvků

ISBN 978-80-88064-79-4 (online ; pdf)

Vědecký výbor

prof. Ing. Tomáš Vojnar, Ph.D., Masarykova univerzita

prof. Ing. Ladislav Beránek CSc., MBA, Jihočeská univerzita v Českých Budějovicích

doc. Ing. Zbyněk Bureš, Ph.D., České vysoké učení technické v Praze

doc. Ing. Karel Richta, CSc., Vysoká škola polytechnická Jihlava

PaedDr. František Smrčka, Ph.D., Vysoká škola polytechnická Jihlava

Mgr. Hana Vojáčková, Ph.D., Vysoká škola polytechnická Jihlava

Ing. Lenka Kuklišová Pavelková, Ph.D., Vysoká škola polytechnická Jihlava

Ing. Bc. Karel Dvořák, Ph.D., Vysoká škola polytechnická Jihlava

Organizační a programový výbor

PaedDr. František Smrčka, Ph.D.

Mgr. Hana Vojáčková, Ph.D.

Mgr. Zdeňka Dostálová

Ing. Marek Musil

Ing. Jakub Novotný, Ph.D.

Mgr. Antonín Příbyl

Michaela Machovcová

Webové stránky konference: <https://informatika2026.vspj.cz/>

Obsah

NARUŠENÍ INFORMAČNÍCH KASKÁD: MECHANISMY, PODMÍNKY A DŮSLEDKY	7
Ladislav Beránek, Radim Remeš, Josef Milota	7
OBSAH A INOVACE PŘEDMĚTU VÝVOJ VÝUKOVÉ APLIKACE	14
Jindra Drábková, Jana Kolaja Ehlerová	14
APLIKOVÁNÍ SYSTÉMU PLM PRO ZVÝŠENÍ KVALITY ŘÍZENÍ PROJEKTOVÝCH DAT	20
Karel Dvořák, Jana Dvořáková	20
MODERNÍ DIGITÁLNÍ TECHNOLOGIE VE VZDĚLÁVÁNÍ	26
Miroslav Hrubý	26
DIGITAL IS PHYSICAL – VÝUKA PROGRAMÁTORŮ S OHLEDEM NA EKOLOGICKOU STOPU	31
Lenka Kosková Třísková, Igor Kopetschke	31
PŘÍPRAVA VÝUKY V TANDEMU – VYUŽITÍ LLM JAKO PARTNERA PŘI TVORBĚ INTERAKTIVNÍHO KURZU PROGRAMOVÁNÍ V JUPYTERLAB	37
Petr Kropík, Lenka Šroubová	37
METRIKY VE SCRUMU: DOTAZNÍKOVÁ APLIKACE PRO ZJIŠTĚNÍ ÚROVNĚ VYSPĚLOSTI POUŽÍVÁNÍ AGILNÍCH METRIK V TÝMU	44
Josef Kunhart	44
INOVACE VE VÝUCE PODNIKOVÉ ARCHITEKTURY A DIGITÁLNÍ TRANSFORMACE V INFORMATICKÝCH PŘEDMĚTECH	51
Martin Lukáš	51
VÝUKA SEMINÁŘE ZÁKLADY ALGORITMIZACE V PRVNÍM ROČNÍKU	55
Radek Matoušek	55
TESTY ŘÍZENÝ VÝVOJ A JEHO NASAZENÍ VE VÝUCE	61
Marek Musil	61
UČITELSTVÍ INFORMATIKY NA FAKULTĚ INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE	65
Zdeněk Muzikář, Petr Zemánek	65
VÝUKA V ÉŘE AI: OD TEORIE K PRAXI PROSTŘEDNICTVÍM SAAS A SPOLUPRÁCE S FIRMAMI	72
Lukáš Novák, Jan Luhan	72
AUTOMATIZACE VIZUALIZACE GEOTAGOVANÉ FOTODOKUMENTACE V PROSTŘEDÍ QGIS: PROPOJENÍ VÝZKUMU V OBLASTI HERITAGE SCIENCE S VÝUKOU APLIKOVANÉ INFORMATIKY	79
Jakub Novotný, Alžběta Frýdlová	79
JAK STUDENTI VYUŽÍVAJÍ GENERATIVNÍ UMĚLÉ INTELIGENCE PŘI ŘEŠENÍ PROGRAMOVACÍCH ÚLOH	86
Radim Remeš, Ladislav Beránek, Josef Milota	86
VÝUKA PHP FRAMEWORKU SYMFONY	92
František Smrčka	92

OD KANCELÁŘSKÝCH BALÍKŮ K INFORMATICKÉMU MYŠLENÍ: PRAKTICKÁ ZKUŠENOST S VÝUKOU NOVÉ INFORMATIKY NA ZÁKLADNÍ ŠKOLE.....	98
Pavel Ševčík, Jiří Vojtěšek	98
ITERATIVNÍ GENEROVÁNÍ A LADĚNÍ MOODLE TESTŮ PRO VÝUKU PROGRAMOVÁNÍ S PODPOROU LLM	104
Lenka Šroubová, Petr Kropík	104
MODERNÍ METODY VÝUKY OBJEKTOVÉHO MODELOVÁNÍ A ROZVOJ DIGITÁLNÍCH KOMPETENCÍ STUDENTŮ	111
David Velner, Petr Tučník	111
OTEVŘENÁ DATA KE ZVYŠOVÁNÍ DIGITÁLNÍCH DOVEDNOSTÍ STUDENTŮ	117
David Velner, Petr Tučník	117
DIGITALIZACE SCINTILAČNÍCH SPEKTER: OD SKENOVANÝCH TABULEK K JSON POMOCÍ GENERATIVNÍ AI.....	123
Jan Zbirovský, Michal Jeřábek, Nikolaj Gribakin	123

Vážené čtenářky, vážení čtenáři,

po roce se Vám do rukou dostává další sborník konference Informatika 2026.

Vždy když se připravuje konference, mám pocit, že už vše bylo řečeno. Že už není o čem diskutovat a psát. O to je příjemnější vidět, že se stále v příspěvcích objevují zajímavá témata, která určitě účastníky konference obohatí.

V obsahu letošního sborníku převažují hlavně didaktická témata týkající se toho, jak učit nebo jak vytvořit syllabus předmětu, který by reflektoval aktuálními potřeby firem. Stále více příspěvků se zabývá umělou inteligencí a jejím využitím ve výuce. Studenty připravujeme, aby byli schopni se uplatnit v praxi, proto je dobře, že se objevují i příspěvky, které řeší spolupráci s firmami, například formou projektů, praxí studentů nebo závěrečných prací.

Věřím, že většina příspěvků tohoto sborníku Vás bude opět inspirovat ve Vašem pedagogickém procesu a ve Vaší další akademické práci.

Za organizační a programový výbor

PaedDr. František Smrčka, Ph.D.

NARUŠENÍ INFORMAČNÍCH KASKÁD: MECHANISMY, PODMÍNKY A DŮSLEDKY

DISRUPTION OF INFORMATION CASCADES: MECHANISMS, CONDITIONS AND CONSEQUENCES

Ladislav Beránek, Radim Remeš, Josef Milota

Abstrakt

Informační kaskády vznikají, když jednotlivci napodobují chování ostatních a ignorují jejich soukromé informace, protože sociální signály se zdají být spolehlivější. I když takové kaskády mohou usnadnit koordinaci a kolektivní akci, často potlačují cenné informace, což vede k rozsáhlým chybám, dezinformacím nebo neefektivnosti trhu. Informační kaskády však nejsou neměnné: mohou být narušeny strukturálními, kognitivními nebo technologickými zásahy, které obnovují informační rozmanitost a nezávislé uvažování. Předkládaný příspěvek popisuje teoretické základy informačních kaskád, zkoumá mechanismy, kterými jsou narušovány, a diskutuje empirické případy ilustrující tuto dynamiku.

Klíčová slova: Informační kaskády, sociální sítě, narušení kaskád

Abstract

Information cascades arise when individuals imitate the behavior of others and ignore their private information because social signals appear to be more reliable. While such cascades can facilitate coordination and collective action, they often suppress valuable information, leading to widespread errors, misinformation, or market inefficiencies. However, information cascades are not immutable: they can be disrupted by structural, cognitive, or technological interventions that restore information diversity and independent reasoning. This paper describes the theoretical foundations of information cascades, examines the mechanisms by which they are disrupted, and discusses empirical cases illustrating these dynamics.

Keywords: Information cascades, social networks, cascade disruptions

Úvod

V moderních společnostech jsou informační toky klíčové pro kolektivní rozhodování. Jednotlivci se zřídka rozhodují izolovaně; místo toho pozorují ostatní a podle toho aktualizují svá přesvědčení. Tento proces sociálního učení umožňuje efektivní koordinaci, pokud jsou soukromé a veřejné informace přesně agregovány (Chamley, 2004). Za určitých podmínek však jednotlivci přestávají spoléhat na své vlastní informace a začínají napodobovat jednání ostatních, čímž vytvářejí informační kaskádu (Bikhchandani, Hirshleifer a Welch, 1992). Informační kaskády představují narušení nezávislého úsudku: jakmile si několik prvních aktérů vytvoří vzorec chování, další aktéři je následují v domněnku, že předchozí rozhodnutí odhalují lepší informace. Tato dynamika může vést k systematickým chybám – jako jsou finanční bubliny, masové dezinformace nebo kulturní módní výstřelky – které přetrvávají i za

přítomnosti protichůdných důkazů. Kaskády však nejsou trvalé. Mohou být narušeny, což znamená, že proces slepé imitace je přerušen a jednotlivci se začnou znovu spoléhat na soukromé nebo heterogenní informace. Pochopení toho, jak k takovým narušením dochází, je klíčové pro zlepšení kvality rozhodování, stabilizaci trhů a podporu epistemického zdraví v digitálních společnostech.

Kooperativní chování je pozorováno v různých měřítcích biologické organizace, od genů a buněk po sociální skupiny a technologické systémy (Szolnoki a Perc, 2008). Spolupracující však platí cenu za to, že generují výhody pro ostatní, zatímco podvodníci spolupracující využívají, aniž by přispívali. Toto napětí vyvolává zásadní otázku: jak spolupráce přetrvává tváří v tvář takovému jednání.

Evoluční teorie her, a zejména dilema vězně (PD), poskytuje silný rámec pro studium tohoto problému. Strukturované populace – kde jednotlivci interagují v síti – umožňují spolupracujícím tvořit klastry, které je chrání před vykořisťováním. V dynamických sítích, kde nováčci vstupují a vytvářejí spojení, však přetrvávání spolupráce závisí na tom, jak si nováčci vybírají své interakční partnery (Xiao, Chen a Szolnoki, 2020; Ha, Chen a Szolnoki, 2024; Jia et. al, 2020).

V mnoha systémech nováčci napodobují strategii a sociální okolí vzoru, což je proces známý jako sociální dědictví. V ideálním případě by se nováčci měli přednostně spojovat s kooperátory a vyhýbat se podvodníkům, ale kooperátory často nelze od podvodníků dokonale odlišit. Rozhodování je tedy založeno na kombinaci soukromých informací, které jsou sice nejasné, ale specifické pro jednotlivce, a veřejných informací, které pocházejí z pozorování rozhodnutí předchozích nováčků (Wang et. al, 2015).

Zde zkoumáme, jak kvalita těchto rozhodnutí – kvantifikovaná jejich citlivostí (přesná identifikace spolupracujících) a specifičností (přesné odmítnutí podvodníků) – ovlivňuje dlouhodobou prevalenci spolupráce. Ukazujeme, že optimální rovnováha mezi citlivostí a specifičností silně závisí na síle selekce, tj. na tom, jak silně výnosy ovlivňují to, které jedince jsou napodobováni.

Kromě toho, když jsou zahrnuty veřejné informace, mohou se chybné volby šířit systémem a vytvářet informační kaskády, které degradují spolupráci. Naše práce objasňuje, jak tyto kaskády vznikají a jak interagují s evoluční dynamikou, aby formovaly osud spolupráce.

Tento článek zkoumá narušení informačních kaskád multidisciplinárním pohledem. Oddíl 2 nastiňuje teoretické základy informačních kaskád a model. Oddíl 3 identifikuje a analyzuje mechanismy, které narušují nebo ukončují kaskády. Oddíl 4 uzavírá celý příspěvek obecnými doporučeními.

Model

Uvažujeme populaci N uzlů uspořádaných v dynamicky se vyvíjející síti. Každý uzel přijímá jednu ze dvou strategií modelu vězeňského dilematu (PD): spolupráci nebo zradu. Spolupracující uzel platí náklady c , aby poskytl výhodu b každému sousedovi, zatímco nespolupracující nemá žádné náklady a neposkytuje žádné výhody.

V typické hře PD dvou hráčů má každý hráč na výběr dvě strategie, a to spolupráci (C) a přeběhnutí (D) (Nowak a Sigmund, 2004). Obecnou matici výplat lze zapsat takto

$$\begin{matrix} & \begin{matrix} C & D \end{matrix} \\ \begin{matrix} C \\ D \end{matrix} & \begin{pmatrix} b-c & -c \\ b & 0 \end{pmatrix} \end{matrix} \quad (1)$$

Předpokládáme, že $b > c > 0$

Výplata hráče i v určitém kole hry může být vyjádřena obecným vztahem:

$$\pi_i = \sum_{j \in \Omega_i} \pi_{ij} = \sum_{k=1,2} \sum_{j \in \Omega_i} S_{ij}^T A_k S_{ij} \quad (2)$$

Kde π_i představuje celkový příjem uživatele i , což je součet příjmů získaných hrou uživatele i s každým jeho sousedem; matice A_k odpovídá matici (1) podle umístění příslušných sousedů; Ω_i představuje množinu sousedů uzlu i v daném čase; S_{ij} představuje strategii uzlu i vůči sousedu j :

Výplata P_i , kterou uzel n_i v síti obdrží, je definována jako součet výplat z párových interakcí se všemi jeho sousedy. Má-li spolupracující uzel m kooperujících a n podvádějících sousedů, jeho výplata je dána výrazem

$$P_i = m(b - c) - nc. \quad (3)$$

Podvádějící uzel ve stejném sousedství naopak získává výplatu

$$P_i = mb, \quad (4)$$

neboť využívá výhod kooperace, aniž by sám nesl odpovídající náklady.

Prosperita populace je definována jako průměrná výplata všech uzlů v síti. Vhodnost uzlu n_i je následně specifikována pomocí exponenciální funkce s parametrem selekční síly δ :

$$f_i = (1 + \delta)^{P_i}, \quad (5)$$

kde P_i představuje výplatu příslušného uzlu.

Každý krok aktualizace sítě probíhá následujícím způsobem (viz obr. 1). Nejprve je vybrán vzorový uzel, přičemž pravděpodobnost jeho výběru je úměrná jeho vhodnosti. Parametr δ zde určuje intenzitu selekce: při $\delta = 0$ jsou všechny uzly vybírány se stejnou pravděpodobností bez ohledu na jejich výplatu, zatímco zvýšení hodnoty δ zvyšuje pravděpodobnost výběru uzlů s vyšší výplatou.

Následně je do sítě přidán nový uzel, který s pravděpodobností $1 - \mu$ převeze strategii vzorového uzlu, zatímco s pravděpodobností μ zvolí alternativní strategii. Parametr μ zde reprezentuje míru mutací a předpokládá se, že nabývá malé hodnoty (v našich výpočtech $\mu = 0,0001$).

Pravděpodobnost, že zkopíruje strategii svého soupeře do nového kola lze vyjádřit upraveným Fermiho vztahem (Szolnoki et al.):

$$\mu_{i \leftarrow j} = \frac{1}{1 + \exp\left(\frac{\pi_{i,t} - \pi_{j,t}}{\kappa}\right)} \quad (6)$$

kde κ představuje sílu výběru, resp. množství šumu. Toto aktualizací pravidlo říká, že strategie výkonnějšího hráče je snadno přijata, zatímco je nepravděpodobné (ale ne nemožné) přijmout

strategie méně výkonných hráčů. Parametr k zahrnuje prvek nejistoty při přijímání strategií. V limitě $k \rightarrow \infty$ jsou všechny informace ztraceny, to znamená, že hráč i není schopen získat žádnou informaci o strategii svého soupeře.

Zároveň hráč ale zkoumá, zda by mu tato změna strategie v dalším kole hry s hráči ve stejné skupině nebo s hráči z opačně skupiny přinese vyšší výplatu:

$$\pi_i(S_i^{new}(t+1), S_j(t+1)) > \pi_i(S_i(t+1), S_j(t+1)) \quad (7)$$

Pokud by nová strategie nebyla pro hráče i výhodnější (13), pak hráč změnu strategie neprovede.

Po vložení nového uzlu jsou vytvořeny jeho vazby k ostatním uzlům. Tento proces může probíhat na základě pouze soukromých informací.

Zde pracujeme s možností pouze soukromých informací. Soukromá informace je modelována pomocí pravděpodobnosti $\zeta_i(t)$, kterou určí hráč na základě svých zkušeností Hanaki et al., 2007):

$$\zeta_i(t+1) = \omega \cdot \zeta_i(t) + (1 - \omega) \cdot R_i(t) \quad (8)$$

kde ω je váhový faktor a $R_i(t)$ měří důvěru hráče v ostatní. Hodnota R_i se upravuje podle zkušeností hráče s průběžnými partnery (spolupracovníky) a nově oslovenými partnery. $R_i(t)$ vyjadřujeme poměr hráčů přinášejících zisk mezi všemi partnery hráče i v posledním t herním kroku.

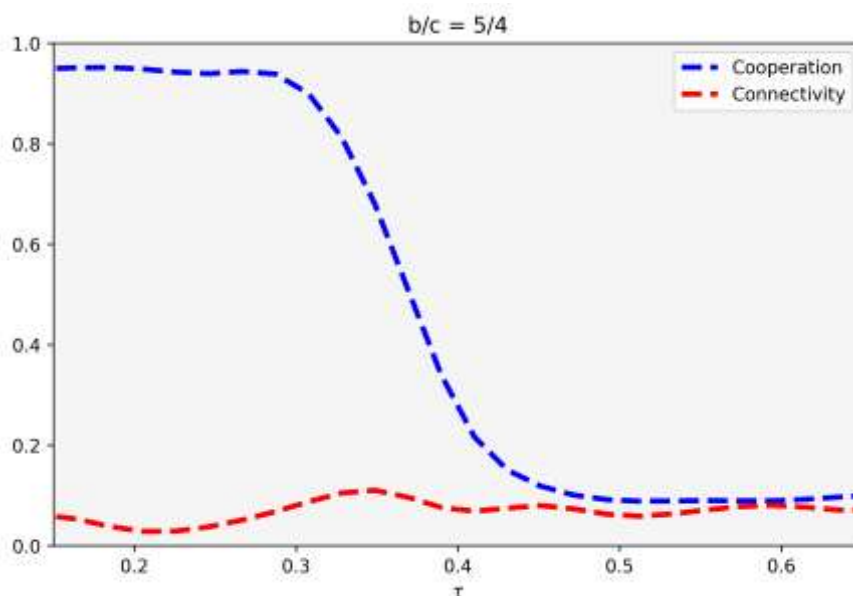
Když však hráč hledá další partnery, marginální zisk z těchto interakcí může být malý ve srovnání s náklady. Při výpočtu očekávané výplaty musí hráči předvídat budoucí akci svého potenciálního partnera. Jeho cílem je optimalizovat svůj celkový příjem (2) optimalizací počtu partnerů, které osloví v každém kole.

Hráči se snaží na základě těchto vztahů optimalizovat počet svých partnerů. Je třeba poznamenat, že nový vztah vzniká, pokud oba potenciální partneři souhlasí. Současně iniciující hráč a potenciální partner provádějí stejné kalkulace nákladů a přínosů.

Důležité v našem modelu je, že pokud přestává s určitým podvádějícím uzlem spolupracovat větší množství spoluhráčů, daný podvádějící uzel je vyloučen ze sítě. Tento poměr označíme písmenem τ .

Výsledky

Navržený výpočetní model byl analyzován prostřednictvím numerických simulací a počítačových vizualizací realizovaných pomocí specializovaného softwarového nástroje vytvořeného pro účely této studie. Simulace byly inicializovány na náhodně propojené síti o velikosti (N) uzlů s průměrnou konektivitou ($k = 4$). Pro každý běh simulace byl rozhodovací práh (τ) fixován na hodnotu náhodně vybranou z rovnoměrného rozdělení na intervalu $([0.1, 0.7])$. Všechny uzly byly na počátku nastaveny tak, aby používaly stejnou strategii. Statistické charakteristiky byly získávány z jednoho běhu zahrnujícího více aktualizací kroků.



Obrázek 1: Spolupráce a propojenost jako funkce rozhodovacího prahu τ . Zde volíme $\delta = 0,01$. Výsledky jsou získány s použitím období 10^3 časových kroků.

Dlouhodobá spolupráce (dále jen *spolupráce*), prosperita a konektivita byly definovány jako průměry příslušných ukazatelů přes všechny kroky simulace, tj. jako součet podílu spolupracujících uzlů, průměrné výplaty a průměrného stupně, vydělený celkovým počtem kroků. Tyto hodnoty byly následně normalizovány na interval $[0, 1]$ vydělením maximálně možnou dosažitelnou hodnotou. Současně jsme zaznamenávali počet přechodů mezi dvěma homogenními stavy sítě – stavem, v němž všechny uzly spolupracují, a stavem, v němž všechny uzly podvádějí.

Protože nově příchozí uzly preferují navazování spojení se spolupracujícími jedinci, monitorovali jsme v průběhu simulací také výskyt falešně pozitivních (FP), falešně negativních (FN), skutečně pozitivních (TP) a skutečně negativních (TN) rozhodnutí. Pokud se nově příchozí uzel rozhodl navázat spojení se spolupracujícím uzlem (x), bylo takové rozhodnutí klasifikováno jako TP; pokud se rozhodl s kooperujícím uzlem nespojit, šlo o FN. Pokud se nově příchozí uzel rozhodl navázat spojení s podvádějícím uzlem, jednalo se o FP, zatímco odmítnutí spojení s podvádějícím bylo považováno za TN.

Na základě těchto údajů byly vypočteny metriky specifity a citlivosti, definované jako

$$\text{specifita} = \frac{TN}{TN + FP}, \quad \text{citlivost} = \frac{TP}{TP + FN}.$$

Různé hodnoty rozhodovacího prahu τ vedou k odlišným kombinacím specifity a citlivosti. Specifita zde udává schopnost nově příchozích správně identifikovat podvodníky, zatímco citlivost měří schopnost správné identifikace spolupracujících uzlů.

Systematický vliv rozhodovacího prahu τ na míru spolupráce je analyzován na obr. 1, kde je rovněž znázorněno, jak se dlouhodobá spolupráce, prosperita a počet přechodů mezi kooperací a defekcí mění v závislosti na hodnotě prahu τ . Za obecného předpokladu platí, že spolupráce může být udržitelná tehdy, pokud spolupracující jedinci úspěšně minimalizují spojení s defektory a současně maximalizují spojení s jinými kooperátory. Tomu odpovídají rozhodnutí nacházející se na střední úrovni specifity a citlivosti.

Obecně lze konstatovat, že vysoce rozhodnutí charakterizovaná nízkou specificitou mají tendenci negativně ovlivňovat dlouhodobou spolupráci. Nicméně míra takového rozhodnutí, která maximalizuje spolupráci, závisí na intenzitě selekčního tlaku mezi spolupracujícími a podvádějícími. Optimální kombinace specificity a citlivosti tedy není triviální: rozhodovací režimy, které jsou za určitých podmínek pro spolupráci škodlivé, mohou při změně síly výběru naopak podporovat její udržení.

Jak vyplývá z obr. 1, při slabé selekci je nejvyšší dlouhodobé spolupráce dosaženo tehdy, když jsou rozhodnutí nově příchozích vysoce citlivá a současně s nízkou specificitou. Při silné selekci jsou však tato rozhodnutí kontraproduktivní (obráz. 1). V tomto případě je nejvyšší míry spolupráce dosaženo tehdy, pokud jedinci přijímají rozhodnutí vyznačující se vysokou specificitou a nízkou citlivostí.

Pokud je poměr přínosu k nákladům obecně nižší, dochází k celkovému snížení míry dlouhodobé spolupráce. Avšak i v těchto podmínkách zůstává rozhodovací režim, který spolupráci usnadňuje, závislý na síle výběru. Současně platí, že rozhodovací práh τ , za kterým dochází k rozpadu kooperace, se se zvyšující se selekční silou systematicky snižuje.

Obecně lze konstatovat, že digitální informační ekosystém zintenzivňuje formování kaskád a zároveň umožňuje rychlejší narušení. Komunikace v reálném čase, algoritmická personalizace a metriky sociálního ověřování podporují stádové chování; digitální systémy však také umožňují rychlou korekci, jakmile se důvěryhodné informace stanou viditelnými.

Závěr

Informační kaskády vznikají z racionálního, ale omezeného sociálního učení, kde imitace nahrazuje nezávislé uvažování. Jejich narušení obnovuje informační rozmanitost a umožňuje společností zotavit se z kolektivních chyb. Tato práce nastínila informační, strukturální a behaviorální mechanismy, které jsou základem kaskádového narušení. V éře digitální hyperkonektivity je kaskádové narušení výzvou i příležitostí. Navrhování platforem a politik, které podporují transparentnost, rozmanitost a kritickou autonomii, může transformovat křehké kaskády v odolné systémy. Konečným cílem není eliminovat sociální vliv – základní součást lidské spolupráce – ale zajistit, aby vliv doplňoval, spíše, než nahrazoval, uvažování založené na důkazech.

Literatura

Bikhchandani, S., Hirshleifer, D., & Welch, I. (1992). A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of Political Economy*, 100(5), 992–1026.

Chamley, C. P. (2004). *Rational herds: Economic models of social learning*. Cambridge University Press.

Szolnoki, A. & Perc, M. (2008). Coevolution of teaching activity promotes cooperation. *New Journal of Physics*, 10, 043036.

Wang, Z., Wang, L., Szolnoki, A. & Perc, M. (2015). Evolutionary games on multilayer networks: a colloquium. *The European Physical Journal B*, 88, 124.

Nowak, M. A. & Sigmund, K. (2004) Evolutionary dynamics of biological games. *Science* 303, 793–799.

Szolnoki, A. Perc, M. & Danku, Z. (2008). Towards effective payoffs in the prisoner's dilemma game on scale-free networks, *Physica A: Statistical Mechanics and its Applications*, 387(8 – 9), 2075-2082.

Xiao, Z. Chen, X. & Szolnoki, A. (2020). Leaving bads provides better outcome than approaching goods in a social dilemma, *New Journal of Physics*, 22 (February 2020) *New Journal of Physics*, 22, 023012.

Han, J. Chen, X. & Szolnoki, A. (2024) When selection pays: Structured public goods game with a generalized interaction mode. *Chaos*, 34 (3), 033124.

Jia, D., Xinyu, W., Zhao, S., Romić I., Li, X., Marko, J. & Wang Z. (2020). Evolutionary dynamics drives role specialization in a community of players. *R. Soc. Interface*, 17(168), 1720200174.

Hanaki, N., Peterhansl, A., Dodds, P. & Watts, D.J. (2007). Cooperation in evolving social networks, *Management Science*, 53(7), 1036–1050.

Kontakní údaje

Jméno a příjmení včetně všech titulů
Název pracoviště
Adresa pracoviště, PSČ Město
e-mail:

Ladislav Beránek, prof. Ing. CSc.
Katedra dat. věd a počítačových systémů, FZT, JČU
Studentská 1668, 370 05 České Budějovice
beranek@jcu.cz

Jméno a příjmení včetně všech titulů
Název pracoviště
Adresa pracoviště, PSČ Město
e-mail:

Radim Remeš, Mgr., PhD.
Katedra dat. věd a počítačových systémů, FZT, JČU
Studentská 1668, 370 05 České Budějovice
inrem@jcu.cz

Jméno a příjmení včetně všech titulů
Název pracoviště
Adresa pracoviště, PSČ Město
e-mail:

Josef Milota, RNDr.
CIT Centrum informačních technologií
Branišovská 1645/31a, 370 05 České Budějovice
milota@jcu.cz

OBSAH A INOVACE PŘEDMĚTU VÝVOJ VÝUKOVÉ APLIKACE

CONTENT AND INNOVATION OF THE COURSE EDUCATIONAL APPLICATION DEVELOPMENT

Jindra Drábková, Jana Kolaja Ehlerová

Abstrakt

Příspěvek představuje cíle a pojetí povinného magisterského předmětu Vývoj výukové aplikace, který je určený pro studenty učitelství informatiky. Kurz připravuje studenty nejen na tvorbu současných vzdělávacích aplikací, ale i na práci s technologiemi, které dosud nejsou známy. Zdůrazňuje adaptabilitu, kritické myšlení a schopnost samostatného vývoje softwaru. Studenti navrhují, programují a obhajují vlastní projekty propojující pedagogické cíle s technickými prostředky. Příspěvek dále popisuje obsah, metody výuky a návrhy inovací reagujících na rychlý technologický vývoj.

Klíčová slova: učitelství informatiky, vývoj výukové aplikace, inovace

Abstract

The paper presents the objectives and concept of the compulsory master's course Educational Application Development, designed for students of computer science teacher training. The course prepares students not only for creating contemporary educational applications but also for working with technologies that are not yet known. It emphasizes adaptability, critical thinking, and the ability to independently develop software. Students design, program, and defend their own projects that connect pedagogical goals with technical tools. The paper also describes the course content, teaching methods, and proposed innovations in response to rapid technological development.

Keywords: computer science teacher training, educational application development, innovation

Úvod

Předmět Vývoj výukové aplikace (VVA) je zařazen do studijního programu pro navazující magisterské studium obor Informatika (program: Učitelství pro střední školy a 2. stupeň základních škol). Více informací o předmětu je možné nalézt v informačním systému STAG Technické univerzity v Liberci (Předmět VVA, 2025). Předmět je do studia zařazen ze dvou důvodů. Jednak studenti musí projít vývojem aplikace od začátku (návrhu) do konce (fungující aplikace), a tak se dostanou do pozice tvůrce, a jednak sami testují hotové aplikace svých kolegů, a tak se dostávají do pozice testera.

Obsah a hodnocení předmětu Vývoj výukové aplikace

Předmět se vyučuje jednou za dva roky, protože je málo studentů informatiky. Učí se tedy pro první a druhý ročník navazujícího magisterského studia najednou. Předmět je ukončen klasifikovaným zápočtem. Znamka je stanovena podle hodnocení několika úkolů, které jsou uvedeny v tabulce 1. Součástí hodnocení jednotlivých aplikací je také testování a připomínkování aplikací, které vytvořili ostatní studenti.

Tabulka 1: Tabulka hodnocení jednotlivých projektů

Zhodnocení 5 výukových aplikací	5
Vytvoření výukového materiálu pro informatiku	10
Tvorba aplikace základní počty (zkoušení)	20
Tvorba aplikace zkoušení slovíček	20
Tvorba vlastní aplikace	45

Hodnocení hotových výukových aplikací

V úvodních hodinách studenti vyhledávají a popisují výukové aplikace. Studentům jsou v e-learningovém kurzu k dispozici odkazy na různé aplikace, a to jak online, tak s instalací na lokální počítač. Většinou mají studenti zkušenosti s aplikacemi, se kterými se setkali na praxi nebo ve výuce. Studenti vybírají z velké části aplikace, které jsou k dispozici online a zdarma. V tabulce 2 jsou uvedeny body, co vše studenti hodnotí. Uvádíme zde i příklady aplikací, které si studenti vybírají k hodnocení:

- Umíme informatiku
- Duolingo
- Photomath
- Blockly Games
- Wolfram Alfa
- Učebnice online

Tabulka 2: Otázky pro hodnocení výukové aplikace

Kde jste se o výukové aplikaci dozvěděli?
Odkud jste výukovou aplikaci stáhli? / Kde je výuková aplikace k dispozici?
Je nutná instalace?
Je aplikace dostupná v českém jazyce?
Cíl výukové aplikace
Stručná charakteristika
Hodnocení grafické stránky aplikace
Hodnocení didaktické stránky aplikace
Celkové hodnocení aplikace (použitelnost)
Klady a zápory výukové aplikace
Věk, pro který je aplikace určena

Vytvoření výukového materiálu pro informatiku

Další úkol, který studenti odevzdávají, je vytvořený výukový materiál v online prostředí pro informatiku. Studenti si mohou vybrat jak téma, tak prostředí, ve kterém materiál vytvoří.

Téma obvykle souvisí s tím, co probírali nebo probírají s žáky na praxích (historie počítačů, hardware, počítačové sítě, internet, bezpečnost, modely, Python, kódování, typografická pravidla, Scratch – operátory). Většina vytváří testy, které je možné použít jak na začátku výuky k zopakování nebo testování látky z minule, tak po výkladu na konci, pro rekapitulaci hodiny.

Uvádíme zde také příklady některých z prostředí:

- Formuláře Google
- Kahoot
- Blooket
- Quizizz
- Quizlet
- Wordwall
- H5P
- Baamboozle

Vytvoření vlastních aplikací

Hlavním cílem předmětu je vytvoření vlastních aplikací. Studenti mohou použít libovolný programovací jazyk, který umožňuje vytvářet aplikace v grafickém prostředí. Mezi programovacími jazyky, se kterými se studenti setkali během studia, dominuje Python. Jsou jedinci, kteří vytvářejí webové aplikace. Studenti vytvářejí tři aplikace, zkoušení základních počtů, zkoušení slovíček (hlavních měst, chemických prvků apod.) a svoji vlastní výukovou aplikaci, která většinou rozšiřuje některou z uvedených dvou aplikací. Každá aplikace prochází testováním. Studenti si testují aplikace navzájem s tím, že také navrhuji úpravy aplikace. Finální testování provádí vyučující.

Použití nástrojů umělé inteligence při tvorbě aplikací

V současné době nástroje umělé inteligence dokážou velmi dobře vygenerovat zdrojový kód nějaké aplikace. Studenti mohou tyto nástroje používat. Při tvorbě aplikací je vždy nutné, aby vygenerovanému kódu rozuměli a aby uměli zdrojový kód upravit.

Velké jazykové modely mohou začínajícím programátorům významně usnadnit první kroky v novém programovacím jazyce. Jak však ukazují aktuální studie (Jošt, Taneski, Karakatič, 2024), s rostoucí složitostí úloh již jejich používání nepodporuje hlubší porozumění principům jazyka ani rozvoj klíčových kognitivních dovedností. Při inovaci předmětu je proto zohledněno, že studenti tyto nástroje běžně využívají, avšak cílem předmětu není zlepšování technik vyhledávání metod či návrhu aplikací a uživatelských interakcí, ale především rozvoj jejich schopnosti programovat a tvořit s porozuměním.

Inovace předmětu Vývoj výukové aplikace

Cíle výuky

Cílem výuky v předmětu Vývoj výukové aplikace je, aby student získal ucelený přehled o typech výukových aplikací a dokázal je smyslně a pedagogicky vhodně využívat ve výuce. Současně má být schopen navrhnout a vytvořit vlastní výukovou aplikaci v programovacím jazyce, který je mu blízký a ve kterém se dokáže samostatně orientovat.

Metody výuky

K výuce jsou nově zařazeny aktivizační metody. Má-li se student stát aktivním učitelem, měl by sám zažít aktivní výukové přístupy. Tyto metody zároveň představují efektivní způsob (Freeman, 2014, Theobald, 2020), jak podpořit zapojení studentů a jejich hlubší porozumění probírané problematice. Pro naplnění cíle, aby se studenti dokázali orientovat v nabídce výukových aplikací, je zvolena rešerše doplněná o vrstevnickou výuku, která umožňuje sdílení zkušeností a vzájemné obohacení mezi studenty.

Časový harmonogram aktivit

Studenti mají za úkol samostatně nastudovat šest různých výukových aplikací a zpracovat rešerši, která zahrnuje jejich popis a návrh vhodných úloh pro výukové použití. Následně si z této sady vyberou dvě aplikace, které jsou nejvhodnější pro jejich oborové zaměření, a připraví pro ně dvě patnáctiminutové výukové aktivity určené pro své spolužáky. Tyto aktivity následně realizují v rámci cvičení, kde si vyzkoušejí roli učitele v praxi. Každá aktivita je následně reflektována jak vyučujícím, tak i spolužáky.

Část předmětu zaměřená na návrh vlastní výukové aplikace je strukturována do několika fází: návrh, vypracování, testování a prezentace. Samotný návrh připravuje student individuálně, avšak následně jej představuje v rámci cvičení, kde získává zpětnou vazbu ohledně proveditelnosti, didaktické vhodnosti a pedagogického přínosu svého záměru. Fáze vypracování probíhá samostatně a navazuje na ni testování, které si student plánuje a realizuje během své povinné pedagogické praxe. Závěrečná prezentace probíhá formou výukové aktivity před spolužáky, po níž následuje řízená reflexe zaměřená na kvalitu návrhu, provedení i pedagogické aspekty výsledné aplikace.

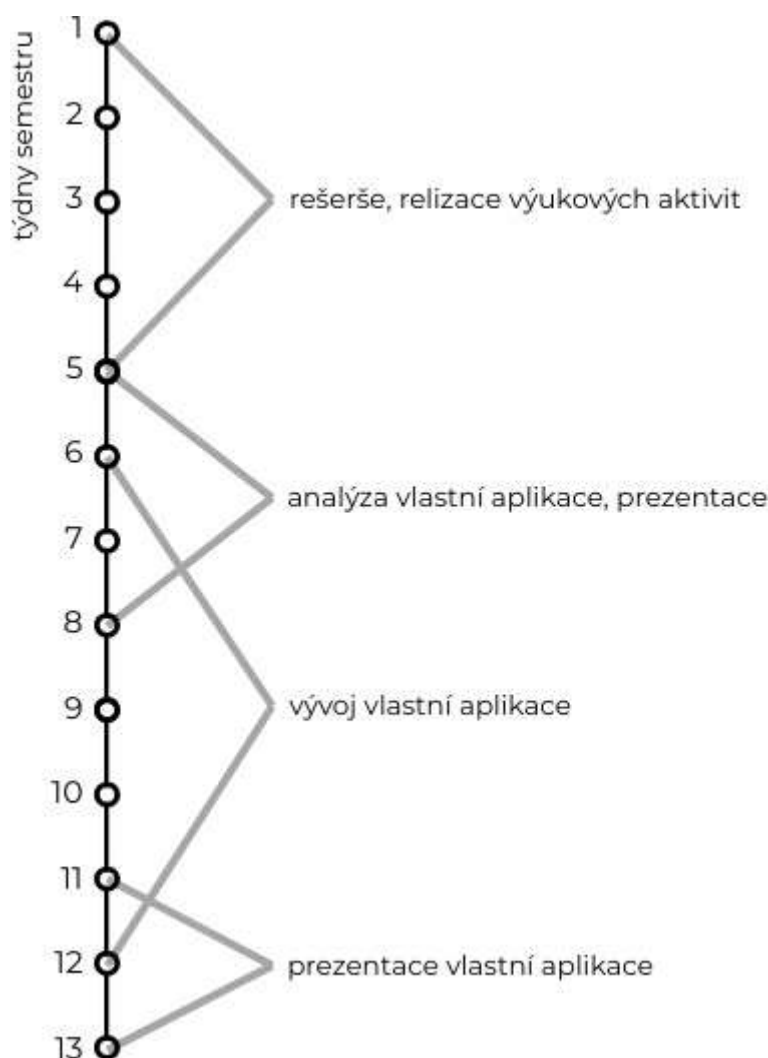
Zkouška z předmětu představuje souhrnné zhodnocení všech aktivit, které student v průběhu semestru realizoval, a zároveň ověřuje jeho schopnost samostatně připravit vlastní výukovou aplikaci. Součástí hodnocení je také orientace v dostupných výukových aplikacích, metodách a možnostech jejich pedagogického využití.

Zpětná vazba

Zkouška v závěru semestru poskytuje důležitou formu zpětné vazby, protože zachycuje skutečný stav znalostí a dovedností studentů po absolvování předmětu. Umožňuje ověřit, zda studenti dosáhli stanovených výstupů učení a zda jsou schopni samostatně aplikovat získané poznatky při návrhu a vývoji vzdělávacích aplikací. Tato forma hodnocení však nepostihuje celý proces učení – nereflektuje míru individuálního posunu studenta ani rozdíl mezi jeho výchozími znalostmi a schopnostmi na začátku kurzu. Proto je doplněna dalšími nástroji reflexe, které umožňují zachytit i subjektivní stránku studijní zkušenosti.

Pro systematickou reflexi předmětu byl navržen anonymní dotazník, jehož cílem je zachytit nejen zpětnou vazbu k použitým výukovým metodám, ale i konkrétní zkušenosti studentů

s jednotlivými aktivitami. Dotazník se zaměřuje rovněž na subjektivní pocity (Deslauriers, 2019) z průběhu semestru, vnímanou náročnost a přínos předmětu. Součástí je také srovnávací položka, která umožňuje studentům vyjádřit, jak předmět hodnotí ve vztahu k průměrné kvalitě ostatních kurzů v jejich studiu. Tato kombinace poskytuje vyvážený pohled na efektivitu předmětu i na jeho dopad na studijní zkušenost.



Obrázek 1: Časový harmonogram předmětu VVA

Závěr

Je jisté, že velké množství předmětů, které jsou určeny pro budoucí učitele informatiky, musí být koncipováno tak, aby bralo v úvahu strmý vývoj digitálních technologií. V článku je představen předmět Vývoj výukové aplikace tak, jak je vyučován nyní a jaké jsou vize o tom, jak by předmět mohl vypadat v budoucnosti. Důležité je uvědomit si, že předmět je určen pro budoucí učitele informatiky, a tudíž je potřeba zohledňovat zejména pedagogický a didaktický přínos předmětu.

V rámci inovace byla navržena nová koncepce předmětu, která systematicky využívá metody aktivního učení a kromě samostatného vývoje aplikace důrazně podporuje také kooperativní a vrstevnickou výuku; součástí této koncepce je tvorba výukových aktivit určených pro pedagogickou praxi a důraz na jejich následnou reflexi.

Literatura

- [1] Beimel, D., Tsoury, A. a Barnett-Itzhaki, Z. (2024). The impact of extent and variety in active learning methods across online and face-to-face education on students' course evaluations. *Frontiers in Education*, roč. 9. DOI: 10.3389/feduc.2024.1432054.
- [2] Deslauriers, Louis, McCarty, Logan S., Miller, Kelly, et al. (2019). Measuring actual learning versus feeling of learning in response to being actively engaged in the 31classroom. *Proceedings of the National Academy of Sciences*, 116(39), 19251–19257. DOI: 10.1073/pnas.1821936116
- [3] Freeman, Scott, Eddy, Sarah L., McDonough, Miles, et al. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23), 8410–8415. DOI: 10.1073/pnas.1319030111
- [4] Jošt, G., Taneski, V., Karakatič, S. (2024). The Impact of Large Language Models on Programming Education and Student Learning Outcomes. *Applied Sciences* 14(10). DOI: 10.3390/app14104115
- [5] Předmět Vývoj výukové aplikace (2025), [online], [2025-11-25]. IS STAG studijní agenda. Dostupné z: <https://stag.tul.cz/StagPortletsJSR168/CleanUrl?urlid=prohlizeni-predmet-sylabus&predmetZkrPrac=KGD&predmetZkrPred=PVVA&predmetRok=2025&predmetS emestr=ZS>
- [6] Theobald, Elli, Hill, M. J., Tran, Ednali, et al. (2020). Active learning narrows achievement gaps for underrepresented students in undergraduate science, technology, engineering, and math. *Proceedings of the National Academy of Sciences*, 117(12), 6476–6483. DOI: 10.1073/pnas.1916903117.

Kontaktní údaje

Jindra Drábková

Katedra geoinformatiky a didaktiky informatiky, Fakulta přírodovědně-humanitní a pedagogická,
Technická univerzita v Liberci Studentská 1402/2, 46117, Liberec 1

e-mail: jindra.drabkova@tul.cz

APLIKOVÁNÍ SYSTÉMU PLM PRO ZVÝŠENÍ KVALITY ŘÍZENÍ PROJEKTOVÝCH DAT

APPLICATION OF PLM SYSTEM TO IMPROVE PROJECT DATA MANAGEMENT QUALITY

Karel Dvořák, Jana Dvořáková

Abstrakt

Systémy PLM podporují komplexní řízení projektů tím, že propojují data, procesy a týmy v rámci celého životního cyklu produktu. Umožňují plánování milníků, správu úkolů, řízení požadavků i kontrolu rizik. Integrují změnová řízení, správu dokumentace a kusovníků, čímž zajišťují aktuálnost a trasovatelnost informací. Díky workflow automatizují schvalování a podporují efektivní spolupráci napříč odděleními. Projektovým manažerům poskytují přehled o stavu projektu prostřednictvím reportů a dashboardů. PLM tak zvyšuje kvalitu vývoje, zrychluje rozhodování a umožňuje řídit složité technické projekty konzistentně a transparentně.

Klíčová slova: Řízení životního cyklu výrobku, Virtuální prototyp, Sdílení dat, Kvalita řízení dat

Abstract

PLM systems support comprehensive project management by connecting data, processes, and teams throughout the entire product lifecycle. They enable milestone planning, task management, requirements management, and risk control. They integrate change management, documentation management, and bill of materials management, ensuring that information is up to date and traceable. Thanks to workflow, they automate approvals and support effective cross-departmental collaboration. They provide project managers with an overview of the project status through reports and dashboards. PLM thus improves development quality, speeds up decision-making, and enables complex technical projects to be managed consistently and transparently.

Keywords: Product lifecycle management, Virtual prototype, Data sharing, Data quality management

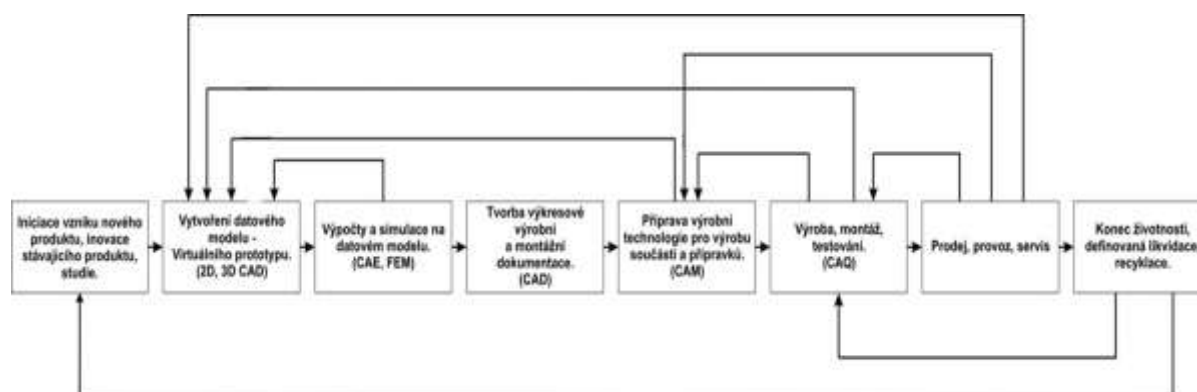
Úvod

Řízení životního cyklu projektu zahrnuje široké spektrum postupů zpracování projektových dat, poskytovaných obvykle více řešiteli. Zejména rozsáhlejší a dlouhodobé projekty, řízené často paralelně s jinými projekty různého zaměření, vyžaduje efektivní přístup k projektovým datům, často z více různých míst a v libovolném čase. Současné informační a komunikační prostředky dovolují široké spektrum metod přístupu, sdílení a obecného řízení projektových dat (Jun 2007; Schuh 2008). Specifickým atributem je typ dat v souvislosti s charakterem řešeného projektu. V oblasti vývoje a výzkumu, zejména ve strojírenství a příbuzných oborech, představují velký objem klíčových dat virtuální prototypy produktů ve formě CAD dat. Jde

o soubor informací prostřednictvím 3D modelové a 2D výkresové dokumentace jednotlivých součástí příslušného produktu, nebo celých sestav finálního výsledného produktu. Paralelně s 3D modely a 2D výkresy jsou součástí projektových dat všeobecné dokumenty ve formě textu, tabulek, obrázků, nebo programových kódů. Jde o široké spektrum formátů souborů s různými požadavky na příslušné softwarové nástroje pro jejich zobrazení a editaci. Například data virtuálních prototypů jsou obvykle zobrazována a editována prostřednictvím příslušného CAD nástroje (Dekhtiar 2018). Globálně existuje více CAD nástrojů od různých poskytovatelů s velmi malou kompatibilitou a přenositelností mezi různými aplikacemi (Fořt 2006). Jde současně o poměrně nákladné licencované aplikace, kde není efektivní a účelné, aby každý člen vývojového týmu disponoval nástrojem a příslušnou licencí. Jednou z významných funkcionalit PLM systému je možnost sdílení 3D dat s relevantními informacemi a možností editace vybraných informací (Stark 2022, Dvořák 2023). Každý člen týmu má k dispozici jednotné sdílené prostředí.

Řízení životního cyklu výrobku – PLM

PLM – Product Lifecycle Management, řízení životního cyklu výrobku, případně produktu představuje metodiku správy dat a souvisejících procesů od prvotní iniciace vzniku nového produktu, nebo významné inovace stávajícího produktu (Mas 2016). Obvykle jde o soubor procesů nad projektovými informacemi a souvisejícími daty v obecné formě. Příklad PLM strojírenského, nebo příbuzného produktu je znázorněn na obr. č. 1. Typickým atributem vývoje reálného výrobku je fáze vývoje, kde výstupem je soubor CAX dat, společně s dalšími souvisejícími daty a informacemi. Obecně produktem nemusí být reálný výrobek. Může jít např. o software, dokument, nebo proces.



Obrázek 1: Základní fáze PLM

PLM současně integruje a propojuje nástroje, používané pro zpracování projektových dat v manuální i digitální formě a řízení veškerých informací. Propojení systémů umožňuje v každé



z fází životního cyklu pracovat s relevantními a aktuálními daty. Aplikování metodiky PLM širšího rozsahu je znázorněno na obr. č. 2.

Obrázek 2: Rozšířené znázornění PLM a souvisejících procesů. Zdroj: Fořt, 2006

PLM systém

Vymezení a efektivní aplikování metodiky PLM v kontextu průmyslových standardů, představujících v současné době vysokou úroveň digitalizace, vytváří východisko pro návrh, tvorbu a integrování spektra nástrojů kategorie informačních technologií. Standardně jde o zpracování velkého množství souvisejících dat různých formátů, vytvořených v širokém spektru stávajících, nebo nově sestavených aplikací. Propojení relevantních dat v kontextu propojení jednotlivých fází PLM dle kapitoly 1 a obr. 1 umožňuje přesné definování a řízení všech aktivit v rámci procesní struktury životního cyklu produktu. Přístup k datům produktu je řízen s využitím obvyklých databázových metod. Na datech jsou současně definovány příslušné procesy, zahrnující například automatické a manuální schvalování příslušných statutů, zobrazování textových a grafických atributů, nebo vytváření reportů. Důraz je kladen na zabezpečení dat. Součástí zabezpečení dat je řízení oprávnění přístupu a modifikace dat, definování vlastnictví a dostupnost historie úprav jednotlivých souborů – datasetů. Příklady nejrozšířenějších PLM systémů, nasazených v aktuální průmyslové praxi jsou např. Windchill, 3DEXperience, nebo Teamcenter. Atributy PLM systému lze přiřadit celé řadě dalších nástrojů, konfigurovatelných pro správu dat a řízení procesů. Významnou charakteristikou uvedených nejrozšířenějších PLM systémů je integrace CAx digitálních dat virtuálních prototypů v rozsahu jejich specifik. Následující postupy jsou znázorněny s využitím CAx dat systému Siemens NX a Teamcenter Share. Očekávaný přínos nasazení systémů PLM spočívá všeobecně ve zvýšení kvality a efektivity řízení životního cyklu produktu ve všech jeho fázích. Historie definování a nasazení PLM začíná v období 80. let minulého století, kdy pro zvýšení efektivity a kvality řízení životního cyklu výrobku manuální formou bylo nutné aplikovat počítačové nástroje. Předchozí generaci systémů PLM představovaly systémy PDM – Product Data Management, představující databázovou správu dat. Významný nástup CAD a následně rozšířených CAx nástrojů definoval specifické požadavky na správu a řízení příslušných dat. V období konce 90. let je metodika PDM rozšířena a postupně nahrazována metodikou PLM. Rozšíření se týká především možností definování širokého spektra procesů na datech produktu. Aktuálně je PLM systém vnímán jako strategická platforma propojení a řízení dat a procesů v rámci celého životního cyklu produktu.

Integrovaní CAx dat v systému PLM

CAx data v životním cyklu výrobku procházejí fázemi od prvotní iniciace vzniku a mohou být i součástí procesu ukončení životnosti, nebo podkladem pro inovaci stávajícího produktu. Fáze životního cyklu jsou znázorněny na obr. č. 1. Největší objem řízení CAx dat zahrnují fáze vývoje a technické přípravy výroby produktu. V uvedených fázích jsou tvořeny virtuální prototypy, představované primárně CAD daty. CAD data jsou východiskem následných procesů technické přípravy výroby. Zpětná vazba provozní fáze servisu je základem pro změnové řízení a úpravy stávajícího technického řešení. Typy a návaznosti dat s přístupem Master-Model-Koncept jsou znázorněny na obr. č. 3.

Základní datovou položkou je Master Model, představovaný obvykle 3D CAD digitálním modelem. Master Model je výchozím informačním zdrojem pro tvorbu 2D, nebo 3D výkresové

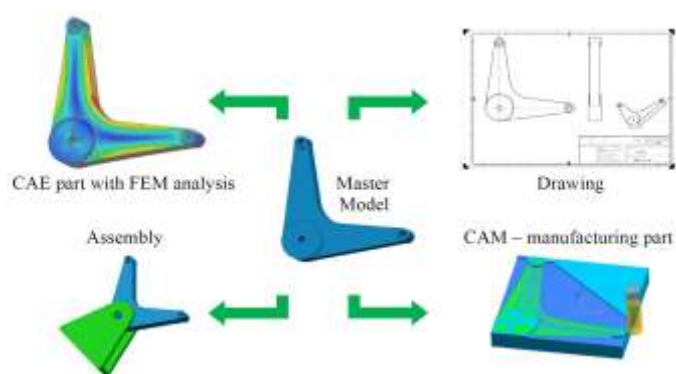
dokumentace. Principem Master-Model-Konceptu je oddělený model od výkresu. Z hlediska struktury a hierarchie dat je Master Model komponentou v nadřazené sestavě, představované výkresem. Oddělení modelu od výkresu poskytuje následující možnosti práce s daty:

- Efektivní řízení zobrazitelnosti modelových objektů na jednom, nebo více výkresových listech.
- Možnost oddělené práce na modelu a výkresu ve stejném čase.
- Přehledné řízení revizí – změn jednotlivých datasetů.

Stejně, jako v případě konceptu Model-Výkres lze uvažovat koncept Model-Vyšší sestava, Model-Simulace, Model-Technologie. Ve všech případech jde o soubor asociativně propojených dat.

Na rozdíl od File-systému, kdy jsou data uložena na lokálním, nebo sdíleném diskovém úložišti, jsou data v PLM systému trvale propojena. Oddělení dat je řízené, na rozdíl od File-Systému, kdy je vyšší riziko úmyslného, nebo neúmyslného narušení integrity.

Výhodou zpracování CAx dat prostřednictvím systému PLM je zpřístupnění celého spektra informací i dalším uživatelům, zapojeným do řešení konkrétního projektu, případně souvisejících projektů.



Obrázek 3: Vizualizace Master-Model Koncepce. Zdroj: Dvořák, 2023

Případová studie řízení projektu v Teamcenter Share

Typickou úlohou s podporou řízení dat s využitím PLM systému je vývoj a technická příprava výroby na úrovni sestavy. Sestava představuje produkt spojený z více součástí. Dle složitosti sestavy lze uvažovat více úrovní stromové struktury. Jednotlivé úrovně mohou zahrnovat jednotlivé součásti, případně pod-sestavy s navazující strukturou.

Generický projekt, obsahující Master a Non-Master data představuje typický postup řízení dat dle přístupu, popsaného výše. Základem je Master Model páky pro přenos síly, která je vytvořena obráběním z polotovaru. Součástí komplexních dat jsou výstupy simulace funkčního zatížení součástí. Dokumentace zahrnuje výpočtovou zprávu, technologické instrukce a další relevantní informace v návaznosti na vývojová a technologická data.



Název	Akte	Vlastník	Datum úpravy	Datum vytvoření
C4u_Data		Karel Dvorák	Dec 17, 2025, 5:04 PM	Dec 17, 2025, 5:04 PM
Documentation		Karel Dvorák	Dec 17, 2025, 5:04 PM	Dec 17, 2025, 5:04 PM

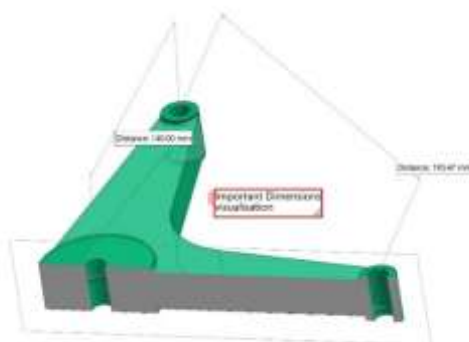
Obrázek 4: Vizualizace prostředí projektu.

Jednotlivé složky obsahují odpovídající datasety. Náhled složky s CAX daty a datasety je na obr. č. 5.

The screenshot displays the 'Master Model Concept' application. At the top, there is a navigation bar with icons for Home, Add, Edit, Delete, and a search icon. Below the navigation bar, there is a header section with the text 'Page: 1/1' and 'Page: 1/1'. The main content area shows a table with columns: 'Model', 'Kategori', 'Status', 'Tanggal', 'Jumlah', 'Jumlah', 'Jumlah', 'Jumlah', 'Jumlah', 'Jumlah', and 'Tipe'. The table lists several models, including 'Model 1', 'Model 2', 'Model 3', 'Model 4', 'Model 5', 'Model 6', 'Model 7', 'Model 8', 'Model 9', 'Model 10', 'Model 11', 'Model 12', 'Model 13', 'Model 14', 'Model 15', 'Model 16', 'Model 17', 'Model 18', 'Model 19', 'Model 20', 'Model 21', 'Model 22', 'Model 23', 'Model 24', 'Model 25', 'Model 26', 'Model 27', 'Model 28', 'Model 29', 'Model 30', 'Model 31', 'Model 32', 'Model 33', 'Model 34', 'Model 35', 'Model 36', 'Model 37', 'Model 38', 'Model 39', 'Model 40', 'Model 41', 'Model 42', 'Model 43', 'Model 44', 'Model 45', 'Model 46', 'Model 47', 'Model 48', 'Model 49', 'Model 50', 'Model 51', 'Model 52', 'Model 53', 'Model 54', 'Model 55', 'Model 56', 'Model 57', 'Model 58', 'Model 59', 'Model 60', 'Model 61', 'Model 62', 'Model 63', 'Model 64', 'Model 65', 'Model 66', 'Model 67', 'Model 68', 'Model 69', 'Model 70', 'Model 71', 'Model 72', 'Model 73', 'Model 74', 'Model 75', 'Model 76', 'Model 77', 'Model 78', 'Model 79', 'Model 80', 'Model 81', 'Model 82', 'Model 83', 'Model 84', 'Model 85', 'Model 86', 'Model 87', 'Model 88', 'Model 89', 'Model 90', 'Model 91', 'Model 92', 'Model 93', 'Model 94', 'Model 95', 'Model 96', 'Model 97', 'Model 98', 'Model 99', 'Model 100'. The table also includes columns for 'Jumlah' and 'Tipe'.

Obrázek 5: CAX datasety a asociované soubory.

Příklad vizualizace a možnosti přiřazení specifických atributů 3D modelu v prostředí TC Share je na obr. č. 6.



Obrázek 6: Vizualizace 3D modelu s možností definování relevantních atributů.

Založení projektu v TC Share představuje vytvoření položky projektu s příslušnými atributy řízení projektu, spočívající v deklaraci termínu finalizace projektu a relevantního popisu projektu. Vizualizace prostředí projektu v Teamcenter Share je na obr. č. 4. Současně jsou znázorněny relevantní složky projektu, obsahující příslušné datasety. Rozdělení a náplň složek je libovolná.

Závěr

Řízení dat s využitím systému PLM lze považovat za významný krok zvýšení kvalitativní úrovně procesu. Výhoda je zřejmá při porovnání s přístupem klasického souborového systému, představujícího uložení dat na síťovém, nebo lokálním úložišti. Jednoznačnost využití dat

a zdokumentování aktivit na nich prováděných zajišťuje snížení chybovosti a efektivní komunikaci mezi členy projektového týmu, případně na manažerské úrovni. Za určitou nevýhodu v porovnání se souborovým diskovým přístupem lze považovat určitou úroveň snížení rychlosti přístupu k datům, což může být zřetelné zejména při práci na velkých CAD sestavách. Snížení rychlosti přenosu dat je v porovnání s bezpečností a spolehlivostí řízení komplexních dat nevýznamné. Představenou metodiku lze použít pro řízení projektových dat libovolných oblastí, např. softwarových a podobných projektů.

Literatura

- Dekhtiar, J., Durupt, A., Bricogne, M., Eynard, B., Rowson, H., & Kiritsis, D. (2018). Deep learning for big data applications in CAD and PLM—Research review, opportunities and case study. *Computers in Industry*, 100, 227-243.
- Dvořák, K. (2023) CAX/PLM SYSTÉMY NÁSTROJEM KOMUNIKACE. In *Sborník příspěvků z konference.Informatika: VŠPJ*
- Fořt, P., & Kletečka, J. (2006). *AutoCad 2006*: učebnice. Computer Press.
- Jun, H. B., Kiritsis, D., & Xirouchakis, P. (2007). Research issues on closed-loop PLM. *Computers in industry*, 58(8-9), 855-868.
- Mas, F., Arista, R., Oliva, M., Hiebert, B., Gilkerson, I., & Rios, J. (2015). A review of PLM impact on US and EU aerospace industry. *Procedia engineering*, 132(10.1016).
- Schuh, G., Rozenfeld, H., Assmus, D., & Zancul, E. (2008). Process oriented framework to support PLM implementation. *Computers in industry*, 59(2-3), 210-218.
- Stark, J. (2022). Product lifecycle management (PLM). In *Product lifecycle management (volume 1) 21st Century paradigm for product realisation* (pp. 1-32). Cham: Springer International Publishing.

Kontaktní údaje

Ing. Bc. Karel Dvořák, Ph.D., Ing. Bc. Jana Dvořáková, Ph.D.
Vysoká škola polytechnická Jihlava
Tolstého 16, 586 01, Jihlava
e-mail: karel.dvorak@vspj.cz; jana.dvorakova@vspj.cz

MODERNÍ DIGITÁLNÍ TECHNOLOGIE VE VZDĚLÁVÁNÍ

MODERN DIGITAL TECHNOLOGIES IN EDUCATION

Miroslav Hrubý

Abstrakt

Příspěvek je zaměřen na problematiku aplikace moderních informačních technologií ve vzdělávání, a to včetně jejich možných negativních dopadů na účastníky takového vzdělávání. Hlavním cílem autora je podnítit diskusi na téma, jak reagovat na nežádoucí jevy a změny v mozkové tkáni, které registrují výzkumná pracoviště pomocí zobrazovacích technologií jako je funkční magnetická rezonance (fMRI). Je položen důraz na potřebu vyváženého používání technologií, ochranu duševního zdraví a prevenci závislostního chování.

Klíčová slova: digitalizace, vzdělávání, závislosti

Abstract

The article focuses on the issue of the application of modern information technologies in education, including their possible negative impacts on participants in such education. The author's main goal is to stimulate a discussion on how to respond to undesirable phenomena and changes in brain tissue that are registered by research centers using imaging technologies such as functional magnetic resonance imaging (fMRI). Emphasis is placed on the need for balanced use of technologies, protection of mental health and prevention of addictive behavior.

Keywords: digitalization, education, addictions

Úvod

Lidé vstupují do života s genetickou výbavou po svých předcích a zejména do své dospělosti dochází k formování neuronových sítí v jejich mozcích s výrazným podílem formálního i neformálního vzdělávání. V současné době do vzdělávání vstupují moderní digitální technologie jako AI, chytré telefony, Internet a sociální sítě. Vlivem dosaženého technického rozvoje soudobé společnosti je současně umožněno provádět výzkum v oblasti vlivu používání moderních digitálních technologií na lidský mozek. Moderní zobrazovací metody používané při vyšetření mozku poskytují zajímavé poznatky, na které by měla společenská praxe nejen v oblasti vzdělávání reagovat.

Shrnutí dosavadních poznatků

Nadužíváním digitálních technologií rozumíme nadměrném trávení času na internetu, a to zejména na sociálních sítích a hraním digitálních her. Tento jev může vést ke vzniku a rozvoji závislostí, které se souhrnně označují pojmem digitální závislosti. Digitální závislosti se projevují u dětí, dospívajících i dospělých a jsou spojeny se zdravotními a sociálními dopady.

Hráčská porucha, tzv. gaming disorder, je zahrnuta v Mezinárodní klasifikaci nemocí. V ČR byla v roce 2022 vydána k problematice digitálních závislostí první souhrnná zpráva. Zatím poslední danou publikací je Zpráva o digitálních závislostech v České republice 2024 (Chominová et al, 2024).

Mezi aktuální otázky v oblasti vzdělávání nejen v ČR lze nesporně zařadit: „Patří mobilní telefony do škol?“. Pokud mobilní telefon není součástí výuky, vede jeho přítomnost zejména k rozptylování pozornosti (Schmitgen et al., 2025; Techzpravy.cz, 2025).

Francie, jedna z ekonomicky nejsilnějších zemí Evropy, zakázala používání mobilních telefonů na základních a středních školách s výjimkou výukových účelů již od roku 2018. Od roku 2021 mohou žáci v Číně mít ve škole mobilní telefon pouze v případě, že to výslovně povolí rodiče a škola. Ve stádiu přípravy je formulace celoevropského zákazu, pro což existuje více důvodů, a to zejména:

- docílit lepší soustředění a koncentraci pozornosti na výuku, vyšší kvalitu výuky
- podpořit prevenci závislosti na technologiích
- podpořit mezilidskou osobní komunikaci a rozvíjet sociální dovednosti
- snížit stres odstraněním tlaku na reagování na přicházející zprávy
- eliminovat sociální nerovnosti.

AI ve vzdělávání je řešena na území Evropské unie v několika klíčových dokumentech. Jedná se zejména o:

- Akční plán digitálního vzdělávání 2021–2027
- Usnesení Evropského parlamentu o umělé inteligenci ve vzdělávání, kultuře a audiovizuálním odvětví (19. 5. 2021)
- Evropský prostor vzdělávání do roku 2025
- Akt o umělé inteligenci (AI Act).

Na druhé straně digitalizace a moderní digitální technologie ve vzdělávání přinášejí nová rizika závislosti, problematického užívání a zdravotních problémů. Existující nebezpečí se výrazně liší s ohledem na různé věkové skupiny. Riziko je nejvyšší v adolescenci ale rovněž platí, že čím mladší je daná osoba, tím vyšší riziko jí s ohledem na vývoj a formování neuronových sítí v mozku hrozí. Neurologové i další odborníci zkoumají vliv obrazovek na kognitivní funkce, spánek a duševní zdraví.

Data a metody

V současné době jsou k dispozici vědecké studie, které pomocí funkční magnetické rezonance (fMRI – functional magnetic resonance imaging) a dalších metod (EEG – electroencephalography, MEG – magnetoencephalography, PET – positron emission tomography) ukázaly, že u osob (zejména dospívajících) s diagnostikovanou závislostí na internetu a sociálních sítích lze pozorovat měřitelné změny v mozkové činnosti a struktuře (Coleman, 2024; Chang and Lee, 2024).

Jedná se o oslabení propojení mezi oblastmi mozku zodpovědnými za paměť, rozhodování a aktivní myšlení, což je označováno jako snížená funkční konektivita. Dochází rovněž k chemickým změnám v neuronových sítích, které jsou na mozku detekovatelné

zobrazovacími metodami. Tyto změny souvisejí s horší sebekontrolou a silnějším návykovým chováním. Některé části mozku vykazují zvýšenou aktivitu při očekávání digitální odměny a jiné naopak utlumení. Zejména pro věkovou kohortu 10-19 let se projevují negativní dopady na vývoj intelektuálních schopností, snížená fyzická koordinace, narušené duševní zdraví vedoucí k úzkostem, depresím a v krajním případě i sebevraždám.

Výsledky a diskuze

Nejvýraznější nárůst využití digitálních technologií v ČR nepochybně nastal během pandemie covidu-19, kdy došlo k uzavření škol a výuka probíhala on-line. V tomto období se rovněž sociální kontakty realizovaly zejména s využitím sociálních sítí.

Od roku 2010 přibýlo o více než 70 procent dětí a mladistvých v péči dětského psychiatra (Růžička, 2025), přičemž v Praze se počet psychiatrických pacientů do 19 let mezi lety 2013 a 2023 téměř zdvojnásobil, a to na více než 8 tisíc (zpravy.kurzy.cz, 2024). U mladých ve věku 14–25 let je sebevražda druhou nejčastější příčinou úmrtí (medium.cz, 2025).

Neurolog Stránský (Český rozhlas Radiožurnál, 2019; Cukrfree, 2024; DVTV, 2024; First Class, 2024; Praha TV, 2025) uvádí:

- pro děti do dvou let absolutně žádná obrazovka
- od dvou do pěti let maximálně hodinu denně, a to pouze se supervizí
- od pěti do 15 let maximálně dvě hodiny denně se supervizí.

Čím dříve dítě dostane chytrý telefon a čím více času tráví na sociálních sítích, tím horší výsledky prokazuje ve standardizovaných testech. Sociální sítě děti „stupidifikují“, to je IQ přestává růst, dítě ztrácí slovní zásobu, schopnost řešit problémy, předvídat věci, formulovat psaný text, určité struktury mozku se zmenšují až mizí.

Mozek dítěte se musí vyvinout normální přirozenou cestou a neoptimálnější cestou. Musí umět řešit problémy, nacházet a zvažovat alternativní řešení, užívat mezilidské kontakty. Sociální sítě jsou digitální heroin, byly a jsou vytvářeny přes neuromarketing. Stejně jako droga jsou návykové, protože zvyšují hladinu dopaminu, to je látky v mozku, jejíž množství stoupá s mírou slasti a očekávání.

Velmi zajímavé je zjištění finského výzkumného týmu o „nakažlivosti“ psychických problémů v kolektivu adolescentů, které zahrnovalo 700 000 žáků (Alho et al., 2024). Pokud se ve třídě objeví dítě s psychickým problémem, stoupne o 9 % pravděpodobnost, že podobnou poruchou začnou trpět i jeho spolužáci. Pokud je ve třídě víc než jedno dítě s psychickými problémy, tato pravděpodobnost stoupá už na 18 %. Tato studie by neměla být opomenuta při předpokládaném přehodnocení inkluze v českém vzdělávacím systému.

Závěr

Moderní vzdělávací technologie mohou vzdělávání činit atraktivnějším a zvyšovat jeho účinnost, ale je třeba mít na paměti nutnost zachovat proporce mezi jejich využitím a novými poznatky z oblasti zdravotních rizik spojenými s jejich nadměrným zapojením. Výsledky výzkumu v oblasti vlivu mobilních telefonů, sociálních sítí a dalších digitálních technologií a nástrojů na lidský mozek nelze přehlížet. Pokud budou stávající tendence negativních vlivů

rychle se rozvíjejících technologií na lidský mozek dalším výzkumem potvrzeny, je nejvyšší čas na diskusi o adekvátní reakci vzdělávacího systému ČR.

Poděkování

Příspěvek vznikl za podpory dílčího záměru rozvoje organizace Kybernetické síly a prostředky (DZRO KYBERSÍLY).

Literatura

Alho, J., Gutvilig, M., Niemi, R., et al. (2024). Transmission of Mental Disorders in Adolescent Peer Networks. *JAMA Psychiatry*, 81 (9), 882-888. [online], [2025-11-27]. Dostupné z: <https://jamanetwork.com/journals/jamapsychiatry/fullarticle/2818735>

Coleman, F. (2024). Internet addiction: What is it doing to teen brains? *StudyFinds*. [online], [2025-11-27]. Dostupné z: <https://studyfinds.org/internet-addiction-what-its-doing-to-teen-brains/>

Cukrfree (2024). Cukrfree Podcast #86: Martin Jan Stránský - Mozek a technologie. [online], [2025-11-27]. Dostupné z: <https://www.youtube.com/watch?v=bDNPFAXZNQO>

Český rozhlas Radiožurnál (2019). Martin Jan Stránský: Pozor, IQ dětí se snižuje. A mohou za to sociální sítě. [online], [2025-11-27]. Dostupné z: <https://www.youtube.com/watch?v=aoSsbWtDVgg>

DVTV (2024). Mobil je pro dítě jako lahev vodky, IQ nové generace padá. Evolučně potřebujeme drastický reset. [online], [2025-11-27]. Dostupné z: <https://www.youtube.com/watch?v=Nj5q35Xx2yk>

First Class (2024). Vzestup a pád lidské mysli | MUDr. Stránský v nové knize varuje před hloupostem lidstva. [online], [2025-11-27]. Dostupné z: https://www.youtube.com/watch?v=pLLVUUVLI_fI

Chang, M. L. Y., Lee, I. O. (2024). Functional connectivity changes in the brain of adolescents with internet addiction: A systematic literature review of imaging studies. *PLOS Mental Health*, 1(1): e0000022. [online], [2025-11-27]. DOI: <https://doi.org/10.1371/journal.pmen.0000022>

Chomynová, P., Dvořáková, Z., Černíková, T., Orlíková, B., Grohmannová, K., Franková, E., Roubalová, M., Galandák, D. (2024). Zpráva o digitálních závislostech v České republice 2024. Praha: Úřad vlády České republiky. [online], [2025-11-27]. Dostupné z: https://www.drogy-info.cz/data/obj_files/35641/1315/Zpr%C3%A1va%20o%20digit%C3%A1ln%C3%ADch%20z%C3%A1vislostech%20v%20C4%8CR%202024_web.pdf

médium.cz (2025). Generace Z v psychické krizi: Proč prudce přibývá duševních onemocnění mezi mladými Čechy? [online], [2025-11-27]. Dostupné z: <https://medium.seznam.cz/clanek/jeden-blazen-generace-z-v-psychicke-krizi-proc-prudce-pribyva-dusevnich-onemocneni-mezi-mladymi-cechy-144138>

Praha TV (2025) Neurolog Stránský: Mozek na drogách vs. mozek na sociálních sítích? Rozdíl nepoznáte. [online], [2025-11-27]. Dostupné z: <https://www.youtube.com/watch?v=moonAgHL7h4>

- Růžička, A. (2025). Alarmující stav. Duševní problémy mladých rostou. Sužuje je samota, hlavně dívky. [online], [2025-11-27]. Dostupné z: <https://zpravy.aktualne.cz/domaci/dramaticke-globalni-zhorseni-dusevniho-zdravi/r~fe2cfd5af44611efbf960cc47ab5f122/>
- Schmitgen, [M. M.](#), Henemann, G. M., Koenig, [J.](#), Otte, M-L., Rosero J. P., Bach, [P.](#), Haage, S. H., Wolf, [N. D.](#), Wolf R. C. (2025). Effects of smartphone restriction on cue-related neural activity, Computer in Human Behavior, 167, June 2025, 108610. [online], [2025-11-27]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S0747563225000573?via%3Dihub#bfn1>
- Techzpravy.cz (2025). Finské školy sebraly dětem mobily. A toto je výsledek. [online], [2025-11-27]. Dostupné z: https://www.techzpravy.cz/finske-skoly-sebraly-detem-mobily-a-toto-je-vysledek/?utm_source=msn&utm_medium=newsfeed&utm_campaign=video#google_vignette
- YouTube (2025). Neurovědce varuje: Co ničí váš mozek? A co mu naopak prospívá? Aleš Stuchlík. [online], [2025-11-27]. Dostupné z: <https://www.youtube.com/watch?v=P570PTzkaNk>
- zpravy.kurzy.cz (2024). Počet dětí s psychickým problémy dramaticky roste, Praha problém řeší. [online], [2025-11-27]. Dostupné z: <https://zpravy.kurzy.cz/789389-pocet-deti-s-psychickym-problemy-dramaticky-roste-praha-problem-resi/>

Kontaktní údaje

Ing. Miroslav Hrubý, CSc.
Univerzita obrany, K-209
Kounicova 65, 66210 Brno
e-mail: miroslav.hruby@unob.cz

DIGITAL IS PHYSICAL – VÝUKA PROGRAMÁTORŮ S OHLEDEM NA EKOLOGICKOU STOPU

DIGITAL IS PHYSICAL: EDUCATING PROGRAMMERS TO RECOGNIZE THEIR
ECOLOGICAL FOOTPRINT

Lenka Kosková Třísková, Igor Kopetschke

Abstrakt

Digitální svět bývá často vnímán jako nehmotný a oddělený od fyzické reality. Ve skutečnosti má však každá softwarová činnost – od psaní kódu po provoz cloudových služeb – svou ekologickou stopu. Spotřeba energie, životnost hardware i efektivita softwaru jsou úzce propojeny. Tento příspěvek se zabývá otázkou, jak tyto souvislosti zpřístupnit studentům informatiky a programování, aby si již během studia osvojili zásady udržitelného digitálního designu. Na základě zkušeností z výuky předmětů zaměřených na embedded systémy, operační systémy a efektivní programování představujeme přístup „Digital is physical“, který propojuje programátorské myšlení s uvědoměním fyzických důsledků digitálních rozhodnutí. Studenti se seznamují s principy energetické efektivity, optimalizace kódu, prodlužování životního cyklu zařízení a odpovědného využívání výpočetních zdrojů. Cílem je rozvíjet technické kompetence společně s environmentálním uvědoměním, a tím připravovat novou generaci vývojářů, kteří chápou, že udržitelnost softwaru je stejně důležitá jako jeho funkčnost.

Klíčová slova: Green software, e-waste, best practices, sustainable embedded software

Abstract

The digital world is often perceived as intangible and detached from physical reality. In practice, however, every software-related activity—from writing code to operating cloud services—has an environmental footprint. Energy consumption, hardware lifetime, and software efficiency are closely interconnected. This paper addresses the question of how these relationships can be made accessible to students of computer science and programming, so that they adopt principles of sustainable digital design already during their studies. Based on teaching experience in courses focused on embedded systems, operating systems, and efficient programming, we present the “Digital is Physical” approach, which links programming mindset with an awareness of the physical consequences of digital decisions. Students are introduced to principles of energy efficiency, code optimization, extending device lifetime, and responsible use of computing resources. The goal is to develop technical competencies together with environmental awareness, and thus to prepare a new generation of developers who understand that software sustainability is as important as software functionality.

Klíčová slova: Green software, e-waste, best practices, sustainable embedded software

Úvod

Co je “green” nebo “sustainable” software?

Zelené (udržitelné) IT a návrh softwaru se zaměřuje na environmentální dopady, energetickou účinnost a celkové náklady na vlastnictví včetně nákladů na likvidaci a recyklaci (Murugesan 2008). Většina publikovaných postupů se soustředí na spotřebu energie a uhlíkovou stopu zejména u serverových systémů. Podle International Energy Agency se spotřeba elektřiny datovými centry odhaduje na ~415 TWh, což v roce 2024 představovalo 1,5 % celosvětové spotřeby elektřiny, a očekává se její výrazný růst (IEA 2025). Do roku 2040 se odhaduje podíl ICT na celkové karbonové stopě až 14 % (Belkhir and Elmeligi 2018).

Rostoucí nároky na energii vedly ke vzniku hnutí “green software”. V roce 2021 společnosti Accenture, GitHub, Microsoft a ThoughtWorks založily pod záštitou Linux Foundation nadaci Green Software Foundation (GSF) s cílem budovat ekosystém standardů, nástrojů a osvědčených postupů (GSF citace). GSF kormě vzdělávání definuje metriky pro měření emisí CO₂, mj. metodiku Software Carbon Intensity (SCI), která byla vydána jako ISO/IEC 21031:2024 (Green Software Foundation 2025; ISO/IEC 2024).

Doporučení green software engineering proto často cílí na minimalizaci emisí při běhu (např. časování výpočetně náročných operací podle dostupnosti nízkoemisních zdrojů) a na návrh aplikační logiky, UX či DevOps procesů (Fontanarrosa 2024; Microsoft 2025; Ailane et al. 2025). Podstatně méně pozornosti se věnuje redukci e-waste.

EECONE – European ECOSystem for green Electronics

EECONE, projekt financovaný v rámci CHIPS-JU, si klade za cíl snížit množství elektronického odpadu v Evropě prostřednictvím nových nástrojů, osvědčených postupů a budování evropského ekosystému. Výzkum probíhá v letech 2023–2026; zapojeny jsou desítky firem a výzkumných organizací z 16 zemí.

Rámcem projektu je princip šesti “pilířů” neboli 6R: Reduce, Reliability, Repair, Reuse, Refurbish, Recycle. Technická univerzita v Liberci se soustředí na výzkum a vývoj doporučení a “best practices” pro návrh embedded softwaru tak, aby umožňoval praktickou aplikaci 6R. TUL odpovídá také za převod výstupů tohoto výzkumu do vzdělávání budoucích inženýrů.

E-waste a EECONE

E-waste (WEEE) zahrnuje elektrická a elektronická zařízení vyřazená bez úmyslu dalšího použití (Global E-waste Statistics Partnership 2025). Podle WHO jde o nejrychleji rostoucí typ odpadu; v roce 2022 vzniklo 62 milionů tun odpadu, ale prokazatelně recyklováno bylo jen 22,3 % (WHO 2025). Nízká recyklace znamená únik nebezpečných látek do prostředí a zároveň ztrátu vzácných materiálů (WHO 2025). Výroba elektroniky má navíc významnou uhlíkovou stopu, takže předčasné vyřazování funkčních zařízení přispívá i k emisím CO₂ (Belkhir and Elmeligi 2018).

Náš výzkum analyzoval jednotlivá „R“ z rámce projektu a sestavil doporučení, jak je lze podpořit návrhem embedded softwaru. Z analýzy jsme vyřadili Recycle, protože probíhá až po ukončení provozu zařízení. Refurbish je typicky realizován novým softwarem; což vyžaduje bezpečný update a změnu funkcí, které řešíme pro jiná R, a proto ani Refurbish není zahrnut do analýzy.

Reduce

Snížování materiálu je primárně hardwarová záležitost, přesto může software nepřímo přispět: klíčovým zdrojem je baterie. Energeticky efektivní software prodlužuje životnost baterie a snižuje počet použitých baterií během života zařízení. Z hlediska optimalizace spotřeby energie to v oblasti embedded software zejména znamená zvolit:

1. **Vhodnou architekturu** (event-driven/stavový model, minimalizace aktivního čekání),
2. **Vhodný framework/RTOS**, který úsporné režimy skutečně podporuje (na rozdíl od velmi jednoduchých smyčkových přístupů typu „Arduino styl“).

Reliability

Spolehlivost a dlouhá životnost výrobku lze výrazně podpořit softwarově. Klíčové jsou:

1. **Diagnostika a monitorování** zařízení za běhu (včasná detekce chyb, podpora servisu).
2. **Dlouhodobě reprodukovatelné sestavení firmware** (uchování konfigurace, verzí nástrojů a knihoven).
3. **Udržitelný systém pro release a verzování**, včetně identifikace komponent (např. možnost generovat SBOM; citace SBOM).
4. **Možnost bezpečných aktualizací** pro opravy a dlouhodobou údržbu (integrita, důvěryhodnost).

Reuse

Reuse lze přirovnat k „digitálnímu vrakovišti“: původní výrobek už neplní svůj účel, ale některé moduly lze znovu použít. Softwarové požadavky jsou podobné jako u Repair (modularita, rozhraní, přenositelnost konfigurace).

Repair

Repair předpokládá modulární HW/SW architekturu a počítá s tím, že během dlouhé životnosti mohou být použity jiné než původní součásti. Software proto musí umožnit změny konfigurace bez zásahu do aplikační logiky.

To vyžaduje:

1. **Modulární architekturu** s jasně definovaným a stabilním rozhraním (API).
2. **Oddělení HW konfigurace od kódu a abstrakci HW.**
3. **Možnost aktualizace SW v zařízení** (přes fyzické rozhraní nebo OTA).
4. **Ověření integrity a původu aktualizace** (kontrolní součty, digitální podpisy).

Obecná doporučení a kontrolní seznam (checklist)

Na základě analýzy jsme vytvořili obecný kontrolní seznam (checklist), který umožňuje kvalitativně posoudit, zda návrh embedded softwaru podporuje cíle 6R. V projektu dále připravujeme detailnější checklisty pro Zephyr a Yocto Project a diskutujeme jejich začlenění do dokumentace k těmto projektům. V tomto článku uvádíme obecnou verzi checklistu, protože byla prakticky ověřena ve výuce; ponecháváme ji v angličtině kvůli terminologické konzistenci.

Tabulka 1: Obecný checklist pro návrh embedded SW v souladu s principy EECONE

	Questions	Yes/No, comments
1. Reduce		
1.1	Does the software consistently use low-power sleep modes whenever possible?	
1.2	Are busy loops avoided (such as event-driven design instead of infinite active loops)?	
1.3	Are peripherals and radios disabled when not in use?	
2. Reliability		
2.1	Is the software modular?	
2.2	Is the API designed to keep at least ABI stable?	
2.3	Are diagnostics implemented (logging, health monitoring, watchdogs)?	
2.4	Is the build reproducible in the future (versioned dependencies, documented build process)?	
2.5	Does the architecture avoid vendor lock-in (use of open standards and portable tools)?	
2.6	Are secure updates supported (signed updates, vulnerability patching, rollback/versioning)?	
3. Reuse		
3.1	Can software modules be reused in other devices or projects?	
3.2	Are interfaces clearly documented to support reuse?	
4. Repair		
4.1	Is hardware abstraction used?	
4.2	Can hardware components be replaced without modifying application code?	
4.4	Is the hardware configuration separated from application logic?	
4.5	Can the software be easily reconfigured and rebuilt when hardware is repaired or replaced?	
4.6	Is the system capable of updates and upgrades in the field?	

Pilotní projekt: Výuka na TUL

Osvědčené postupy byly zavedeny do kurzů pro studenty mechatroniky se specializací na návrh chytrých systémů. Implementace proběhla během tří po sobě jdoucích zimních semestrů. Obsah výuky se postupně měnil tak, jak pokračoval náš výzkum a spolupráce v rámci EECONE.

1. Zimní semestr 2023 a 2024 – povědomí o udržitelnosti a hodnocení

Studenti absolvovali přednášky popisující osvědčené postupy a perspektivu 6R (Reduce, Reliability, Repair, Reuse, Refurbish, Recycle – redukce, spolehlivost, oprava, opětovné použití, renovace, recyklace). Poté aplikovali kontrolní seznam na stávající semestrální projekty v oblasti IoT, na kterých již pracovali. Důraz byl kladen na reflexi: studenti se naučili

analyzovat své návrhy a identifikovat příležitosti ke zlepšení spolehlivosti, opravitelnosti nebo spotřeby energie.

2. Zimní semestr 2025 – udržitelnost integrovaná od samého začátku

Kurz byl přepracován tak, aby byla udržitelnost zohledněna již od prvních kroků návrhu. Studenti si vybrali nebo navrhli vlastní nápad na systém IoT a vyvinuli architekturu s kontrolním seznamem 6R jako vstupem pro návrh, nikoli pouze jako hodnotícím nástrojem.

Metodika výuky

Osvědčené postupy byly zavedeny do kurzů pro studenty mechatroniky se specializací na chytré systémy během tří zimních semestrů:

- **ZS 2023 a 2024** – povědomí a hodnocení: studenti absolvovali přednášky a aplikovali checklist na existující IoT projekty. Důraz byl na reflexi a identifikaci zlepšení (spolehlivost, opravitelnost, energie).
- **ZS 2025** – udržitelnost od začátku: studenti navrhli vlastní IoT systém a checklist použili jako vstup do návrhu, nikoliv jen evaluaci.

Při výuce kombinujeme e-learning (pojmy a principy), praktická cvičení vedená lektorem (procvičení, vedení) a projektové učení (samostatný návrh). Časová dotace je následující: 2× e-learning (60+60 min), 2× lab (2×90 min), projekt (6×90 min). Studenti pracovali v týmech o velikosti 2–5. Lektor působí jako konzultant, nikoliv jako zadavatel úkolů. V ZS 2025 jsme doporučili Zephyr RTOS jako moderní platformu vhodnou pro implementaci doporučených best practices.

Zpětná vazba od studentů

Studenti uváděli, že před kurzem si neuvědomovali rozsah problému e-waste. Checklist a rámec 6R jim pomohly pochopit, že volby v softwaru ovlivňují životnost zařízení, opravitelnost i spotřebu zdrojů. Ocenili propojení udržitelnosti s konkrétními technickými rozhodnutími a byli překvapeni, že software může životnost zařízení zkracovat i prodlužovat. Svoboda volby vlastního úkolu zvýšila motivaci; checklist označili za praktický nástroj pro “měření udržitelnosti během vývoje”. Mnozí uvedli, že budou principy používat i v dalších projektech.

Závěr

Ukazujeme, že udržitelnost lze systematicky začlenit do výuky návrhu embedded softwaru bez snížení technické hloubky. Převodem výstupů výzkumu EEONE do výuky získali studenti rámec 6R, konkrétní osvědčené postupy a checklist jako most mezi teorií a praxí. První výsledky potvrzují, že udržitelnost může být součástí návrhových požadavků, nikoli doplňkovým tématem. Budoucí práce zahrnuje dokončení veřejně přístupného MOOC a podporu širšího přijetí v komunitách projektů Zephyr a Yocto Project.

Poděkování

Projekt EEONE je financován iniciativou Chips JU a jejími členy, včetně doplňkového financování ze strany národních finančních orgánů zúčastněných zemí na základě grantové dohody č. 101112065.

Použitá literatura

Ailane, M.T.; Rubner, C.; Rausch, A. Green DevOps: A Strategic Framework for Sustainable Software Development. *Comput. Sci. Math. Forum* 2025, 10, 5. <https://doi.org/10.3390/cmsf2025010005>

Belkhir L., Elmeligi A.: Assessing ICT global emissions footprint: Trends to 2040 & recommendations, *Journal of Cleaner Production*, Volume 177, 2018, Pages 448-463, ISSN 0959-6526, <https://doi.org/10.1016/j.jclepro.2017.12.239>.

Electronic Waste (WHO), [2025-12-18], Dostupné z: [https://www.who.int/news-room/fact-sheets/detail/electronic-waste-\(e-waste\)](https://www.who.int/news-room/fact-sheets/detail/electronic-waste-(e-waste))

Fontanarrosa S.: *Green Software Engineering*, Packt Publishing, 2024, ISBN: 9781835885895

Green Software Foundation (Media Kit) (2025), online, [2025-12-16], Dostupné z: <https://wiki.greensoftware.foundation/resource-media-kit>

Green Software Foundation (Learn) (2025), online, [2025-12-16], Dostupné z: <https://movement.greensoftware.foundation/spaces/19525645/page>

Green Software Foundation (SCI) (2025), online, [2025-12-18], Dostupné z: <https://sci.greensoftware.foundation/>

IEA (2025), *Energy and AI*, IEA, Paris, [2025-12-17], Dostupné: <https://www.iea.org/reports/energy-and-ai>, Licence: CC BY 4.0

ISO/IEC 21031:2024, Dostupné z: <https://www.iso.org/standard/86612.html>

S. Murugesan, "Harnessing Green IT: Principles and Practices," in *IT Professional*, vol. 10, no. 1, pp. 24-33, Jan.-Feb. 2008, doi: 10.1109/MITP.2008.10.

Salam M., Khan S U.: "Developing green and sustainable software: Success factors for vendors," 2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2016, pp. 1059-1062, doi: 10.1109/ICSESS.2016.7883248.

Software Bill of Material – Definition, [2025-12-18], Dostupné z: <https://www.cisa.gov/sbom>

The Principles of Sustainable Software Engineering, [2025-12-18], Dostupné z: <https://learn.microsoft.com/en-us/training/modules/sustainable-software-engineering-overview/>

What is e-waste, [2025-12-18], Dostupné z: <https://globalewaste.org/what-is-e-waste/>

Kontaktní údaje

Ing. Lenka Kosková Třísková, Ph. D.

Technická univerzita v Liberci, Fakulta mechatroniky, informatiky a mezioborových studií
Studentská 2/1402, Liberec, 46331

Lenka.koskova.triskova@tul.cz

PŘÍPRAVA VÝUKY V TANDEMU – VYUŽITÍ LLM JAKO PARTNERA PŘI TVORBĚ INTERAKTIVNÍHO KURZU PROGRAMOVÁNÍ V JUPYTERLAB

COURSE DESIGN IN TANDEM – LEVERAGING LLMS AS A PARTNER IN CREATING AN INTERACTIVE PROGRAMMING COURSE IN JUPYTERLAB

Petr Kropík, Lenka Šroubová

Abstrakt

Příspěvek představuje metodiku přípravy a výuky kurzu Základy programování při jeho inovaci a modernizaci. Změna zahrnuje též přechod od jazyka C++ k jazyku Python. Jako hlavní platforma slouží JupyterLab s pluginy JupyterSlides a ipywidgets, umožňující tvorbu plně interaktivních materiálů, včetně spustitelných bloků kódu, grafů a dalších vizualizací výsledků vytvářených aplikací. Je popsán proces tvorby podkladů – od sylabu až po generování prezentací a příkladů, a to ve spolupráci s velkým jazykovým modelem (LLM), který fungoval jako partner v procesu přípravy. Cílem bylo zatraktivnit výuku, zvýšit zapojení studentů a efektivně rozvíjet praktické dovednosti v souladu s principy softwarového inženýrství.

Klíčová slova: Výuka programování, Python, JupyterLab, LLM, Softwarové inženýrství, Interaktivní výuka, AI gramotnost.

Abstract

The paper presents a methodology for preparing and teaching the Fundamentals of Programming course during its innovation and modernization. The shift involves transitioning from C++ to Python. JupyterLab serves as the main platform, utilizing plugins such as JupyterSlides and ipywidgets to enable the creation of fully interactive materials, including executable code blocks, graphs, and other visualizations of application results. The process of creating course materials is described – from the syllabus to the generation of presentations and examples – in cooperation with a Large Language Model (LLM), which acted as a partner in the preparation process. The aim was to make teaching more attractive, increase student engagement, and effectively develop practical skills in accordance with software engineering principles.

Keywords: Programming Education, Python, JupyterLab, LLM, Software Engineering, Interactive Teaching, AI Literacy.

Úvod

Výuka základů programování na technických univerzitách prochází dynamickým vývojem. Tradiční přístup, často postavený na jazyce C/C++, naráží u studentů prvních ročníků na vysokou bariéru vstupu danou složitou syntaxí a nutností správy paměti.

V rámci inovace předmětu pro studenty elektrotechniky byl zvolen přechod na jazyk Python. Hlavním cílem této změny nebylo pouze „naučit jiný jazyk“, ale umožnit studentům rychlejší adopci principů softwarového inženýrství – jako jsou objektově orientované programování (OOP), principy SOLID, testování a verzování kódu – aniž by byli zahlceni syntaktickými detaily.

Součástí inovace byla i změna formy výukových materiálů. Místo statických PDF prezentací byl zvolen ekosystém JupyterLab, který umožňuje spojit výkladový text, spustitelný kód a vizualizace do jednoho interaktivního dokumentu. Tento článek popisuje proces tvorby těchto materiálů, při kterém sehrál klíčovou roli dialog s velkým jazykovým modelem (LLM).

1 Metodika tandemové přípravy s LLM

Při tvorbě kurzu nebyl LLM (konkrétně modely rodiny Gemini a GPT) využíván jako pouhý generátor textu, ale jako kritický partner v diskusi. Proces lze charakterizovat jako iterativní dialog, kde pedagog vystupuje v roli architekta a garanta odbornosti, zatímco LLM plní roli juniorního kolegy s encyklopedickými znalostmi a schopností rychlého prototypování. Významnou pomocí je již samotný návrh struktury výuky, např. cvičení a jeho rozvržení (Obrázek 1)

Průběh cvičení:

1. (15 min) **Motivace a teorie:**
 - Navažte na příklad s evidencí součástek. "Seznam slovníků je fajn, ale co kdybychom chtěli, aby každá součástka 'uměla' něco sama? Třeba se hezky vypsát?"
 - Představte koncept třídy jako "šablony" a objektu jako "výrobku".
 - Na tabuli nebo v kódu ukažte základní syntaxi: `class Soucastka:`, `def __init__(self, ...):`, `self.atribut = ...`, `def nejaka_metoda(self):`. Vysvětlete, že `self` je odkaz na konkrétní objekt.
2. (25 min) **Praktická ukázka - refaktoring:**
 - Společně se studenty převeďte kód z minulého cvičení (evidenční součástek) z použití slovníků na použití třídy `Soucastka`.
 - Místo seznamu slovníků bude `seznam_objektu_soucastka`.
 - Ukažte, jak je kód najednou čitelnější: `soucastka.pocet_ki` místo `soucastka["pocet_ki"]`.
 - Přidejte metodu `__str__(self)` nebo `zobraz_info(self)`, aby se objekt uměl sám "reprezentovat".
3. (15 min) **Dědičnost:**
 - "Co když chceme evidovat specifické vlastnosti jen pro některé typy součástek?"
 - Vytvořte třídu `class Rezistor(Soucastka):`, která dědí od `Soucastka`.
 - Ukažte, jak pomocí `super().__init__(...)` zavolat konstruktor předka a přidat nový atribut, např. `self.odpor`.
 - Ukažte, jak přidat novou metodu specifickou jen pro rezistor, např. `vypocitej_stratovy_vykon(self, napeti)`.
4. (35 min) **Aplikace na semestrální projekt (Fáze 2):**
 - Studenti začnou pracovat na svém projektu.
 - Jejich úkolem je navrhnout a implementovat základní třídy pro jejich projekt.
 - **Pro hru:** `class Hrac:`, `class Nepritel:`, `class Predmet:`. Každá bude mít základní atributy (`jezmeno`, `zdravi`, ...) a konstruktor.
 - Vyučující prochází a pomáhá se správným návrhem.

Obrázek 1: Návrh struktury cvičení s časových rozvrhem generovaný LLM.

1.1 AI jako partner, ne autorita

Zásadním poznatkem z naší spolupráce je fakt, že není třeba se AI obávat, ale je nutné k ní přistupovat s odbornou skepsí. AI není autorita a její výstupy často obsahují subtilní chyby, halucinace nebo řešení, která jsou sice funkční, ale pedagogicky nevhodná.

Příkladem může být tvorba instalačního skriptu (`install.bat`) pro nastavení prostředí na studentských počítačích s OS Windows. LLM navrhl funkční skript, který však selhal při pokusu o reinstalaci virtuálního prostředí kvůli zamykání souborů běžícím procesem. Teprve

po zásahu pedagoga a specifikaci problému model navrhl pokročilé řešení s využitím příkazů (PUSHD %TEMP%), které problém eliminovalo. Tento případ ilustruje nutnost expertního dohledu – pedagog musí být schopen kód nejen přečíst, ale i kriticky zhodnotit jeho chování v okrajových podmínkách.



Obrázek 2: Ukázka dialogu s LLM – prompt pedagoga s požadavkem na vylepšení a rozšíření přednášek a odpověď LLM.

1.2 Přístup k použití LLM

Součástí filozofie inovovaného předmětu je otevřenost k moderním nástrojům. Místo zákazů používání AI, které se v akademické sféře často objevují, volíme cestu edukace. Studentům jsou přímo v rámci přednášek demonstrovány možnosti, ale i limity LLM.

Jako výukový materiál využíváme i autentické záznamy našich chatů s AI (Obrázek 2), které vznikaly při přípravě předmětu. Na těchto „meta-příkladech“ studenti vidí, že ani vyučující nepřevezme první výstup modelu bez výhrad. Vidí proces ladění, upřesňování zadání (prompt engineering) a opravování chyb. Tímto transparentním přístupem demystifikujeme AI a vedeme studenty k tomu, aby ji chápali jako mocný nástroj, za jehož výstupy však nesou plnou odpovědnost oni sami.

2 JupyterLab v praxi

Jádrem výuky se stal JupyterLab. Pro prezentaci na přednáškách je využíván plugin *jupyterlab-slideshow* (fork pluginu *jupyterlab-deck*), který umožňuje transformovat buňky notebooku do formátu prezentace, přičemž kód zůstává živý a spustitelný.

2.1 Markdown a LLM

Volba JupyterLabu se ukázala jako strategická i z hlediska spolupráce s AI. Jupyter notebooky jsou ve své podstatě textové soubory (JSON), které kombinují kód a text ve formátu Markdown. Pro LLM je tento textový formát „přirozeným jazykem“.

Modelu tak lze jednoduše zadat instrukci: „Vytvoř výkladovou část o cyklech v Markdownu, přidej příklad v Pythonu a následně navrhni kvízovou otázku.“ AI generuje strukturovaný text, který pedagog pouze zkopíruje do buňky notebooku. Plugin JupyterSlides pak na základě jednoduchých metadat tento obsah automaticky naformátuje do vizuálně atraktivní prezentace. Tento proces je řádově rychlejší než tvorba slidů v grafických nástrojích typu PowerPoint, kde AI naráží na bariéru vizuálního rozložení prvků.

2.2 Interaktivita a vizualizace

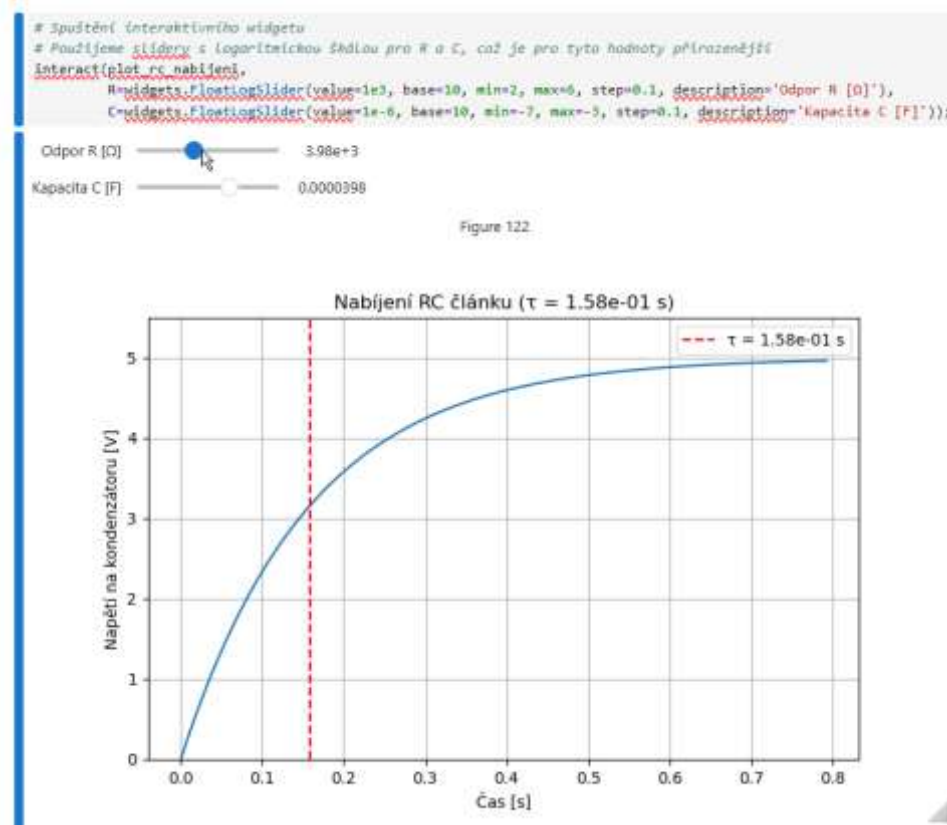
Pro demonstraci složitějších konceptů, jako je zpracování signálů nebo matematické funkce, byly využity knihovny *ipywidgets*, *matplotlib* a *plotly*. Studenti tak nemusejí pouze pasivně sledovat statické grafy, ale mohou v reálném čase měnit parametry (např. frekvenci) a sledovat dopad na výsledek (Obrázek 3).

Při přípravě těchto materiálů jsme narazili na technické specifikum školního prostředí. Webové vizualizace pomocí *plotly* se standardně spoléhají na načítání JavaScriptu z externích CDN serverů. To v offline režimu nebo za školním firewallem způsobovalo nefunkčnost materiálů. Řešením, nalezeným v kooperaci s LLM, byla vynucená inkluze JS knihovny přímo do souboru notebooku (`include_plotlyjs=True`), což zajistilo robustnost materiálů. Při použití *plotly* v rámci JupyterLabu bylo nutné nainstalovat také knihovnu *anywidget* apod. Opět tedy bylo třeba korekce návodů a materiálů z lidské strany.

2.3 Přechod k profesionálnímu IDE

Ačkoliv JupyterLab tvoří didaktickou páteř výuky, pro budoucí inženýry je klíčové osvojit si práci v plnohodnotném vývojovém prostředí. V pokročilejší fázi kurzu proto probíhá řízený přechod do prostředí PyCharm (případně MS VS Code). Toto prostředí je nezbytné pro strukturální vývoj komplexnějších projektů, jako jsou desktopové (PySide6) a webové aplikace, a pro pokročilá témata, která se v noteboocích obtížně simulují – například práce s korutinami, resp. asynchronním programováním (`async/await`).

I v této fázi hraje AI ústřední roli. Studenti se učí využívat integrované AI asistenty a copiloty přímo v editoru pro refaktoring a generování boilerplate kódu. Metodika tohoto přechodu, včetně sady cvičení zaměřených na efektivní práci s IDE (debugging, správa virtuálních prostředí), byla opět vyvinuta a validována v tandemu s LLM.



Obrázek 3: JupyterLab v režimu prezentace zobrazující interaktivní graf s posuvníky.

3 Výuka objektového programování

Jedním z cílů bylo demonstrovat principy OOP na atraktivních a oborově relevantních příkladech. Spolupráce s LLM zde umožnila masivní generování variabilních zadání, což by manuálně bylo časově neúnosné.

3.1 Tvorba výukových her

Ve spolupráci s LLM byly vyvinuty výukové minihry, a to jednak textové a následně i grafické (využívající knihovny *tkinter* a *pygame*), které slouží jako „pískoviště“ pro studenty.

Při vývoji hry „Chyt' mě!“ (*tkinter*) jsme řešili problém s inicializací grafického plátna. Metoda pro zjištění rozměrů okna vracela chybné hodnoty před plným vykreslením aplikace, což vedlo k chybám při generování náhodných souřadnic. Řešení spočívalo v odloženém startu herní smyčky (`.after()`) a vynucení aktualizace widgetů (`update_idletasks`). Tento technický detail se stal vhodnou ukázkou, na kterém lze studentům demonstrovat životní cyklus GUI aplikace a událostmi řízené programování.

3.2 Vazba na obor studia

Klíčovou výhodou LLM je schopnost kontextualizace. Pro studenty elektrotechniky jsme mohli snadno generovat příklady „na míru“. Místo abstraktních tříd typu *Zvire* a *Pes* jsme společně s modelem vytvořili příklad evidence elektronických součástek. Zde demonstrujeme přechod od procedurálního kódu (seznam slovníků reprezentujících rezistory) k objektovému návrhu (třídy *Soucastka*, *Kondenzator* dědící ze *Soucastka*). LLM pomohl vygenerovat nejen kód, ale

i srovnávací analýzu výhod a nevýhod obou přístupů, což studentům pomáhá pochopit motivaci pro použití OOP v inženýrské praxi.

4 NotebookLM

V poslední fázi přípravy byly materiály exportované z JupyterLab (ve formátu PDF) integrovány do nástroje Google NotebookLM. Tento systém na bázi RAG (Retrieval-Augmented Generation) umožňuje studentům generovat souhrny, audio podcasty simulující diskusi nad látkou či cvičné kvízy přímo z našich výukových textů.

Zde je však opět nutné zdůraznit rizika, která studentům komunikujeme. I pokročilé modely mohou při generování audia nebo interpretaci grafiky v PDF halucinovat nebo zjednodušovat fakta. Studentům je tento nástroj prezentován jako doplněk pro opakování, u kterého je nutné zachovat kritické myšlení a ověřovat informace vůči primárním zdrojům (samotným notebookům).

Závěr

Využití tandemu pedagog-LLM při tvorbě kurzu se ukázalo jako vysoce efektivní. Umožnilo vytvořit technicky náročnější a interaktivnější materiály v rozsahu, který by při tradiční manuální přípravě nebyl dosažitelný. Hlavním přínosem však není jen efektivita, ale změna paradigmatu. Tím, že studentům ukazujeme AI jako nedokonalého, ale užitečného partnera, a transparentně sdílíme proces tvorby, je připravujeme na realitu moderního inženýrství, kde je schopnost kooperace s AI klíčovou kompetencí. Výsledný kurz tak kombinuje moderní technologie s ověřenými inženýrskými postupy a kritickým myšlením.

Poděkování

Tento příspěvek vznikl za podpory projektu SGS-2024-025 a Fakulty elektrotechnické Západočeské univerzity v Plzni.

Literatura

Barba, L. A. et al. (2019). [online], [2025-11-28]. Teaching and Learning with Jupyter. Dostupné z: <https://jupyter4edu.github.io/jupyter-edu-book/>

Mollick, E. R., Mollick, L. (2023). [online], [2025-11-28]. Assigning AI: Seven Approaches for Students, with Prompts. SSRN Electronic Journal. Dostupné z: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4475995

Jupyter Project. JupyterLab Documentation [online], [2025-11-28]. Dostupné z: <https://jupyterlab.readthedocs.io/en/stable/>

Jupyter Widgets. Jupyter Widgets documentation [online], [2025-11-28]. Dostupné z: <https://ipywidgets.readthedocs.io/en/stable/>

Plotly. Plotly Open Source Graphing Library for Python [online], [2025-11-28]. Dostupné z: <https://plotly.com/python/>

JupyterlabSlideshow. Lightweight presentations for JupyterLab [online], [2025-11-28]. Dostupné z: <https://github.com/jupyterlab-contrib/jupyterlab-slideshow>

JupyterlabDeck. Lightweight presentations for JupyterLab [online], [2025-11-28]. Dostupné z: <https://github.com/deathbeds/jupyterlab-deck>

Kontaktní údaje

Ing. Petr Kropík, Ph.D., Ing. Lenka Šroubová, Ph.D.
Katedra elektrotechniky a počítačového modelování
Západočeská univerzita v Plzni, Fakulta elektrotechnická
Univerzitní 26, 301 00 Plzeň
e-mail: pkropik@fel.zcu.cz, lsroubov@fel.zcu.cz

METRIKY VE SCRUMU: DOTAZNÍKOVÁ APLIKACE PRO ZJIŠTĚNÍ ÚROVNĚ VYSPĚLOSTI POUŽÍVÁNÍ AGILNÍCH METRIK V TÝMU

METRICS IN SCRUM: A SURVEY APP TO DETERMINE THE METRIC MATURITY LEVEL OF AGILE METRICS UTILIZATION IN THE TEAM

Josef Kunhart

Abstrakt

Scrum v současné době představuje nejúspěšnější metodu pro vývoj software. Agilní projekty přesto selhávají kvůli nízké efektivitě a vyspělosti týmů. V rámci původního výzkumu jsme zkoumali faktory a metriky, které ovlivňují efektivitu Scrum týmů. Na základě výzkumu jsme připravili dotazníkovou aplikaci pro sběr dat a ověření závěrů výzkumu v praxi. Dotazník je určen pro členy agilních týmů. Respondent po vyplnění dotazníku získá cennou informaci o úrovni využívání metrik ve svém týmu (skóre vyspělosti používání metrik). Přínosem řešení pro autory výzkumu je získání dat pro porovnání závěrů první části výzkumu a reálného využívání metrik v praxi.

Klíčová slova: Agilní projektové řízení, efektivita, metriky, Scrum, týmová práce

Abstract

Scrum is currently the most successful method for developing software. However, agile projects still fail due to low team efficiency and maturity. In the original research, we examined the factors and metrics that affect the efficiency of Scrum teams. Based on previous research, we implemented a survey app to collect data and verify the results in practice. The survey is intended for agile professionals. Upon completing the survey, the respondent receives priceless information about the use of metrics in the team (metric maturity score). The main benefit of the solution for the authors is obtaining data to compare the results of the initial research and the actual use of metrics in practice.

Keywords: Agile project management, efficiency, metrics, Scrum, teamwork

Úvod

Agilní řízení projektů a vývoje softwaru je inovativní přístup k řízení projektů, který se odlišuje od tradičních postupů projektového řízení (Shore, 2021). Agilní metody velmi dobře fungují v dynamickém prostředí změn, u start-upů a inovativních softwarových projektů (Highsmith, 2010). Scrum je v současné době široce využíván nejen pro vývoj softwarových aplikací (Hobbs a Petit, 2017), ale našel uplatnění i v různorodých oblastech mimo tradiční projektové řízení, například ve výzkumu (Hron a Obwegeser, 2022). Podle Kadenice a kol. (2023) a 17th State of Agile Report od Digital.ai (2024) jsou v současné době nejpoužívanějšími agilními metodami Scrum a hybridní metody založené na Scrumu. Metriky ve Scrumu představují softwarové řešení pro dotazníkovou aplikaci, která vychází z původního výzkumu autora v oblasti měření efektivity Scrum týmů pomocí metrik. Dotazníková aplikace bude sloužit pro sběr dat, která využijeme v navazující fázi výzkumu, tedy porovnání výstupů úvodního výzkumu a praxe.

Shrnutí dosavadního výzkumu

V rámci širšího výzkumu se zabýváme faktory, které ovlivňují vyspělost agilních týmů při vývoji software. Tyto faktory lze rozdělit na měkké (soft) a tvrdé (hard). Měkké faktory zahrnují interakci v týmu, vliv lidského faktoru a další obtížně měřitelná kritéria. Pro analýzu měkkých faktorů a problematických situací v agilních týmech lze využít postupů měkké systémové metodologie (Kunhart a Bartoška, 2025c). Tvrdé faktory naopak zahrnují kvantifikovatelné údaje, v případě agilních týmů se jedná zejména o metriky pro efektivitu a produktivitu týmů. V první části výzkumu jsme na základě analýzy a rešerše literatury identifikovali metriky, které ovlivňují efektivitu a produktivitu týmů (Kunhart a Bartoška, 2025a). V navazujícím výzkumu plánujeme s využitím dotazníkové aplikace sesbírat data a ověřit závěry úvodního výzkumu v praxi. Zde budeme cílit na efektivitu týmů a metodu Scrum jako nejpoblárnější agilní metodu.

Data a metody

Pro řešení dotazníku využíváme seznam metrik pro měření efektivit Scrum týmů na základě původního výzkumu autora, modely vyspělosti projektového řízení pro definici úrovní vyspělosti metrik a softwarové řešení pro implementaci dotazníkové aplikace.

Metriky pro měření efektivit Scrum týmů

Základem řešení je seznam nejdůležitějších metrik, které ovlivňují efektivitu Scrum týmů (viz tabulka 1). Pořadí metrik, domény, skupiny a váhy metrik v tabulce přímo vychází z původního výzkumu autora tohoto příspěvku (Kunhart a Bartoška, 2025a). Pro potřeby dotazníku využíváme nejdůležitějších 21 metrik z celkového počtu 61 metrik, které jsme identifikovali pro efektivitu Scrum týmů na základě úvodního výzkumu. Počet metrik pro dotazník byl zvolen na základě zjištěných vah metrik a rozumného rozsahu dotazníku, kdy ideální doba vyplňování je 10-15 minut (Revilla a Höhne, 2020). Váhy metrik v tabulce vycházejí z celkového seznamu metrik podle výzkumu.

Tabulka 1: Seznam nejdůležitějších metrik, které ovlivňují efektivitu Scrum týmů.

Metrika	Orientační překlad/popis	Doména	Skupina	Váha
Velocity	Rychlost práce	Team	Výkon	0,0971
Cycle time	Doba cyklu	Team	Výkon	0,0706
Work capacity	Pracovní kapacita	Team	Výkon	0,0530
Focus factor	Faktor soustředění	Team	Motivace	0,0353
Target value increase	Zvýšení cílové hodnoty	Team	Výkon	0,0353
Estimation accuracy	Přesnost odhadu	Sprint	Review/Demo	0,0259
Sprint Burn-down	Zbývajíc práce ve Sprintu	Artefakty	Sprint Backlog	0,0259
Defect count	Počet defektů	Tech	Kvalita	0,0259
Defect count per user story	Počet defektů na storky	Tech	Kvalita	0,0259
Test automation percentage	Podíl automat. testování	Tech	Testování	0,0207
Defect removal percentage	Podíl odstranění defektů	Tech	Kvalita	0,0207
Defects severity	Závažnost defektů	Tech	Kvalita	0,0207
Team size	Velikost týmu	Sprint	Planning	0,0207
Number of accepted user stories	Počet akceptovaných storek	Sprint	Review/Demo	0,0207
Number of rejected user stories	Počet zamítnutých storek	Sprint	Review/Demo	0,0207
Team satisfaction score	Skóre spokojenosti týmu	Team	Motivace	0,0177
Team member turnover	Fluktuace členů týmu	Team	Výkon	0,0177
Interruption time	Podíl přerušené práce	Team	Výkon	0,0177
Team efficiency	Efektivita využití zdrojů	Team	Výkon	0,0177
Number of user stories in the Sprint	Počet storek ve Sprintu	Artefakty	Sprint Backlog	0,0155

Zdroj: Kunhart a Bartoška (2025a)

Modely úrovně projektového řízení

Modely úrovně projektového řízení (project management maturity models, PMMM) jsou nástroje, které pomáhají měřit úroveň projektového řízení v organizacích a analyzovat schopnost organizací řešit dílčí projekty (Košťálová, 2023). Mezi základní modely úrovně projektového řízení patří Capability Maturity Model Integration (CMMI) pro hodnocení zralosti organizací (CMMI Institute, 2025) a Agile Maturity Model (AMM) pro hodnocení vyspělosti agilního projektového řízení (Chetankumar a Ramachandran, 2009). CMMI definuje pět úrovní zralosti (1: Initial, 2: Managed, 3: Defined, 4: Quantitatively managed, 5: Optimizing), zatímco AMM definuje pět úrovní vyspělosti agilního řízení (0: Initial, 1: Just started, 2: Defined, 3: Measured, 4: Optimal). Další PMMM jsou analyzovány v publikaci od Košťálové (2023).

Softwarové řešení

Softwarové řešení pro Metriky ve Scrumu zahrnuje back-endovou a front-endovou aplikaci, které spolu komunikují přes standardizované REST¹ API². Back-endová část zahrnuje databázi, administraci řešení a aplikační rozhraní pro front-end. Front-endová část představuje samotný dotazník pro sběr dat, který odesílá vyplněné údaje na back-end. Softwarové řešení technicky navazuje na dřívější řešení autora POMS³ Online (Kunhart, 2025b). V tomto příspěvku jsou zároveň detailně popsány koncepty a metody pro implementaci softwarového řešení.

Výsledky a implementace

Metric maturity level

Metric maturity level (MML, úroveň vyspělosti používání metrik) označuje úroveň používání zvolené metriky v agilním týmu. Pro potřeby výzkumu a dotazníku jsme definovali čtyři úrovně MML, které vychází z modelů úrovně projektového řízení. Úrovně MML jsou celkem čtyři:

1. Not used (Nepoužívaná): Metrika není v týmu využívána (výchozí úroveň).
2. Collected (Sledovaná): Metrika je pravidelně sledována a evidována.
3. Evaluated (Vyhodnocovaná): Metrika je iterativně vyhodnocována na základě evidovaných údajů.
4. Improved (Zlepšovaná): Na základě vyhodnocování metriky jsou podnikány kroky ke zlepšení metriky a efektivity týmu.

Jednotlivé úrovně MML vychází z úrovní vyspělosti podle modelů CMMI a AMM. Hlavní rozdíl je ten, že se jedná o úrovně metrik, nikoliv procesů a řízení. Nelze tedy hodnotit úroveň plánování a definice postupů jako u hodnocení vyspělosti organizací. Dílčí úrovně MML jsou proto přizpůsobeny metrikám. Z pohledu dotazníku odpovídá ohodnocení dílčích úrovní MML čtyřbodové Likertově škále (Likert, 1932). Tyto úrovně odpovídají enumeraci s hodnotami 1-4.

Metric maturity score

Metric maturity score (skóre vyspělosti používání metrik, MMS) udává absolutní nebo relativní číselnou hodnotu vyspělosti týmu ve vybrané oblasti. V rámci dotazníku počítáme MMS pro tři dílčí oblasti a celkové MMS pro všechny metriky. Každá oblast zahrnuje sadu metrik podle tabulky 1.

¹ REpresentational State Transfer.

² Application Programming Interface.

³ Profile of Mood States.

- Team: Zahrnuje doménu Team (skupiny Motivace a Výkon).
- Sprint: Zahrnuje domény Sprint a Artefakty (včetně všech skupin).
- Tech: Zahrnuje doménu Tech (skupiny Kvalita a Testování).

Původní domény Sprint a Artefakty jsme v dotazníku spojili do jedné oblasti, kvůli zachování přibližně stejného počtu metrik v dílčích oblastech.

$$s_{abs} = \sum_{i=0}^n (u_i - 1)w_i \quad (1)$$

Výpočet absolutního skóre s_{abs} pro vybranou oblast vyspělosti používání metrik je zobrazen ve vzorci 1, výpočet maximálního skóre s_{max} ve vzorci 2.

$$s_{max} = \sum_{i=0}^n (\max(u) - 1)w_i \quad (2)$$

V obou vzorcích označuje n počet metrik ve vybrané oblasti, i index metriky, u_i úroveň MML, w_i váhu dané metriky podle tabulky 1. Hodnoty MML (u) 1-4 jsou odečtením jedničky transformovány na hodnoty 0-3. Důvodem je, aby nejnižší možné skóre s_{abs} bylo nula.

$$s_{rel} = \frac{s_{abs}}{s_{max}} \quad (3)$$

Relativní skóre s_{rel} pro dotazník je dopočítáno jako podíl obou hodnot, viz vzorec 3.

Dotazníková aplikace

Dotazník (Kunhart, 2025a) je implementován jako front-endová aplikace, která zahrnuje formulář pro dotazník, stránku s výsledky, a doplňkovou obsahovou stránku. Aplikace je plně lokalizována pro anglický a český jazyk. Náhled formuláře dotazníku je zobrazen na obrázku 1 (na obrázku je kvůli přehlednosti zobrazen pouze formulář bez doprovodných textů). Na stránce s formulářem je zobrazen základní popis výzkumu s odkazem na detail, instrukce pro vyplnění, vysvětlení úrovně MML a samotný formulář. Formulář zahrnuje pole pro vyplnění e-mailu a metriky podle tabulky 1. Názvy metrik jsou vždy uvedeny v angličtině, u každé metriky je zároveň zobrazen popis ve zvoleném jazyce.

Dotazníková aplikace byla vyvíjena s důrazem na pozitivní uživatelskou zkušenost. Aplikace je moderní, rychlá a plně responzivní. Jednotlivé úrovně MML jsou graficky rozlišeny. Formulář dále na pozadí implementuje víceúrovňovou ochranu na bázi ověření rychlosti vyplňování formuláře, honeypot mechanismu a kontroly duplicitního vyplnění. Cílem bylo vyhnout se uživatelsky nevhodným řešením jako je captcha. Stránka s výsledky obsahuje graf s MMS pro dílčí oblasti a celkové MMS pro všechny metriky. Náhled grafu je zobrazen na obrázku 2 (kromě grafu stránka obsahuje ještě doplňkové texty). Zobrazované údaje jsou v procentech, se zaokrouhlením na celá čísla. Na samostatné stránce jsou pak zobrazeny detaily o výzkumu a autorech úvodního výzkumu.

E-mail *

@

Velocity *Množství práce, které tým dokončí během Sprintu.*

☒ Nepoužívaná ☐ Sledovaná ☐ Vyhodnocovaná ☐ Zlepšovaná

Cycle time *Doba od začátku do dokončení user story.*

☒ Nepoužívaná ☐ Sledovaná ☐ Vyhodnocovaná ☐ Zlepšovaná

Work capacity *Pracovní čas týmu dostupný ve Sprintu.*

☒ Nepoužívaná ☐ Sledovaná ☐ Vyhodnocovaná ☐ Zlepšovaná

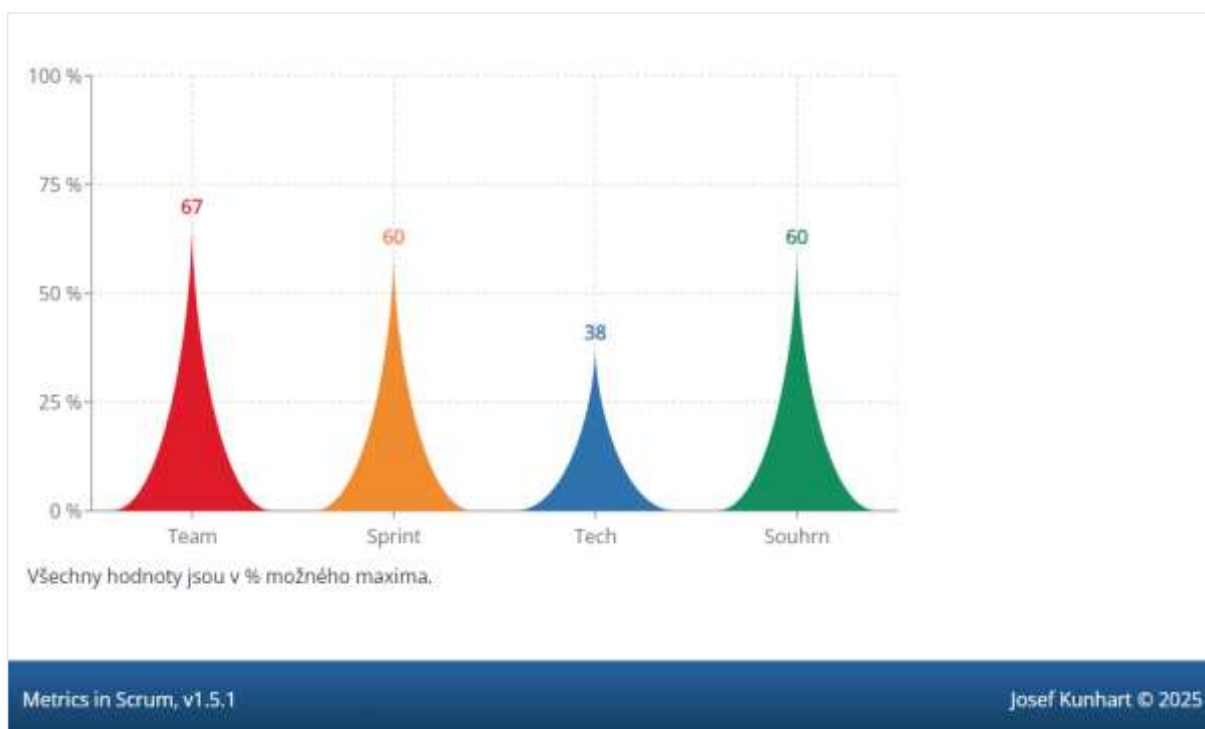
Focus factor *Využití pracovní kapacity týmu pro práci na user stories (vs. na jiných úkolech).*

☒ Nepoužívaná ☐ Sledovaná ☐ Vyhodnocovaná ☐ Zlepšovaná

Metrics in Scrum, v1.5.1 Josef Kunhart © 2025

Obrázek 1: Náhled dotazníku pro metriky. Zdroj: vlastní zpracování.

Po vyplnění formuláře jsou údaje uloženy přes REST API na back-endu do databáze. Kromě vyplněného e-mailu a úrovně metrik ukládáme jazyk formuláře, čas vyplnění a další technické údaje. E-mail je po vyplnění anonymizován jednosměrnou hashovací funkcí, která dále slouží jako unikátní identifikátor odpovědi. Vyplněné údaje lze prohlížet a exportovat v administraci.



Obrázek 2: Náhled výsledného skóre vypěstlosti metrik. Zdroj: vlastní zpracování.

Diskuze

Dotazníková aplikace je v době psaní článku (listopad 2025) plně funkční, po dokončení testovacího provozu. Oficiální spuštění a propagace řešení je plánována na začátek prosince 2025, po návratu autora ze zahraničního pobytu. Propagace řešení i analýza vyplněných odpovědí budou probíhat iterativně. Aplikace bude ponechána v provozu i po získání dostatečného vzorku dat pro navazující výzkum. Pro potřeby vývoje a zkoušení nových funkcí zároveň na odlišné doméně existuje testovací (demo) verze aplikace.

Omezení navrhovaného řešení se týkají především metrik. Seznam metrik v rámci úvodního výzkumu vychází primárně z literárních zdrojů v oboru agilního projektového řízení, agilního vývoje software a využívání metody Scrum. Tato literatura nemusí plně odpovídat reálnému využívání metrik v praxi. To je zároveň jeden z hlavních důvodů, proč budeme realizovat navazující výzkum, ve kterém porovnáme závěry úvodního výzkumu s informacemi, které získáme na základě sesbíraných dat od členů agilních týmů. Seznam metrik pro efektivitu Scrum týmů podle závěrů původního výzkumu obsahoval 61 metrik.

Pro potřeby dotazníku jsme seznam metrik zkrátili na 21 metrik (viz tabulka 1). Zkrácení proběhlo na základě vah metrik, kdy mezi 21. a 22. pozicí je významný (50 %) rozdíl mezi vahami metrik. Kumulativní váha pro prvních 21 metrik představuje přibližně 63 % kumulativní váhy všech metrik pro efektivitu. Pro potřeby dotazníku jsme oproti původnímu výzkumu spojili domény Sprint a Artefakty do jedné oblasti, Sprint. Hlavním důvodem pro spojení byl záměr mít v každé oblasti vyhodnocované v dotazníku přibližně stejný počet metrik. Na výsledné skóre mají největší vliv metriky v oblasti Team, protože metriky z této oblasti mají největší váhy a zaujímají v seznamu metrik čelní pozice. Původní výzkum se kromě efektivity zabýval také produktivitou Scrum týmů, kterou v rámci dotazníku ani navazujícího výzkumu dále nezkoumáme.

Závěr

Na základě původního výzkumu v oblasti měření efektivity Scrum týmů a metrik jsme navrhli a implementovali dotazníkovou aplikaci pro sběr dat k těmto metrikám. Dotazník je určený pro členy agilních týmů. Na základě dat získaných od členů agilních týmů v budoucím výzkumu porovnáme reálné využívání metrik v praxi se závěry úvodní části výzkumu, který vycházel zejména z literární rešerše a sémantických sítí pro určení důležitosti metrik. Respondent po vyplnění dotazníku zároveň získá cennou informaci o úrovni využívání metrik pro efektivitu ve svém agilním týmu.

Literatura

Chetankumar, P., Ramachandran, M. (2009). Agile Maturity Model (AMM): A software process improvement framework for agile software development practices. *International Journal of Software Engineering* 2(1), 32–43.

CMMI Institute (2025). [online], [2025-11-02]. ISACA. Dostupné z <https://cmminstitute.com/>.

Digital.ai. (2024). *17th State of Agile Report: Analyst reports*. [online], [2025-11-02]. Dostupné z: <https://digital.ai/resource-center/analyst-reports/state-of-agile-report/>.

Highsmith, J. A. (2010). *Agile project management, 2nd Edition*. Addison-Wesley. ISBN 978-0-32-165920-0.

- Hobbs, B., Petit, Y. (2017). Agile methods on large projects in large organizations. *Project Management Journal* 48(3), 3–19. <https://doi.org/10.1177/875697281704800301>.
- Hron, M., Obwegeser, N., Why and how is Scrum being adapted in practice: A systematic review. *Journal of Systems and Software* 183. <https://doi.org/10.1016/j.jss.2021.111110>
- Kadenic, M. D., Koumaditis, K., Junker-Jensen, L. (2023). Mastering Scrum with a focus on team maturity and key components of Scrum, *Information and Software Technology* 153(C). <https://doi.org/10.1016/j.infsof.2022.107079>.
- Košťálová, J. (2023). *Modely hodnocení úrovně projektového řízení*. Habilitační práce. Vysoká škola báňská. Technická univerzita Ostrava. Ekonomická fakulta.
- Kunhart, J. (2025a). *Metrics in Scrum: Survey*. [online], [2025-11-02]. Dostupné na: <https://metrics.kunhart.eu/>
- Kunhart, J. (2025b). POMS Online: Softwarové řešení pro sledování změn nálad student při výuce projektového řízení na základě psychologické metody Profile of Mood States. *Konference Informatika 2025*, 32–38.
- Kunhart, J., Bartoška, J. (2025a). Key metrics to improve the efficiency and productivity of Scrum teams: A semantic network approach. *Array* [Článek v review].
- Kunhart, J., Bartoška, J. (2025b). Mood states as a key factor to assess student learning in project management teaching. *Journal on Efficiency and Responsibility in Education and Science*, 18(2), 78–90. <https://doi.org/10.7160/eriesj.2025.180202>
- Kunhart, J., Bartoška, J. (2025c). Využití metodologie měkkých systémů pro analýzu situace ve Scrum týmu: Případová studie. *Mýty a fakta v projektovém řízení*. UTB Zlín, 76–93. ISBN 978-80-7678-371-3.
- Likert, R. (1932). A technique for the measurement of attitudes. *Archives of Psychology*, 22(140), 55.
- Revilla, M., Höhne, J. K. (2020). How long do respondents think online surveys should be? New evidence from two online panels in Germany. *International Journal of Market Research*, 62(5), 538–545. <https://doi.org/10.1177/1470785320943049>
- Shore, J. (2021). *The art of agile development, 2nd Edition*. O'Reilly. ISBN 978-1-49-208068-8.

Kontaktní údaje

Ing. Josef Kunhart, MBA
Katedra systémového inženýrství
Provozně ekonomická fakulta
Česká zemědělská univerzita v Praze
Kamýcká 129, 165 00 Praha – Suchbátka
E-mail: kunhart@pef.czu.cz

INOVACE VE VÝUCE PODNIKOVÉ ARCHITEKTURY A DIGITÁLNÍ TRANSFORMACE V INFORMATICKÝCH PŘEDMĚTECH

INNOVATION IN TEACHING ENTERPRISE ARCHITECTURE AND DIGITAL TRANSFORMATION IN COMPUTER SCIENCES

Martin Lukáš

Abstrakt

Předmět Digitální transformace podniku, vyučovaný na PEF ČZU od roku 2021, prošel v roce 2025 zásadní inovací díky projektu Trojvýzva. V rámci projektu byla vytvořena nová studijní opora – skriptu Digitální transformace a podniková architektura, která propojuje teoretické koncepty s praktickými nástroji jako TOGAF, ArchiMate, BPMN, EA Sparx či Archi. Výuka byla rozšířena o multimediální materiály, instruktážní videa, self-testy a semestrální projekty, které reflektují aktuální trendy jako DevOps, agilní řízení, udržitelnost a efektivitu. Cílem inovace je rozvoj digitálních kompetencí studentů a jejich schopnost navrhovat, analyzovat a ořídit digitální transformaci v různých typech organizací. Příspěvek prezentuje pedagogickou zkušenost s implementací těchto nástrojů a metod do výuky ICT předmětu.

Klíčová slova: digitální transformace, podniková architektura, informační systémy

Abstract

The course Digital Transformation of Enterprises, taught at the Faculty of Economics and Management of the Czech University of Life Sciences Prague since 2021, was significantly updated in 2025 through the Trojvýzva project. A new textbook, Digital Transformation and Enterprise Architecture, was developed to link theoretical concepts with practical tools such as TOGAF, ArchiMate, BPMN, EA Sparx, and Archi. Teaching materials were enhanced with multimedia content, instructional videos, self-tests, and semester projects reflecting trends like DevOps, agile management, sustainability, and efficiency. The innovation aims to strengthen students' digital competencies and their ability to manage digital transformation across various organizations. This paper shares pedagogical experience with integrating these tools and methods into ICT education.

Úvod

Digitální transformace je dnes jedním z nejdiskutovanějších témat v oblasti řízení organizací, technologií i vzdělávání. Vysoké školy, které připravují budoucí odborníky pro praxi, čelí výzvě, jak tento komplexní fenomén uchopit didakticky, srozumitelně a zároveň prakticky. Na Provozně ekonomické fakultě ČZU v Praze jsme se této výzvě postavili prostřednictvím předmětu Digitální transformace podniku, který je od roku 2021 součástí magisterského studia. V roce 2025 došlo k zásadní inovaci výuky díky projektu *Zvýšení kvality a účinnosti vzdělávacího procesu a strategického řízení na ČZU v Praze, reg.č. CZ.02.02.XX/00/23_022/0009069*, jehož cílem bylo propojit teoretické koncepty digitální transformace s praktickými nástroji podnikové architektury. Výsledkem je modernizovaný

kurz. Souběžně s tímto projektem (ovšem mimo něj) byla vypracována nová studijní opora – skripta Digitální transformace a podniková architektura, která reflektují trendy a praxi.

Metodologie

Kurz byl přepracován do 12 tematických bloků, každý s vlastní strukturou v LMS Moodle. Studenti mají k dispozici modernizované prezentace, PDF studijní texty, instruktážní videa, self-testy a případové studie. Výuka je postavena na aktivním zapojení studentů, kteří v rámci semestrálních projektů zastávají role jako Business Analyst nebo ICT Director a modelují scénáře digitální transformace pomocí nástrojů jako EA Sparx, ArchiMate s podporou BPMN či metodického rámce TOGAF. Skripta, která vznikla mimo projekt, pokrývají široké spektrum témat – od základních principů digitální transformace, přes návrh podnikové architektury, až po etické a bezpečnostní aspekty digitálního prostředí.

Výsledky

Projekt modernizace kurzu přinesl následující výstupy: 12 tematických bloků v LMS Moodle, 12 modernizovaných prezentací, shrnující studijní texty ke každému tématu ve formátu PDF, 16 instruktážních videí pro modelovací program Enterprise Architect Sparx, databáze 50 testových otázek, strukturované semestrální projekty, případové studie v závěru skript a cvičebnice – 20 konkrétních ukázkových nejlepších případových studií. Vypracované studenty za celou dobu výuky předmětu

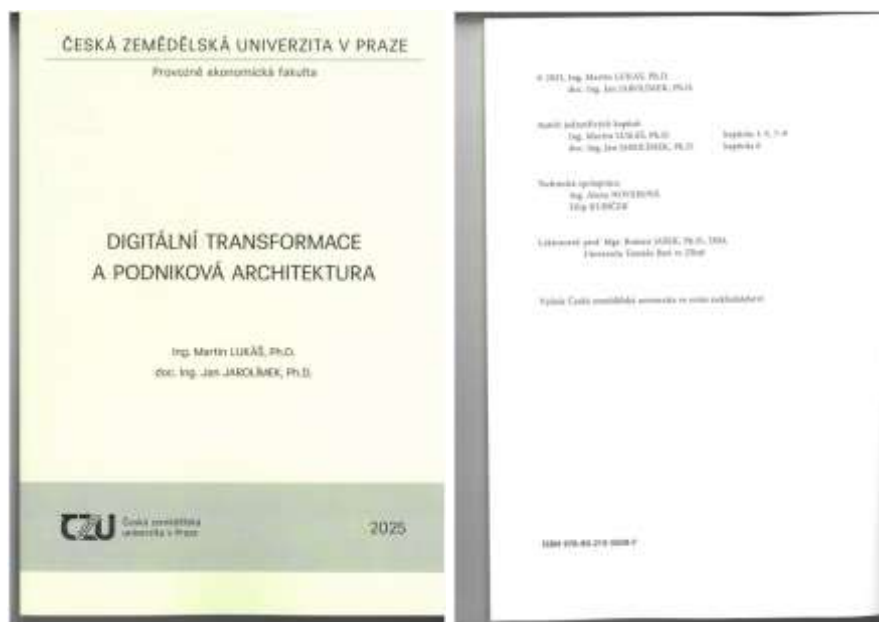
Tematické bloky kurzu:

- | | |
|---|---|
| 1. Úvod do digitální transformace | 6. BPMN – modelování procesů |
| 2. Digitální vyspělost organizací | 7. EA Sparx – nástroj pro modelování |
| 3. Podniková architektura – základní principy | 8. DevOps a agilní přístupy |
| 4. ArchiMate – modelovací jazyk | 9. Řízení změn a projektové řízení |
| 5. TOGAF – rámec podnikové architektury | 10. Role v týmu digitální transformace |
| | 11. Udržitelnost a efektivita v DTP |
| | 12. Případové studie a aplikace v praxi |

Obsah skript „Digitální transformace a podniková architektura“:

- | | |
|--|---|
| 1. Digitální transformace, vymezení a historický vývoj, sektory národního hospodářství | 5. Business architektura a procesní modely, typologie, přístupy |
| 2. Digitální vyspělost subjektů sektorů národního hospodářství, indikátory | 6. Digitální inovace, trendy a nové technologie |
| 3. Základy podnikové architektury, modely a rámce podnikové architektury | 7. Kybernetická bezpečnost v digitálním prostředí |
| 4. Strategie a motivační faktory digitální transformace, metody a modely | 8. Řízení iniciativ digitální transformace, metody, nástroje |
| | 9. Etika, legislativa, fenomény a udržitelnost digitální transformace |

Propojení bloků a kapitol: Tematické bloky kurzu byly navrženy tak, aby odpovídaly jednotlivým kapitolám skript. Například blok 1 „Úvod do digitální transformace“ přímo vychází z kapitoly 1 skript, kde je popsán historický vývoj a sektorové kontexty. Bloky 2 a 3 se opírají o kapitoly 2 a 3, které rozvíjejí koncept digitální vyspělosti a podnikové architektury. Bloky 4–6 reflektují kapitoly o strategiích, business architektuře a digitálních inovacích. Bezpečnostní a etické aspekty jsou pokryty v blocích 7, 9 a 11, které čerpají z kapitol 7–9 skript. Blok 12 pak slouží jako syntéza, kde studenti aplikují získané poznatky na případových studiích.



Obrázek 1: Obálka a úvodní strana skript. Zdroj: Lukáš, Jarolímek (2025)

Výsledky propojení a přínosy: Toto propojení přináší několik zásadních výhod:

- **Konzistence výuky:** Studenti se pohybují v jednotném rámci, kde se teorie ze skript přímo promítá do praktických bloků kurzu.
- **Lepší orientace v tématu:** Jasné členění umožňuje sledovat logickou návaznost témat.
- **Reakce na výzvy současnosti:** Témata jako digitální vyspělost, kybernetická bezpečnost, udržitelnost, DevOps či etika reflektují aktuální potřeby praxe i společnosti.
- **Podpora kompetencí:** Studenti si osvojují nejen znalosti, ale i dovednosti v oblasti modelování, analýzy a řízení digitální transformace.
- **Přenositelnost:** Struktura kurzu a skript je navržena tak, aby byla využitelná i na jiných vysokých školách a v různých oborech.

Inovovaný předmět tak představuje moderní vzdělávací rámec, který propojuje akademickou teorii s praktickými nástroji a reaguje na dynamiku digitálního prostředí. Jeho cílem je připravit studenty na roli aktivních tvůrců digitální budoucnosti.

Diskuse

Zároveň se ukazuje, že oblast digitální transformace a podnikové architektury je stále mladá, dynamická a interdisciplinární. Skripta i kurz jsou navrženy tak, aby byly využitelné nejen na PEF ČZU, ale i na dalších vysokých školách, které se věnují ICT vzdělávání. Tam, kde to bylo účelné, jsou kapitoly obohaceny o modely v jazyce ArchiMate, které slouží jako výrazový prostředek pro komunikaci mezi IT a business komunitou. Z pohledu pedagoga je největším

přínosem projektu umožnění systematicky rozvíjet výuku v oblasti, která je pro budoucnost digitálního Česka klíčová.

Závěr

Inovace výuky předmětu Digitální transformace podniku prostřednictvím projektu a nových skript představuje úspěšný příklad pedagogické transformace ICT vzdělávání. Výsledky ukazují, že kombinace podnikové architektury, digitálních nástrojů a aktivních metod výuky vede k rozvoji klíčových kompetencí studentů pro digitální ekonomiku.

Literatura

LUKÁŠ Martin a JAROLÍMEK Jan. 2025. Digitální transformace a podniková architektura (skripta pro Česká zemědělská univerzita v Praze, Provozně ekonomická fakulta, Katedra informačních technologií, ISBN 978-80-213-3509-7

LUKÁŠ Martin. Enhancing Student's Skills Of Enterprise Architecture By AI Support: A Case of Czech Republic. Online.. In: Proceedings of 2025 International Conference on Artificial Intelligence and Education (ICAIE 2025), IEEE Catalog Number: CFP2577J-USB. Suzhou, China:Institute of Electrical and Electronics Engineers, Inc., 2025, s. 681-685, 979-8-3315-2295-7. <https://ieeexplore.ieee.org/document/11158170>

LUKÁŠ Martin a BRABEC Miroslav. Support of digital transformation of human resource processes at City hall by enterprise approach: A case of the regional Czechia administration. Online.. In: IDIMT-2025 ICT in Business: AI Everywhere? Glory and Disgrace of AI. Prague, Czech Republic:TRAUNER Verlag + Buchservice GmbH, Königstrasse 14, Linz, Austria, 2025, s. 223-231, 978-3-99151-856-3. <https://epub.jku.at/doi/10.35011/IDIMT-2025-223>

LUKÁŠ Martin; ULMAN Miloš; PAVELKA Miroslav a AMBRUZ Pavel. Support of Project & Program Management of Digital Transformation Initiatives by Holistic Enterprise Architecture in Sectors of National Economy: A Case of the Czech Republic. In: ProjMAN conference proceeding. Portugal:IPCA, 2024, DOI: [10.1016/j.procs.2025.02.289](https://doi.org/10.1016/j.procs.2025.02.289)

LUKÁŠ Martin. Digital Transformation of Primary and Secondary Institutions: A Case Of the Czech Republic. In: ICHELT2024 - International Conference on Higher Education Learning and Teaching 2024. Miri, Malaysia, Borne:Curtin University, 2024, s. 30-30.

LUKÁŠ Martin; ULMAN Miloš a LOHR Václav. Support of digital maturity by enterprise architecture approach in different sectors of national economy: A case of Czech Republic. Online.. In: IDIMT-2024 Changes to ICT, Management, and Business Processes through AI. Prague:TRAUNER Verlag + Buchservice GmbH, 2024, s. 223-230, 978-3-99151-527-2. Dostupné z: DOI: [10.35011/IDIMT-2024-223](https://doi.org/10.35011/IDIMT-2024-223)

Kontaktní údaje

Ing.Martin Lukáš, Ph.D.

Katedra informačních technologií

Provozně ekonomická fakulta, Česká zemědělská univerzita v Praze

Kamýcká 129, 165 00 Praha – Suchbátka

tel: +420 731 137 393, email: lukas@pef.czu.cz

VÝUKA SEMINÁŘE ZÁKLADY ALGORITMIZACE V PRVNÍM ROČNÍKU

TEACHING SEMINAR INTRODUCTION TO ALGORITHMS IN THE FIRST YEAR

Radek Matoušek

Abstrakt

Seminář s novým harmonogramem vznikl v roce 2023 s cílem připravit studenty pro zvládnutí navazujících předmětů. Průběžné výsledky byly vyhodnocovány, a mohly tak posloužit k sestavení závěrečné hodnotící zprávy. Konferenční příspěvek ze zprávy čerpá – ve zkratce ukazuje způsob vedení výuky, který byl aplikován, a předkládá závěry plynoucí z analýzy, které by měly vést ke zlepšení standardu výuky.

Klíčová slova: výuka, revidovaná Bloomova taxonomie, konceptuální porozumění

Abstract

The seminar with a new schedule was created in 2023 with the aim of preparing students to master follow-up subjects. The interim results were evaluated and could thus be used to compile the final evaluation report. The conference paper draws on the report – it briefly shows the teaching method that was applied and presents conclusions from the analysis that should lead to an improvement in teaching standards.

Keywords: teaching, revised Bloom's taxonomy, conceptual understanding

Úvod

Motivací k sepsání práce nebylo nic jiného než úvaha nad tím, kam až je možné se snižováním nároků na studenty zajít, a co se s tím dá dělat. V úvahách byly použity teorie z didaktiky, jako je revidovaná Bloomova taxonomie, procedurální dovednosti a konceptuální porozumění. Po zavedení zmíněných pojmů byla tříletá výuka předmětu podrobena strukturovanému analytickému zkoumání s cílem najít příčiny problémů, které se objevily. Po analýze, ze které příspěvek vychází (Matoušek, 2025), byla stanovena řešení, která by mohla kvalitu výuky zlepšit.

Model výuky

Vzhledem k vysoké diverzitě znalostí studentů se zdálo být vhodné aplikovat adaptivní způsob výuky. Po promyšlení všech pro a proti tradičního modelu byla výuka nastavena jako kombinace adaptivního, procedurálního a konceptuálního modelu pro jehož možnou aplikaci musely být nejprve v první polovině semestru položeny základy u studentů, kteří začínají s nulovými znalostmi, aby podle paradigmatu konceptuálního modelu mohla být vedena výuka bohatá na relace.

Adaptivní výuka představuje technicky náročný úkol, který se nedá zvládnout za jeden rok. V této oblasti se nám již podařilo vyvinout několik nástrojů, které práci výrazně usnadňují. Další projekty, které by nám umožnily náš technický arzenál pro automatické vyhodnocování

úkolů rozšířit, máme v plánu; nejraději bychom samozřejmě byly, pokud by se jich zvládli ujmout talentovaní studenti.

Pro potřebu velkého množství testových otázek byl vytvořen jednoduchý metajazyk, který umožňuje otázky generovat na základě vytvořené vzoru v množství variant. Přičemž chybné odpovědi nejsou jen náhodně vygenerované, ale obsahují chybné odpovědi na základě špatného (často) nesprávně pochopeného algoritmu.

Úvod do první poloviny semestru

První polovina semestru je věnována naprostým základům. Výuka je zaměřena na rozvoj procedurálních dovedností pro zautomatizování základních návyků na jednoduchých algoritmech – naučení potřebné syntaxe a její automatické používání pro složitější úlohy. Na seminářích je probráno přes 30 příkladů, na kterých jsou naučeny všechny základní principy a vysvětleny ty nejjednodušší algoritmy.

Každý příklad je nejprve zadán jako slovní úloha, ke které jsou vysvětleny potřebné náležitosti – vztahy, vzorce, typy proměnných aj. Studenti jsou obcházeni a pokud jsou na dobré cestě k vyřešení, jen potřebují pomoc s nějakým problémem, vyučující pomůže problém odstranit. Ostatní jsou ponecháni ještě chvíli v procesu přemýšlení. Po přiměřené době je rozkreslen na tabuli vývojový diagram a stejné kolečko kontroly provedeno ještě jednou. Na seminářích se tímto studenti učí propojovat vizuální myšlenkovou mapu v podobě vývojového diagramu a přepisovat ji do zdrojového kódu. Na závěr je zdrojový kód promítnut vedle nakresleného vývojového diagramu a je poukázáno na důležité části.

První polovina semestru končí procházením polí, vyhledáváním různých hodnot a naplnění pole Fibonacciho posloupností. Studenti už by měli například samostatně zvládnout vymyslet a naprogramovat algoritmus na převod čísla z desítkové soustavy do dvojkové. Přitom, pro správné pořadí bitů, použít ukládání do pole a vytisknout výsledek ve správném pořadí s vloženými mezerami po čtveřici bitů a zarovnáním na celé bajty.

- Seminář 1. týden (vývojové diagramy)
- Seminář 2. týden (základy syntaxe jazyka C/C++, větvení)
- Seminář 3. týden (smyčky typu *while*)
- Seminář 4. týden (*break*, *continue*, smyčka typu *for* a *for-each*)
- Seminář 5. týden (další jednoduché algoritmy)
- Seminář 6. týden (jednorozměrná statická pole)
- Seminář 7. týden (test, nejčastější chyby, max/min hodnota pole)

Zvládnutí první části si studenti ověří v prvním testu, který je povinný. Nízké doporučené skóre bylo nastaveno záměrně motivačně na 50 %. Pár studentů po testu obvykle ještě přehodnotí svoji chybně odhadnutou úroveň znalostí a rozhodnou se na nepovinné semináře docházet a věnovat jim více času.

Úvod do druhé poloviny semestru

Potvrdilo se, že studenti, kteří získali známku z prvního testu na spodní hraně, mají v druhé půlce už značné problémy. Studentům už také není na semináři pomáháno, z časových důvodů, s problémy, které mají mít zvládnuté z první půlky, a jsou odkázáni na procvičované příklady případně na příklady z knihy Algoritmizace (Pšenčíková, 2009). Respektive rozsáhlejší

pomoc se jim dostane možná ve zbylém čase, přednost mají studenti, kteří řeší problémy z aktuální hodiny.

Průběh seminářů druhé části probíhá podobně. Jsou probírány složitější algoritmy, které vyžadují více času pro vysvětlení principu. Vývojových diagramů postupně ubývá – a kreslí se jen tam, kde je to užitečné, například pokud je vysvětlován nový princip. Dostává se tak více času na vyřešení slovní úlohy podle zadání.

- Seminář 8. týden (algoritmy *bubble sort*, *reverse*, *unique*, *insert sort*)
- Seminář 9. týden (*selection sort*, hledání prvku v poli, fronta FIFO)
- Seminář 10. týden (zásobník LIFO, dvourozměrná pole, zapouzdření)
- Seminář 11. týden (parametry funkcí, ráce s řetězci, dynamické pole)
- Seminář 12. týden (spojový seznam, binární strom)
- Seminář 13. týden (test – při splnění podmínek udělení zápočtu)

Studenti se snaží více porozumět strukturám algoritmů a vzájemným souvislostem, rozvíjí se konceptuální porozumění. Tři úrovně porozumění popisuje Krathwohl (Krathwohl, 2002), na jejichž základě jsme odvodili vlastnosti pro problémy, které se řeší v předmětech algoritmizace:

- znalosti klasifikací – spíše izolované povědomí o algoritmech a jejich vlastnostech,
- znalosti struktur – schopnost vnímat vlastnosti algoritmů – porovnávat je, třídit, vhodně používat, vybírat lepší apod.
- znalost principů – schopnost rozpoznat příležitosti pro aplikace algoritmů, pravidel a zákonitostí, tj. vytvářet vlastní modifikované algoritmy na míru podle potřeby, vhodně spojovat již existující aj.

Semestr končí druhým zápočtovým testem s odevzdaným krátkým naprogramovaným algoritmem. Podle výsledků je zřejmé, že pro studenty je druhý test už obtížněji zvládnutelný. Vzhledem k tomu, že vhodných úkolů do testu není příliš, tak většina nakonec uspěje, někteří při opakování zápočtu (až třikrát). Minimálně si ale odnesou poznatek, že jejich úspěch měl řadu nedostatků, které bude třeba nejspíš ještě vylepšit, pokud se chtějí posunout výše.

Zkouška

Zkouška se skládá ze vstupní části, kterou je opět test ze stejné databanky jako všechny doposud předkládané, jen je potřeba dosáhnout vyšší minimální skóre 60 %. Na testu mohou ti, kteří získali zápočet s velkým štěstím ještě vypadnout. Zároveň je to pro studenty opakování hlavně faktických či procedurálních znalostí, které jsou pro některé studenty pro zvládnutí studia zřejmě klíčové. Pro mnohé test na začátku zkoušky není problém a může jim jen vylepšit známku ze druhé části, která je náročnější.

Druhá část zkoušky tvoří vyřešení problému, který je zadán slovní úlohou – způsob, který byl se studenty nacvičován od začátku. Jen u zkoušky se nejedná o dílčí malý úkol, ale již komplexnější řešení, který mnohem více odpovídá reálnému problému. Je třeba prokázat vyšší schopnost konceptuálního porozumění (schopnost najít analyticky řešení) a k tomu uplatnit procedurální dovednosti (vytvořit funkční program).

Někteří byli schopni uspět hned na první pokus, protože přípravu nepodcenili. Jiní byli nuceni opakovat a k samostudiu byli tak přinuceni. Zbylí se s náročností vůbec nechtěli smířit a onálepkovali předmět jako nepřiměřeně složitý. I přesto, že mnoho z nich ve výsledku zkoušku udělalo. Ti, co neuspěli, většinou neuspěli z důvodu, že nevyužili všechny možné pokusy. – Píši v minulém čase, protože v dalším roce muselo být zkoušení zjednodušeno.

Úspěšnost zvládnutí předmětu

Zadávané úkoly nebyly zcela triviální a bylo třeba prokázat alespoň konceptuální znalost na úrovni znalosti struktur. Přestože studenti potřebovali při obhajobě programu občas pomocnou berličku, zvládnutí úkolů u většiny v prvním roce potěšilo. Průměrná známka B odpovídala záměru spíše motivovat, přičemž rozložení zbylých známek kopírovalo Gaussovu křivku až po známku E. Některým k lepší známce pomohlo, že do výsledné známky se započítávala i známka ze vstupního testu.

Studentů, kteří neuspěli a museli si zapsat předmět příští rok bylo pět. Další studenti, kteří neuspěli ukončili z důvodu nesplnění dalších předmětů, někteří z nich studují znovu. Většina studentů, kteří neuspěli, nevyužili všechny tři pokusy, které využít mohli.

Stížnosti studentů na náročnost předmětu byly tak nečekaným překvapením.

Ukázalo, že mnozí studenti nejsou připraveni na zadání slovních úloh ze střední školy. Pro studenty z netechnických škol je samotné porozumění zadání někdy velkou překážkou a základní matematické vzorce je také velmi rozhodí.

V dalším roce muselo být zkoušení zjednodušeno na ověření procedurální dovedností spíše než principiálních znalostí, s tím musely být zjednodušeny zadávané slovní úlohy, aby neobsahovaly středoškolskou matematiku a také vyřazeny úlohy, které vyžadovali složitější analytické uvažování.

Diskuse

Schopnost vytvářet efektivní algoritmy vyžaduje nejenom procedurální dovednosti, ale také konceptuální porozumění na úrovni znalosti principů. Procedurální a principiální znalosti se navzájem posilují a vedou k dobrému růstu. Vyžadují nicméně jistý dril, jako při procvičování matematických úloh; programátor by řekl, „Bez programování se ještě nikdo programovat nenaučil.“ Je pochopitelné, proč algoritmizace nepatří mezi oblíbené předměty na školách – zmiňuje například Kalous (Kalous, 2025) –, které nejsou vyloženě matematicky orientované.

Závěr

Seminář je platforma, která má sloužit k jinému účelu než k doučování studentů základů. Pokud poslouží tam, kde je snaha sloučit dohromady obsah přednášky a cvičení, s tím, že přednáška by se stejně obsahově nenaplnila, nedá se očekávat, že by mohlo mít stejný efekt jako oddělená přednáška a cvičení. Perry uvádí podobný závěr při testování na žácích (Perry, 1991). Pro vyučujícího to pak může znamenat zklamání, když namísto očekávaného přenosu

procedurálních i principiálních znalostí volí studenti snadnější cestu a očekávají, že s tím uspějí také u zkoušky. Jednalo se o zásadní poučení vycházející ze seminářů v prvním roce.

Poté, co byla výuka předmětu Základy algoritmizace přizpůsobována studentům, bylo logické hledat příčiny také jinde než jen u sebe. Dále bylo tedy stanoveno několik bodů k dalším diskusím, které by mohly vést ke zkvalitnění výuky a celkově lepšímu českému školství:

Lepší připravenost přicházejících studentů ze středních škol, hlavně netechnického typu a učilišť. Nadějí se zdá být nový RVP (Nová informatika), na který školy přicházejí (Kalous, 2025).

Po přechodné období jsou řešením přípravné kurzy naprostých základů před začátkem semestru plus také kurz algebry s obsahem základů středoškolského učiva. Na chybějící neznalost těchto základů by se již v průběhu semestru nebral ohled.

Zlepšení organizaci veškeré výuky na škole a takzvané zdravé *sociální klima školy* (Allodi, 2010) s politikou školy, která je orientována na kvalitu spíše než na kvantitu.

S důrazem na duševní zdraví studentů se zvětšuje tlak na dobře vedenou koncepci výuky. S dobře vedenou koncepcí výuky se dá dosáhnout často dobrého výsledku při menším tlaku na studenty. To je způsob výuky, který funguje v mnoha rozvinutých zemích a přináší výsledky.

Konceptuální porozumění je potřeba na vysoké škole rozvíjet (nerezignovat) – dává totiž studentům flexibilitu, která jim umožňuje řešit širokou škálu problémů, a hlavně vidět příležitosti, kde lze naučené znalosti aplikovat. To je dnes důležitější než v minulosti. – Dělicí čára mezi vyšší odbornou školou a vysokou školou by měla být znatelnější pro učitele (způsobem výuky procedurální/konceptuální) i pro studenty (při výběru školy).

Zvážit používání AI. Může se zdát, že se jedná o výuku bohatou na relace, která posiluje konceptuální porozumění. To platí ale pouze v případě, že uživatel dokáže klást správné otázky – má již vytvořenou nějakou kostru znalostí.

Samozřejmostí by mělo být celoživotní kvalitní vzdělávání učitelů (nejlépe od zahraničních pedagogů) s širokým přesahem znalostí do jiných oborů jako jsou přírodní vědy, psychologie, sociologie a péče o duševní zdraví (EU-PROMENS, 2024). Kdo se sám rád nevzdělává, ať raději nikoho neučí.

Vyučující na technických vysokých školách by měli mít minimálně několikaměsíční praxi z výuky v posledním ročníku střední školy a absolvované pedagogické minimum, aby získali širší přehled.

Literatura

Matoušek, R. (2025) Hodnotící zpráva předmětu Základy algoritmizace 2023 – 2025. Zenodo, [online], [2025-11-30]. Dostupné z: <https://doi.org/10.5281/zenodo.17767918>

Pšenčíková, J. (2009) Algoritmizace. Vyd. 2. Kralice na Hané: Computer Media, 128 s. ISBN 978-80-7402-034-6.

Krathwohl, D. R. (2002). A revision of Bloom's taxonomy: An overview. In Theory into Practice, 41(4), 212–218.

Perry M. (1991). Learning and transfer: Instructional conditions of conceptual change. Cognitive Development, 6, 449-468.

Kalous, L. (2025) Nová koncepce výuky informatiky na gymnáziu ústí nad orlicí. In Informatika 2025 - Sborník příspěvků z konference. Jihlava: Vysoká škola polytechnická Jihlava, 2025, s. 23-27. ISBN 978-80-88064-75-6

Allodi, M. W. (2010). The meaning of social climate of learning environments: Some reasons why we do not care enough about it. In Learning Environments Research, 13(2), pp. 89-104.

EU-PROMENS. (2024). National Training Programme 2025. [online], [2025-11-30]. Dostupné z: <https://eu-promens.eu/eu-promens/pages/training>

Kontaktní údaje

Ing. Radek Matoušek

Katedra informačních technologií,

Fakulta elektrotechniky a informatiky UPCE

Studentská 95, 532 10 Pardubice

e-mail: radek.matousek@upce.cz

TESTY ŘÍZENÝ VÝVOJ A JEHO NASAZENÍ VE VÝUCE

TEST-DRIVEN DEVELOPMENT AND ITS APPLICATION IN TEACHING

Marek Musil

Abstrakt

Testy řízený vývoj (známý také jako Test-Driven Development) představuje moderní přístup k tvorbě softwaru, v němž hraje testování klíčovou roli ve všech fázích vývojového procesu. Výsledky testů poskytují vývojáři jistotu a zároveň poukazují na celkovou kvalitu vznikajícího produktu. S rostoucí komplexitou systémů a zvyšujícími se nároky na efektivní vývoj roste i význam kvalitního testování. Testování software je nezbytným prvkem pro přípravu budoucích softwarových inženýrů na profesní praxi, avšak studenti jej často vnímají jako málo atraktivní. Motivací studentů i efektivními didaktickými strategiemi se zabývá řada studií. Tento článek vybrané studie představuje a shrnuje jejich poznatky týkající se integrace testování softwaru do vzdělávacího procesu.

Klíčová slova: testování software, testy řízený vývoj, vzdělávání v testování software, vzdělávání v softwarovém inženýrství

Abstract

Test Driven Development is a modern approach to software development, in which testing plays a key role in all phases of the development process. Test results provide developers with confidence and at the same time indicate the overall quality of the resulting product. With the increasing complexity of systems and increasing demands for effective development, the importance of quality testing is also growing. Software testing is an essential element for preparing future software engineers for professional practice, but students often perceive it as unattractive. A number of studies deal with student motivation and effective didactic strategies. This article presents and summarizes their findings regarding the integration of software testing into the educational process.

Keywords: software testing, test-driven development, software testing education, software engineering education

Úvod

Testy řízený vývoj (Test Driven Development – TDD) patří mezi moderní přístupy k vývoji software a zdůrazňuje význam testování v celém životním cyklu aplikace. Testování software přispívá k odhalování chyb, zvyšuje spolehlivost systému a poskytuje vývojářům zpětnou vazbu už v raných fázích vývoje. Výsledek testování software dodává vývojáři odvahu a ukazuje na celkovou kvalitu vyvíjeného softwarového produktu. Testy řízený vývoj představuje nejen moderní přístup k tvorbě software, ale i výborný didaktický nástroj. Vede studenty k disciplinovanému přemýšlení, kvalitnímu návrhu a přirozenému osvojování profesionálních

návyků. Jeho zařazení do výuky umožňuje propojit teoretické znalosti s praxí a připravit budoucí IT odborníky na reálné požadavky trhu.

S rostoucí robustností moderních aplikací a tlakem na efektivní vývoj v softwarových firmách se význam testování dále zvyšuje. Proto je nezbytné toto téma systematicky zařazovat do výuky budoucích softwarových inženýrů takovým způsobem, aby byli připraveni porozumět významu testování i různým testovacím strategiím. V praxi se však ukazuje, že studenti často vnímají testování jako „nudný“, neatraktivní nebo okrajový prvek vývoje, což vede k jeho podceňování ve výuce. Testování software je opomíjeno ze strany vyučujících.

Zařazení testování software do výuky informatických oborů je diskutovaným tématem. Zvýšení motivace studentů, zatraktivnění výuky testování softwaru i efektivní integrace tohoto tématu do výuky představují oblasti, jimž se již věnovalo mnoho výzkumů, jmenujme některé, například (Clark, 2004; Krutz et al., 2014; Lauvas Jr and Arcuri, 2018; Deng et al., 2020; Wei, 2023). Tématu vývoje řízeném testy se pak věnují knihy (Farcic and Garcia, 2015; Garcia and Farcic, 2018). Tento článek si klade za cíl představit vybrané studie, shrnout jejich hlavní poznatky a zprostředkovat pohled na diskutovanou problematiku formou přehledných odkazů na relevantní výzkumy.

Proč je testování důležité?

Testování software je klíčová disciplína zajišťující kvalitu aplikací ještě před jejich nasazením do praxe. Jeho význam neustále roste:

- Snižuje riziko chyb, které mohou způsobit finanční ztráty i ohrozit bezpečnost.
- Umožňuje zvyšovat spolehlivost a použitelnost software.
- Podporuje rychlejší a efektivnější vývoj díky automatizaci a průběžnému testování.
- Stává se standardní součástí vývojových procesů (CI/CD), proto firmy potřebují kvalifikované testery i vývojáře s testovacími kompetencemi.

Problematika výuky testování software

Krutz et al. (2014) říkají, že téma testování software je v informatických osnovách často přehlíženo. Lauvas Jr and Arcuri (2018) uvádí studii (výzkum) tvrzením, že testování je kritickým aspektem vývoje software. Dále zmiňují, že příliš často je software vydáván s kritickými chybami. Je otevřenou výzvou, jak učit testování software efektivním způsobem, který dokáže studenty úspěšně motivovat. (Lauvas Jr and Arcuri, 2018)

Otázka atraktivní výuky testování software je řešena po mnoho let a ve vědecké oblasti diskutovaným tématem. V roce 2004, Clark představuje výsledky výzkumu *peer testingu* mezi vývojáři a testery prováděný po dobu několika let (od roku 1998). Výzkum se zaměřuje na využití peer testování v týmových projektech studentů softwarového inženýrství. Autorka studie zvažuje dva formáty peer testovacích aktivit – párové prohlídky kódu a skupinové inspekce, které studenti využívají k vzájemné kontrole a zlepšování svých projektů. Výzkum odhalil, že peer testování se ukázalo jako účinné nejen při odhalování chyb a zvyšování kvality odevzdaných projektů, ale také jako významný pedagogický nástroj. Pomáhá studentům lépe se učit, slouží jako přirozený milník v průběhu vývoje a podporuje spolupráci mezi týmy. Studie tak potvrzuje, že zapojení studentů do vzájemného testování má přínos pro kvalitu výsledků i pro rozvoj jejich profesních dovedností.

Problematiku nudného a neatraktivního tématu testování software ve výuce vyzdvihují studie (Krutz et al., 2014; Lauvas Jr and Arcuri, 2018). Autoři těchto studií explicitně konstatují, že studenti vidí toto téma jako nudný a nezajímavý úkol. Krutz et al. dále říkají, že vzdělávání se obvykle zaměřuje na tvorbu software, nikoliv na zajištění jeho kvality. Prezентují následující řešení. Jako inovace kurzu testování software, studenti kurzu si vybírají reálný projekt s otevřeným zdrojovým kódem, který budou následně během semestru testovat. V koordinaci s vyučujícím se od studentů očekává, že předloží plán testování, závěrečnou zprávu a několik prezentací v rámci celé třídy. Projekt dosáhl značného uznání studentů, a zpětná vazba ukázala zvýšenou spokojenost a splnění kurikulárních požadavků. Článek (Lauvas Jr and Arcuri, 2018) prezentuje systematickou literární rešerši zaměřenou na metody výuky testování software. Autoři analyzovali 30 studií publikovaných v letech 2013–2017 a identifikovali aktuální trendy ve vzdělávání této disciplíny. Výsledky ukazují několik směrů, z nichž jedním z nejvýraznějších je využívání gamifikace jako prostředku ke zvýšení atraktivity výuky a snížení vnímané monotónnosti testování.

Autoři Deng et al. (2020) se zaměřují na výuku testování software pomocí Free and Open Source Software (FOSS). Článek (2020) popisuje pět stále více zapojených vzdělávacích intervencí, které využívají FOSS v kurzu testování softwaru, aby studentům poskytly praktické znalosti a zkušenosti s testováním softwaru v reálném světě.

Výše uvedené studie jsou citovány v článku (Wei, 2023). Autoři se zaměřují na vývoj řízený testy a jako vyučující zdůrazňují potřebu efektivní výuky vývoje řízeném testy. Jako zásadní nedostatek (lack) uvádí, že toto téma je vyučováno v teoretické rovině a studenti pak používají tento přístup v projektech. Článek (Wei, 2023) představuje jednoduchý pracovní postup vývoje softwaru, který bezproblémově integruje zpřesňování požadavků, objektově orientovaný návrh a vývoj řízený testy. Součástí je také podrobná demonstrace živého kódování, která se podobá úkolům vývoje řízeného testy, se kterými se studenti v oboru setkají. Autoři doufají, že předloženou metodikou uplatněnou v kurzu programování studentům poskytnou praktické zkušenosti a pomohou jim rozvíjet dobré návyky v programování.

Výsledky a diskuze

Testy řízený vývoj patří mezi moderní softwarové přístupy, které propojují programování, kvalitu kódu i způsob uvažování vývojáře. Princip „nejdřív test, potom kód“ mění tradiční workflow a přináší řadu výhod – od zvýšení spolehlivosti až po lepší čitelnost a udržitelnost systémů. V oblasti vzdělávání se TDD ukazuje jako velmi efektivní pedagogický nástroj, protože studenty vede k systematickému přemýšlení, analytickému postupu a důrazu na kvalitu programového návrhu. V tomto článku byly představeny vybrané příspěvky, které se zabývají problematikou výuky testování software a nabízejí řešení vedoucí ke zvýšení motivace studentů.

Závěr

Testy řízený vývoj představuje potenciál využití nejen při vývoji všech typů aplikací. Měl by být vhodným způsobem integrován do výuky informatických oborů a studenti si tak osvojit návyky pro budoucí profesní práci.

Literatura

Clark, N., 2004. Peer testing in software engineering projects .

Deng, L., Dehlinger, J., Chakraborty, S., 2020. Teaching software testing with free and open source software, in: 2020 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), pp. 412–418. doi:10.1109/ICSTW50294.2020.00074.

Farcic, V., Garcia, A., 2015. Test-driven Java development. volume 284. Packt Publishing Birmingham.

Garcia, A., Farcic, V., 2018. Test-Driven Java Development - Second Edition: Invoke TDD principles for end-to-end application development. 2nd ed., Packt Publishing.

Krutz, D.E., Malachowsky, S.A., Reichlmayr, T., 2014. Using a real world project in a software testing course, in: Proceedings of the 45th ACM technical symposium on Computer science education, pp. 49–54.

Lauvas Jr, P., Arcuri, A., 2018. Recent trends in software testing education: A systematic literature review, in: Norsk IKT-konferanse for forskning og utdanning.

Wei, B., 2023. Teaching test-driven development and object-oriented design by example, in: 2023 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW), pp. 413–421. doi:10.1109/ICSTW58534.2023.00075.

Kontaktní údaje

Ing. Marek Musil

Katedra technických studií, Vysoká škola polytechnická Jihlava

Tolstého 16, 586 01 Jihlava

e-mail: marek.musil@vspj.cz

UČITELSTVÍ INFORMATIKY NA FAKULTĚ INFORMAČNÍCH TECHNOLOGIÍ ČVUT V PRAZE

EDUCATION OF SECONDARY SCHOOL TEACHERS IN INFORMATICS AT THE
FACULTY OF INFORMATION TECHNOLOGY CTU IN PRAGUE

Zdeněk Muzikář, Petr Zemánek

Abstrakt

Příspěvek se zabývá novým magisterským studijním programem Učitelství informatiky pro střední školy na fakultě informačních technologií ČVUT v Praze. Program se zaměřuje na obecné pedagogické, psychologické a didaktické disciplíny s tím, že oborová didaktika je úzce propojena s oborem informatika a soustřeďuje se na metody výuky a didaktické přístupy tak, aby studenti pochopili logiku a kognitivní procesy v daném oboru a byli tak schopni své budoucí žáky naučit rozvíjet IT myšlení.

Klíčová slova: vzdělávání v informatice, vzdělávání učitelů

Abstract

The article deals with the new master's degree program in *Informatics Teaching* for Secondary Schools at the Faculty of Information Technologies of the Czech Technical University in Prague. The program focuses on general pedagogical, psychological, and didactic disciplines, with the subject didactics being closely connected to the field of informatics and focusing on teaching methods and didactic approaches so that students understand the logic and cognitive processes in the field and are thus able to teach their future students to develop IT thinking.

Keywords: education in informatics, education of teachers

Úvod

Cílem navazujícího magisterského profesně zaměřeného studijního programu **Učitelství informatiky pro střední školy** je připravit kvalifikované učitele tohoto oboru pro střední školy a pomoci tak zacelit současný velký nedostatek kvalifikovaných učitelů. Učitelé naleznou uplatnění na různých typech středních škol, na gymnáziích, průmyslových i učňovských školách.

Studium je určeno pro absolventy bakalářského studia v oblasti informatiky. Studijní program se zaměřuje na obecné pedagogické, psychologické a didaktické disciplíny s tím, že **oborová didaktika je úzce propojena s oborem a soustřeďuje se na metody výuky a didaktické přístupy tak, aby studenti pochopili logiku a kognitivní procesy v daném oboru a byli tak schopni své budoucí žáky naučit rozvíjet oborové myšlení.**

Studijní program rozvíjí a prohlubuje oborové znalosti z bakalářské úrovně a pokrývá základní teoretická a profilující témata tak, **aby byl absolvent odborně způsobilý daný předmět**

vyučovat na střední škole; zároveň student získává kompetence odpovídající absolventu učitelského programu.

Studium spolu s praxí, jež v přípravě zahrnuje i **nácvik učitelských vystoupení a schopnost sebereflexe** a dále i s povinností absolvovat některé předměty v anglickém jazyce, by studium mělo přispět ke zkvalitnění výuky na středních školách, ke zvýšení zájmu středoškolských studentů o přírodovědné a technické obory a poskytnout kvalitní znalosti z oblasti informatiky pro absolventy středních škol, ať na pracovním trhu, tak při dalším studiu na vysokých školách.

Profil absolventa studijního programu

Absolvent čtyřsemestrálního navazujícího magisterského profesně zaměřeného studijního programu získává znalosti a kompetence potřebné pro výkon povolání učitele informatiky na středních školách. **Studijní program respektuje požadavky školské politiky na přípravu učitelů pro regulovaná povolání (Rámcové požadavky - čj. MSMT- 4626/2024- 2).**

Absolvent získává odpovídající znalosti z psychologie, pedagogiky a didaktiky, dále své didaktické schopnosti prohlubuje v rámci oborové didaktiky informatiky. Nastavení studijního programu vychází z kompetenčního rámce absolventa učitelství. Své oborové znalosti z bakalářského studia rozvíjí v rámci základních teoretických a profilujících odborných předmětů.

Předměty učitelské propedeutiky, oborové didaktiky, oboru i praxe zohledňují kompetenční rámec absolventů učitelství viz [1].

V průběhu studia musí absolvovat i **předměty v angličtině (6 ECTS)** a absolvovat učitelskou praxi v rozsahu 720 hodin (24 ECTS). Absolvent tak získává odborné kompetence v oblasti informatiky, je schopen dalšího sebevzdělávání, umí komunikovat v anglickém jazyce **a má pedagogicko-psychologické didaktické znalosti a dovednosti potřebné pro moderní způsoby efektivního vedení výuky a podchycení zájmu studentů o obor.**

Pravidla pro tvorbu studijních plánů

Studijní program respektuje požadavky školské politiky na přípravu učitelů pro regulovaná povolání. Předměty jsou proto členěny **v souladu s Rámcovými požadavky na studijní programy navazujícího magisterského studia pro SŠ**, jejichž absolvováním se získává odborná kvalifikace k výkonu regulovaných povolání pedagogických pracovníků čj. MSMT- 4626/2024- 2).

Tabulka 1: Podíl složek přípravy, kredity a hodiny přípravy

Podíl složek přípravy	Kredity doporučené	Kredity realizované	Hodiny
Učitelská propedeutika	30	30	900
Oborová didaktika	30	32 (+2)	900
Obor	12	19 (+7)	360 + 270
Praxe	24	24	720
Disponibilní kredity	24	15	660
Celkem	120	120	3600

Disponibilní kredity (24 ECTS) byly rozděleny takto:

Oborová didaktika: 2 ECTS
 Předmět v angličtině: 6 ECTS
 Diplomová práce: 9 ECTS
 Obor: 7 ECTS
 Celkem 24 ECTS

Tabulka 2: Vzorový studijní plán pro prezenční studium – ECTS:

Semestr	PP Uč. prop.	PV Uč. prop.	PV St. základ (angl.)	Obor. didak.	Obor	Praxe	Dipl. práce	Celkem
1.	18			6	6			30
2.			6	11	7	6		30
3.	3			3	6	18		30
4.	3	6		12			9	30
Celkem	24	6	6	32	19	24	9	120

Pozn. PP – povinné předměty, PV povinně volitelné

Zdroj: [2]

Tabulka 3: Vzorový studijní plán pro kombinované studium ECTS/kontaktní výuka:

Semestr	PP Uč. prop.	PV Uč. prop.	PV St. základ (angl.)	Obor. didak.	PP Obor	Praxe	Dipl. práce	Celkem
1.	18/56			6/18	6/18			30/92
2.	-	-	6/18	11/34	7/20	6/8		30/80
3.	3/8			3/2	6/20	18/8		30/38
4.	3/8	6/16		12/36			9	30/60
Celkem	24/72	6/16	6/18	32/90	19/58	24/16	9	120/270

Zdroj: [2]

Číselné údaje vyznačují ECTS/B – **počet blokových setkání**. 1B=45 minut (1 vyučovací hodina). Počet blokových setkání tvoří v prvním semestru 92, ve druhém semestru 80, ve třetím semestru, kde již probíhá i praxe na školách, je bloková výuka jen 38 hodin.

Ve čtvrtém semestru student **píše diplomovou práci**, tudíž setkání je jen 60. Bloková výuka v rámci je praxe je vztahována pouze na propedeutikum praxe a reflexi praxe, nikoli na uskutečňování praxe na školách.

Podmínkou k přijetí ke studiu je absolvování bakalářského studijního programu v oboru informatika, případně příbuzného oboru. Přijímací komise může v případě, že bakalářské studium nezahrnovalo nějakou tematiku, jejíž absolvování se předpokládá, či uplynula již delší doba od absolvování studia, přikázat absolvování některého bakalářského předmětu s tematikou teoretického či profilujícího základu z našeho bakalářského programu Informatika.

Studijní plány programu

Tabulka 4: Studijní plány programu

B-IIa – Studijní plány a návrh témat prací (bakalářské a magisterské studijní programy)					
Označení studijního plánu	Učitelství informatiky pro střední školy, profesně orientované, prezenční				
Povinné předměty – studijní základ					
Název předmětu	Rozsah	způsob ověření	počet kreditů	doporučený ročník/semestr	profilující základ
Role učitele v moderní společnosti	26P+0C	Zk	3	1/1	
Obecná pedagogika	26P+13C	Z, Zk	5	1/1	PZ
Psychologie v edukačním procesu	26P+13C	Z, Zk	5	1/1	PZ
Obecná didaktika	26P+13C	Z, Zk	5	1/1	ZT
Metody společensko-vědního výzkumu	13P+13C	Z, Zk	3	2/3	
Sociální pedagogika	26P+0C	Zk	3	2/4	
Celkem			24		
Povinné předměty – Oborová didaktika					
Technologie počítačů	26P+26C	Z, Zk	6	1/1	
Didaktika informatiky I	13P+ 26C	Z, Zk	6	1/2	PZ
Informační bezpečnost	26P + 26C	Z, Zk	5	1/2	

Semestrální projekt k diplomové práci	90 hodin	Z	3	2/3	
Moderní technologie a umělá inteligence	26P+26C	Z, Zk	5	2/4	
Didaktika informatiky II	13P+26C	KZ	7	2/4	PZ
Celkem			32		
Povinné předměty – oborové					
Počítače a sítě	26P+26C	Z, Zk	6	1/1	PZ
Algoritmy a datové struktury	26P+26C	Z, Zk	7	1/2	ZT
Vývoj SW aplikací	26P+26C	Z, Zk	6	2/3	PZ
Celkem			19		
Diplomová práce	270 hodin		9	2/4	
Povinné předměty – Praxe					
Propedeutikum pedagogické praxe	26P+26C	KZ	6	1/2	
Souvislá pedagogická praxe	450 hodin	KZ	15	2/3	
Reflexe pedagogické praxe	13P+13C	Z	3	2/3	
Celkem	720		24		
Povinně volitelné předměty – Skupina 1:					
Předměty v angličtině I		6		2/3	
Podmínka pro splnění této skupiny předmětů: získat 6 kreditů. Student si volí z aktuální nabídky fakultních magisterských předmětů v angličtině, do této skupiny se započítávají i předměty absolvované v rámci programu ERASMUS, EuroTeQ					
Povinně volitelné předměty – Skupina 2: učitelská propedeutika					
Sociální a pedagogická komunikace	0P+26C	KZ	3	2/4	
Psychohygiena v práci učitele	13P+13C	Z,Zk	3	2/4	
Společenská rizika moderních komunikačních technologií	13P+13C	Z, Zk	3	2/4	

Syndrom rizikového chování ve školách	13P+13C	KZ	3	2/4	
Didaktika laboratoří	0P+26C	KZ	3	2/4	
Osobnost, patologie, normalita	13P + 13C	KZ	3	2/4	
Podmínka pro splnění této skupiny předmětů: Získat 6 kreditů. Student si volí předměty z nabídky.					
Součásti SZZ a jejich obsah					
Státní závěrečná zkouška bude mít 3 části: 1) Obhajoba diplomové práce 2) Učitelské propedeutika: Vstupují předměty: Obecná psychologie, obecná pedagogika, obecná didaktika 3) Obor studia + oborová didaktika Vstupují předměty: Obor: Algoritmy a datové struktury, Vývoj SW aplikací					

Zdroj: [2]

Závěr

Jsme převědčeni, že nový studijní program ***Učitelství informatiky pro střední školy*** přispěje k výchově nových učitelů informatiky na středních školách.

Velmi uvítáme připomínky a zkušenosti (s výukou učitelů informatiky) od našich kolegů z jiných vysokých škol i názory aktuálních učitelů informatiky ze středních škol.

Literatura

[1] Kompetenční rámec absolventa a absolventky učitelství Společné profesní kompetence (Verze platná od října 2023). Praha, MŠMT 2023

Dostupné z: https://msmt.gov.cz/uploads/kompetencni_ramec_absolventa_2023_10.pdf

[2] ŽÁDOST O AKREDITACI MAGISTERSKÉHO PROFESNĚ ZAMĚŘENÉHO PREZENČNÍHO A KOMBINOVANÉHO STUDIJNÍHO PROGRAMU UČITELSTVÍ INFORMATIKY PRO STŘEDNÍ ŠKOLY. Praha, FIT ČVUT 2024 (dostupné od autorů příspěvku).

Kontaktní údaje

Ing. Zdeněk Muzikář, CSc.
proděkan pro studium a pedagogickou činnost
katedra počítačových systémů
Fakulta informačních technologií ČVUT v Praze
Thákurova 9, 160 00 Praha 6
e-mail: muzikar@fit.cvut.cz

doc. RNDr. Ing. Petr Zemánek, CSc.
katedra počítačových systémů
Fakulta informačních technologií ČVUT v Praze
Thákurova 9, 160 00 Praha 6
e-mail: muzikar@fit.cvut.cz

VÝUKA V ÉŘE AI: OD TEORIE K PRAXI PROSTŘEDNICTVÍM SAAS A SPOLUPRÁCE S FIRMAMI

TEACHING IN THE AI ERA: FROM THEORY TO PRACTICE THROUGH SAAS AND INDUSTRY COLLABORATION

Lukáš Novák, Jan Luhan

Abstrakt

Příspěvek se zaměřuje na proměnu výuky informatických předmětů v éře generativní umělé inteligence. Tradiční důraz na teoretické projekty je postupně nahrazován praktickým přístupem, který studentům umožňuje vyzkoušet si reálné nástroje a postupy díky dostupnosti SaaS platform. Klíčovým prvkem se stává spolupráce s praxí prostřednictvím stáží a zvaných přednášek odborníků z firem. Článek představí zkušenosti s implementací tohoto přístupu do výuky a ukáže, jak může kombinace AI, SaaS a firemní spolupráce zvýšit motivaci studentů a lépe je připravit na reálné potřeby trhu práce.

Klíčová slova: umělá inteligence ve výuce, SaaS platformy, spolupráce s firmami, motivace studentů

Abstract

The paper focuses on the transformation of teaching computer science courses in the era of generative artificial intelligence. The traditional emphasis on theoretical projects is gradually being replaced by a practical approach that allows students to experiment with real tools and methods through the availability of SaaS platforms. A key element is collaboration with industry, enabled through internships and guest lectures by company experts. The paper presents experiences with implementing this approach in teaching and demonstrates how the combination of AI, SaaS, and industry collaboration can enhance student motivation and better prepare them for the real needs of the labor market.

Keywords: artificial intelligence in education, SaaS platforms, industry collaboration, student motivation

Úvod

Dynamický rozvoj informačních technologií, zejména v oblasti umělé inteligence, cloudu a softwaru poskytovaného jako služba, výrazně ovlivňuje způsob, jakým se dnes vyučují informatické a manažersko-informatické disciplíny. Tradiční model výuky, založený převážně na teoretických znalostech a simulovaných příkladech, již nedokáže plně reagovat na potřeby praxe ani na očekávání současné generace studentů. Ti jsou zvyklí pracovat s dostupnými technologiemi, experimentovat a okamžitě vidět výsledky své práce.

Tento článek představuje přístup, který je uplatňován na Fakultě podnikatelské Vysokého učení technického v Brně na Ústavu informatiky, a ukazuje, jak lze propojit tradiční výuku podnikových informačních systémů s moderními technologickými trendy a potřebami praxe.

Shrnutí dosavadních poznatků

Rychlá adopce generativní umělé inteligence (GenAI) proměňuje didaktiku informatických předmětů. Zjištění z vysokých škol ukazují, že učitelé už revidují zadání a způsob hodnocení. Nejčastější změnou jsou úpravy samostatných nebo týmových projektů právě kvůli dostupnosti nástrojů, jako je ChatGPT (Lee et al., 2024). Tento posun vyvolává dvě paralelní potřeby: chránit akademickou integritu a zvyšovat autenticitu zadání tak, aby studenti rozvíjeli kompetence využitelné v praxi (U.S. Department of Education, 2023).

Vliv GenAI na motivaci a učení

Data z více zemí naznačují, že studenti vnímají GenAI jako akcelerator produktivity a kreativity, ale zároveň přiznávají limity spolehlivosti a nepřesnosti (Fan et al., 2025). Didaktické studie současně ukazují, že autenticita učebních materiálů a zadání posiluje zájem a zapojení studentů, tedy klíčové faktory motivace. To je v souladu s širší literaturou o problem-based či project-based učení, které prokazatelně zvyšuje angažovanost i dovednosti (Condliffe et al., 2017).

Studenti se podle výzkumů nejvíce učí tehdy, když mají možnost aktivně konstruovat své poznání prostřednictvím řešení reálných problémů a projektů. Právě modely *problem-based learning* a *project-based learning* vycházejí z přístupu, který klade důraz na samostatné objevování, týmovou spolupráci a reflexi (Condliffe et al., 2017). Vysokoškolské studie potvrzují, že tyto přístupy vedou k vyšší úrovni porozumění, lepšímu zapamatování a rozvoji přenositelných dovedností, jako jsou analytické myšlení, komunikace a řešení problémů (Thomas, 2000; Strobel & van Barneveld, 2009).

V kontextu nástupu generativní umělé inteligence získávají tyto metody nový rozměr. AI může sloužit jako facilitátor učení – například při generování nápadů, simulaci rolí, analýze dat nebo poskytování okamžité zpětné vazby (Holmes et al., 2023). Tím se posiluje autonomní učení a kreativní přístup studentů, aniž by docházelo ke ztrátě didaktické hodnoty úlohy. Naopak, integrace AI umožňuje studentům kriticky reflektovat limity umělé inteligence, hodnotit kvalitu jejích výstupů a rozvíjet tzv. *AI literacy* – tedy schopnost rozumět, hodnotit a efektivně využívat nástroje umělé inteligence. Tyto dovednosti se stávají zásadní pro budoucí zaměstnatelnost a jsou plně v souladu s cíli moderní vysokoškolské výuky orientované na praxi.

Cloudové a SaaS nástroje ve výuce informatických předmětů

Rozvoj cloud computingu, tedy poskytování výpočetních a softwarových služeb prostřednictvím internetu, zásadně ovlivnil přístup k výuce informačních a komunikačních technologií (ICT). Model Software as a Service (SaaS), který umožňuje využívat softwarové aplikace bez nutnosti jejich instalace a správy na lokálních zařízeních, se stal jedním z klíčových pilířů ICT sektoru a posléze i nástrojů moderní výuky (Armbrust et al., 2010; Sultan, 2014).

Dostupnost cloudových řešení výrazně snižuje technické a finanční bariéry při výuce. Studenti mají možnost pracovat s profesionálními nástroji, které dříve vyžadovaly složitou instalaci, licenční správu či provoz vlastní infrastruktury. Prostřednictvím SaaS platformy lze realizovat

praktické úlohy a projekty v prostředí, které odpovídá skutečné praxi v podnicích – od správy datových úložišť a integrace aplikačních rozhraní API (Application Programming Interface) až po automatizaci procesů a tvorbu analytických nástrojů.

Zkušenosti z období pandemie onemocnění COVID-19 ukázaly, že právě cloudové služby představují spolehlivý rámec pro zajištění kontinuity výuky. Mnoho univerzit implementovalo cloudové platformy pro sdílení informací a řízení projektů, čímž umožnilo studentům spolupracovat na společných úlohách bez ohledu na místo a čas. Tento trend přetrvává i v postpandemickém období – cloudové prostředí se stalo standardní součástí výuky a nástrojem pro rozvoj digitálních kompetencí studentů (Bond et al., 2021).

Podle aktuálních výzkumů přináší integrace SaaS do vysokoškolského prostředí významné výhody zejména v oblasti efektivity výuky, flexibility a přístupu k aktuálním technologiím (Azubuike, 2025). Zároveň vyžaduje od institucí nový přístup k otázkám správy dat, kybernetické bezpečnosti a integrace různých cloudových systémů (Marinescu, 2017).

Spolupráce s praxí: stáže a zvané přednášky

Empirické studie dokládají, že výuka obohacená o interakce s firmami významně zvyšují angažovanost studentů (+ 30 % zapojení; + 35 % spokojenost) (Naseer et al., 2025). Specificky k zvaným přednáškám se objevuje studie, podle níž hosté (zejména alumni a odborníci z praxe) zvyšují vnímané přínosy daného předmětu (Pepple, 2025).

Takové zapojení odborníků přináší několik klíčových přínosů. Za prvé, studenti získávají možnost konfrontovat teoretické poznatky s reálnou praxí a pochopit, jak jsou probírané koncepty aplikovány v reálném podnikovém prostředí. Za druhé, přítomnost odborníků z firem posiluje vnímání relevance studovaného oboru a zvyšuje vnitřní motivaci studentů k aktivnímu zapojení do výuky. Studie potvrzují, že studenti hodnotí nejlépe ty předměty, které propojují teoretické znalosti s praktickými zkušenostmi odborníků.

Dalším přínosem je rozvoj přenositelných kompetencí, jako je týmová spolupráce, profesní komunikace a porozumění očekáváním zaměstnavatelů. Programy, které kombinují akademickou výuku s reálnými projekty nebo krátkodobými stážemi, prokazatelně zvyšují připravenost absolventů na trh práce a posilují jejich sebedůvěru při řešení komplexních úloh (Jackson & Bridgstock, 2021). Z pedagogického hlediska lze tuto formu spolupráce zařadit do rámce tzv. *experiential learning* (zkušenostního učení), které zdůrazňuje aktivní zapojení studentů, reflexi a aplikaci teoretických poznatků (Kolb, 2015).

Implikace pro výuku

Rozvoj generativní umělé inteligence, cloudových služeb a zapojení podnikové praxe přináší zásadní proměnu ve způsobu, jakým se dnes vyučují informatické předměty. Tradiční pojetí, založené na teoretických úlohách a izolovaných projektech, ustupuje přístupům, které více reflektují reálné potřeby praxe, podporují samostatnost studentů a rozvíjejí jejich digitální i měkké dovednosti.

Klíčové poznatky z výzkumu

Z provedené rešerše i praktických zkušeností vyplývá několik hlavních závěrů.

- **Umělá inteligence** mění způsob, jakým studenti přistupují k řešení problémů. Slouží jako podpůrný nástroj, který rozšiřuje jejich kreativitu a produktivitu, nikoli jako náhrada lidského myšlení.
- **Model výuky založený na projektech a problémech** vede k hlubšímu porozumění látce, posiluje motivaci a podporuje dlouhodobé zapamatování.
- **Cloudové a SaaS nástroje** umožňují realizovat praktické úlohy i bez složité infrastruktury a přibližují výuku reálnému podnikovému prostředí.
- **Spolupráce s firmami** – formou stáží, hostovaných přednášek či konzultací – přináší studentům autenticitu, profesní zkušenost a posiluje jejich připravenost na trh práce.

Doporučení pro praxi

Z hlediska pedagogické praxe se ukazuje jako efektivní:

- Zařazovat úlohy, které propojují teoretické poznatky s reálnými daty a případy z praxe.
- Umožnit studentům využívat dostupné nástroje umělé inteligence a současně klást důraz na jejich kritické hodnocení.
- Zapojovat odborníky z podnikové sféry přímo do výuky – ať už formou hostujících přednášek, konzultací projektů nebo krátkodobých workshopů.
- Podporovat týmovou spolupráci, rozvoj sebereflexe a schopnost plánovat a řídit vlastní učení.

Diskuze

Autoři působí na Vysokém učení technickém v Brně, Fakultě podnikatelské, na Ústavu informatiky, který zastřešuje bakalářské a magisterské studijní programy Manažerská informatika a Informační management. Autor článku je garantem a vyučujícím předmětů Podnikové informační systémy 1, Podnikové informační systémy 2 a Management informačních systémů, které na fakultě vyučuje již více než deset let.

Využití cloudových a SaaS platforem

Jedním z nejvýznamnějších trendů posledních let je přechod od tradičních lokálních (on-premise) řešení ke cloudovým a SaaS (Software as a Service) aplikacím. Tyto platformy umožňují studentům „osahat si“ technologie, které dříve nebylo možné během výuky jednoduše implementovat či provozovat. V rámci výuky se tak studenti setkávají s podnikovými systémy, jako jsou Oracle NetSuite, Salesforce nebo integrační platforma Keboola, které poskytují realistické prostředí pro procvičování principů podnikové informatiky.

Řada dalších nástrojů navíc podporuje akademickou sféru – buď poskytováním licencí zdarma pro nekomerční použití (např. UiPath, VMware), nebo formou volných kreditů v tzv. spotřebním modelu cloudových služeb (např. Microsoft Azure). Studenti tak získávají přístup nejen ke studijním materiálům, které firmy volně sdílejí, ale i k prostředkům a technologiím, jež firmy skutečně využívají ve své každodenní činnosti. To zásadně zvyšuje autenticitu výuky a umožňuje studentům experimentovat s reálnými scénáři.

Propojení s praxí: hosté z firem a studentské stáže

Na teoretickou část výuky navazuje systematická spolupráce s podnikovou praxí, která probíhá na dvou úrovních.

První úroveň představují zvané přednášky odborníků z praxe, které se autorovi článku dlouhodobě daří organizovat ve spolupráci se seniorskými zástupci firem, jež patří mezi lídry ve svých oborech – například SAP, Oracle NetSuite, Innogy, Kaufland či Kiwi.com. Tito experti přinášejí do výuky reálné příklady z praxe a potvrzují, že teoretické poznatky vyučované na fakultě mají přímé uplatnění. Takové interakce zároveň posilují motivaci studentů a jejich zájem o obor.

Druhou úroveň spolupráce tvoří studentské stáže a internshipy, které jsou často propojeny s vypracováním závěrečných prací na reálných firemních zadáních. Spolupráce s firmami jako Legito, Innogy nebo PwC umožňuje studentům zapojit se do skutečných projektů, poznat firemní kulturu a lépe porozumět aktuálním výzvám, které podniky řeší. Tato zkušenost jim následně usnadňuje přechod na trh práce a výrazně zvyšuje jejich uplatnitelnost.

Přizpůsobení výuky éře umělé inteligence

Posledním, avšak stále významnějším faktorem, který ovlivňuje podobu výuky, je nástup generativní umělé inteligence (GenAI). Je iluzorní předpokládat, že jakýkoli zákaz ze strany vyučujícího dokáže studentům zabránit tyto nástroje používat. Výuka se proto musí změnit a přizpůsobit – stejně jako to dělají firmy na trhu.

Autor článku se proto postupně odklání od zadávání rozsáhlých semestrálních prací a klade větší důraz na praktickou práci s technologiemi. Studenti si sami vybírají konkrétní řešení (např. SaaS, PaaS nebo IaaS platformy), tato řešení testují a následně své zkušenosti prezentují spolužákům v rámci krátkých referátů či demonstračních projektů. Cílem je, aby studenti nové přístupy a technologie nejen teoreticky popsali, ale skutečně pochopili a použili v praxi. V době, kdy dokáže nástroj umělé inteligence během několika minut vygenerovat kvalitní semestrální práci, je důležitější než kdy dřív rozvíjet schopnost porozumět, aplikovat a kriticky hodnotit moderní technologie, nikoli pouze o nich psát.

Závěr

Informační technologie se vyvíjejí nebyvalou rychlostí. I když oblast podnikové informatiky reaguje na nové trendy o něco pomaleji než jiné části IT sektoru, i zde je nutné výuku průběžně modernizovat a přizpůsobovat současné praxi. Studenti dnes potřebují nejen teoretické znalosti, ale především schopnost orientovat se v aktuálních technologiích a nástrojích, které jsou používány ve firmách.

Praktická výuka prostřednictvím SaaS přináší několik didaktických přínosů. Za prvé, studenti získávají přímou zkušenost s technologiemi, které se běžně používají v praxi. Za druhé, okamžitá dostupnost nástrojů umožňuje vyučujícím zaměřit se více na koncepční porozumění a méně na řešení technických problémů s instalací a kompatibilitou. A konečně, díky možnosti práce v týmech na reálných datech a úlohách se přirozeně rozvíjejí i měkké dovednosti – spolupráce, komunikace a řízení projektů.

Z pedagogického hlediska lze využití SaaS považovat za přirozený krok směrem k tzv. autentické výuce, tedy k výuce, která studentům umožňuje řešit úkoly podobné těm, s nimiž se setkají ve svém profesním životě. Kombinace generativní umělé inteligence a cloudových

nástrojů vytváří nové příležitosti pro individualizaci učení, okamžitou zpětnou vazbu a adaptivní hodnocení.

V současném vysokoškolském prostředí je tedy klíčové vytvářet platformy, které umožňují dlouhodobější spolupráci mezi univerzitami a firmami – ať už formou projektových partnerství, studentských stáží, nebo systematicky organizovaných hostujících přednášek.

Poděkování

Případný odkaz na grant či projekt.

Literatura

Článek v časopise (online verze časopisu):

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A View of Cloud Computing. *Communications of the ACM*, 53(4), 50–58. <https://doi.org/10.1145/1721654.1721672>

Bond, M., Bedenlier, S., Marín, V. I., & Händel, M. (2021). Emergency remote teaching in higher education: Mapping the first global online semester. *International Journal of Educational Technology in Higher Education*, 18, 50. <https://doi.org/10.1186/s41239-021-00282-x>

Fan, L., et al. (2025). Educational impacts of generative artificial intelligence on learning and performance of engineering students in China. *Scientific Reports*, 15, 26521. <https://doi.org/10.1038/s41598-025-06930-w>

Jackson, D., & Bridgstock, R. (2021). What Actually Works to Enhance Graduate Employability? The Relative Value of Curricular, Co-curricular, and Extra-curricular Learning and Networking. *Higher Education*, 81(4), 723–739. <https://doi.org/10.1007/s10734-020-00544-2>

Lee, D., Arnold, M., Srivastava, A., Plastow, K., Strelan, P., Ploeckl, F., Lekkas, D., & Palmer, E. (2024). The Impact of Generative AI on Higher Education Learning and Teaching: A Study of Educators' Perspectives. *Computers and Education: Artificial Intelligence*, 6(2), 100221. <https://doi.org/10.1016/j.caeai.2024.100221>

Naseer, F., Tariq, R., Alshahrani, H. M., Alruwais, N., & Al-Wesabi, F. N. (2025). Project based learning framework integrating industry collaboration to enhance student future readiness in higher education. *Scientific Reports*, 15, 24985. <https://doi.org/10.1038/s41598-025-10385-4>

Pepple, D. G. (2025). Using Guest Lectures to Enhance Student Employability: Pedagogical Considerations. *Cogent Education*, 12(1), 2452076. <https://doi.org/10.1080/2331186X.2025.2452076>

Azubuike, C., Sule, A. K., Adepoju, P. A., Ikwuanusi, U. F., & Odionu, C. S. (2025). Integrating SaaS products in higher education: Challenges and best practices in enterprise architecture. *International Journal of Research and Scientific Innovation*, 11(12), 948–957. <https://doi.org/10.51244/IJRSI.2024.11120082>

Sultan, N. (2014). Cloud Computing for Education: A New Dawn? *International Journal of Information Management*, 34(2), 109–116. <https://doi.org/10.1016/j.ijinfomgt.2013.09.002>

Strobel, J., & van Barneveld, A. (2009). When Is PBL More Effective? A Meta-Synthesis of Meta-Analyses Comparing PBL to Conventional Classrooms. *Interdisciplinary Journal of Problem-Based Learning*, 3(1), 44–58. <https://doi.org/10.7771/1541-5015.1046>

Knižní publikace:

Holmes, W., Porayska-Pomsta, K., & Bialik, M. (2023). *Artificial Intelligence in Education: Promises and Implications for Teaching and Learning*. MA: Center for Curriculum Redesign.

Kolb, D. A. (2015). *Experiential Learning: Experience as the Source of Learning and Development* (2nd ed.). Pearson Education.

Marinescu, D. C. (2017). *Cloud Computing: Theory and Practice* (2nd ed.). Morgan Kaufmann.

Condliffe, B., Quint, J., Visher, M. R., Bangser, M. R., Drohojowska, S., Saco, L., & Nelson, E. (2017). *Project-Based Learning: A Literature Review*. MDRC. Retrieved from https://www.mdrc.org/sites/default/files/Project-Based_Learning-LitRev_Final.pdf

Thomas, J. W. (2000). *A Review of Research on Project-Based Learning*. Autodesk Foundation.

U.S. Department of Education, Office of Educational Technology. (2023). *Artificial Intelligence and the Future of Teaching and Learning: Insights and Recommendations*. Washington, DC.

Kontaktní údaje

Ing. Lukáš Novák, Ph.D.
Vysoké učení technické v Brně
Fakulta podnikatelská
Kolejní 2906/4, 612 00 Brno
e-mail: Lukas.Novak2@vut.cz

AUTOMATIZACE VIZUALIZACE GEOTAGOVANÉ FOTODOKUMENTACE V PROSTŘEDÍ QGIS: PROPOJENÍ VÝZKUMU V OBLASTI HERITAGE SCIENCE S VÝUKOU APLIKOVANÉ INFORMATIKY

AUTOMATION OF GEOTAGGED PHOTO DOCUMENTATION VISUALIZATION IN
QGIS: CONNECTING HERITAGE SCIENCE RESEARCH WITH APPLIED INFORMATICS
EDUCATION

Jakub Novotný, Alžběta Frýdlová

Abstrakt

Příspěvek se zabývá problematikou efektivní vizualizace prostorových dat v oblasti Heritage Science, konkrétně geotagované fotodokumentace. Představuje vývoj zásuvného modulu (pluginu) SurveyWebMap Exporter pro open-source platformu QGIS. Tento nástroj vznikl jako výsledek spolupráce mezi výzkumným pracovištěm ÚTAM AV ČR a studentkou oboru Aplikovaná informatika (VŠPJ) v rámci odborné praxe. Cílem bylo automatizovat proces importu, zpracování a webové publikace terénních dat, přičemž důraz byl kladen na vizualizaci směru pohledu (azimutu) a přenositelnost výsledné mapy bez nutnosti serverové infrastruktury. Příspěvek popisuje technické řešení pluginu, využití knihoven jazyka Python pro práci s EXIF metadaty a pedagogický přínos této spolupráce v kontextu projektově orientované výuky (Project-Based Learning).

Klíčová slova: QGIS, Python, Heritage Science, geotagovaná fotodokumentace, Leaflet, odborná praxe

Abstract

This paper addresses the efficient visualization of spatial data in Heritage Science, specifically geotagged photographic documentation. It presents the development of the SurveyWebMap Exporter plugin for the open-source QGIS platform. This tool was created through collaboration between the Institute of Theoretical and Applied Mechanics of the Czech Academy of Sciences and an Applied Informatics student during a professional internship. The aim was to automate the import, processing and web publishing of field data, with an emphasis on visualizing the camera viewing direction (azimuth) and producing a portable offline web map without server infrastructure. The paper describes the technical solution, the use of Python libraries for EXIF metadata processing, and the educational contribution of this collaboration in the context of project-based learning.

Keywords: QGIS, Python, Heritage Science, photographic documentation, Leaflet, internship

Úvod

Oblast Heritage Science, interdisciplinární obor propojující přírodní, technické a humanitní vědy za účelem ochrany kulturního dědictví, je závislá na precizní dokumentaci. Ať už se jedná

o monitoring historických budov nebo archeologický průzkum, primární metodou záznamu zůstává fotografie. S nástupem moderních technologií se stalo standardem pořizování geotagovaných fotografií, které v metadatech (EXIF) nesou informace o poloze (GPS souřadnice) a často i o směru pořízení snímku (azimut/kompas).

Tato data mají obrovský potenciál pro vizualizaci v geografických informačních systémech (GIS), kde umožňují přesnou prostorovou identifikaci pořízených snímků (Kerski 2020). V praxi však narážíme na problém „poslední míle“ – jak tato data efektivně předat koncovému uživateli (památkáři, investorovi), který často nedisponuje specializovaným GIS softwarem. Standardní předání složky s fotografiemi postrádá prostorový kontext a komplexní GIS projekty jsou pro laiky uživatelsky náročné.

Tento článek prezentuje řešení tohoto problému prostřednictvím zásuvného modulu „SurveyWebMap Exporter“ pro QGIS. Nástroj automatizuje převod terénních dat do interaktivní webové mapy. Unikátnost projektu spočívá v jeho genezi – vznikl primárně jako výstup odborné praxe studentky oboru Aplikovaná informatika Vysoké školy polytechnické Jihlava (VŠPJ), což demonstruje efektivní propojení akademického výzkumu s výukou.

Kontext a analýza existujících řešení

Platforma QGIS je v oblasti Heritage Science široce využívána díky své rozšiřitelnosti prostřednictvím zásuvných modulů v jazyce Python. Pro publikaci map na web existují zavedené nástroje, které však pro specifické potřeby fotodokumentace vykazují určité limity:

- qgis2web: Univerzální nástroj pro export QGIS projektů do webové mapy (Leaflet/OpenLayers). Pro účely jednoduché fotodokumentace je však generovaný kód a výsledná struktura často zbytečně komplexní. Při práci s fotografiemi (např. jako přílohami či v pop-up oknech) navíc může být nutné dodatečně řešit cesty k souborům a přenositelnost mezi počítači (QGIS2WEB 2024).
- ImportPhotos: Plugin zaměřený na import geotagovaných fotografií do QGIS a extrakci vybraných metadat. Neřeší však vytvoření a balíčkování offline webové prezentace; pro webový export je obvykle nutné využít další nástroje (Kyriakou 2019).
- Serverová řešení (QGIS Cloud, Lizard): Vyžadují nahrání dat na server, což může být limitující v terénu bez připojení k internetu nebo při práci s citlivými daty, která nesmí opustit interní síť.

Na základě této analýzy jsme definovali požadavek na vytvoření lehkého, jednoúčelového nástroje („single-purpose tool“), který zajistí:

- Nezávislost: Přímé čtení EXIF metadat bez závislosti na externích nástrojích či ručním předzpracování.
- Vizualizace azimutu: Automatické natočení symbolu šipky podle směru pohledu fotoaparátu.
- Přenositelnost: Výstupem musí být složka fungující offline (bez serveru) a snadno přenositelná (např. na USB disku).

Pedagogický rozměr spolupráce se studenty

Vývoj prezentovaného zásuvného modulu probíhal v rámci odborné praxe spoluautorky příspěvku, studentky závěrečného ročníku bakalářského studia oboru Aplikovaná informatika. Praxe proběhla na pracovišti Ústavu teoretické a aplikované mechaniky AV ČR v Centru Telč. Pedagogickým rámcem spolupráce se stala metoda Project-Based Learning (PBL), která umožnila aplikaci teoretických znalostí při řešení reálných výzkumných problémů s bezprostředním dopadem do praxe.

- V průběhu celosemestrální praxe (560 hodin) studentka prošla – jako jeden z úkolů – kompletním životním cyklem vývoje softwaru (SDLC):
- Inženýrství požadavků: Analýza potřeb koncových uživatelů (výzkumných pracovníků) a specifikace funkčních požadavků.
- Implementace: Vývoj v jazyce Python s využitím knihovny PyQGIS a návrh grafického uživatelského rozhraní (GUI).
- Validace a nasazení: Testování kompatibility napříč operačními systémy a tvorba technické dokumentace.

Hodnocení praxe prokázalo nejen vysokou úroveň studentčiných technických kompetencí, ale také rozvoj měkkých dovedností, zejména samostatnosti při řešení problémů a důslednosti.

Technické řešení pluginu

Zásuvný modul „SurveyWebMap Exporter“ je napsán v jazyce Python a integruje se do prostředí QGIS. Jeho jádro tvoří tři hlavní komponenty:

1. Robustní parsování metadat (Python a Pillow): Aby se předešlo rozdílům v interpretaci EXIF metadat napříč platformami, využívá plugin externí knihovnu Pillow (PIL) pro přímé čtení metadat obrázků. Klíčová je metoda `get_geo_info`, která extrahuje GPS souřadnice a (pokud jsou dostupné) i směr pořízení snímku (tag `GPSImgDirection`; případně `GPSImgDirectionRef` pro určení, zda je směr vztažen k pravému nebo magnetickému severu). Kód ošetřuje různé formáty zápisu hodnot (desetinná čísla vs. zlomky / typ `Rational`) a správně interpretuje referenční tagy pro souřadnice (N/S a E/W) pro výpočet znaménka. Data nejsou ukládána do dočasných souborů na disku, ale přímo do paměťové vrstvy QGIS (memory layer), což zrychluje zpracování.

```

from PIL import Image, ExifTags

def get_geo_info(self, filepath):
    """Extrakce GPS a azimutu pomocí knihovny Pillow"""
    try:
        img = Image.open(filepath)
        exif_raw = img._getexif()
        # ... (získání GPSInfo tagu) ...

        # Čtení azimutu (Tag 17 = GPSTagDirection)
        azimuth = 0
        if 17 in gps_info:
            val = gps_info[17]
            # Ošetření racionálních čísel (zlomků)
            try:
                azimuth = float(val)
            except:
                if isinstance(val, tuple) and len(val) == 2 and val[1] != 0:
                    azimuth = float(val[0]) / float(val[1])

        return lat, lon, azimuth
    except Exception:
        return None, None, None

```

Ukázka kódu 1: Extrakce EXIF metadat a výpočet azimutu (zdroj: Vlastní zpracování autorů).

2. Automatická vizualizace a rotace: Po importu bodů plugin automaticky aplikuje kartografickou symbologii. Bodová značka je nahrazena symbolem šipky. Pomocí datově definované vlastnosti je rotace symbolu programově navázána na atribut azimuth načtený z EXIF metadat. Uživatel tak okamžitě po nahrání fotek vidí v mapě nejen pozici, ale i směr, kterým byl snímek pořízen, což je pro dokumentaci památek zásadní (umožňuje odlišit snímky stejného místa pořízené z různých směrů).

```

# Vytvoření symbolu (červený trojúhelník)
symbol = QgsMarkerSymbol.createSimple({
    'name': 'triangle',
    'color': 'red',
    'size': '6',
    'outline_color': 'black'
})

# Získání vrstvy symbolu a nastavení rotace dle atributu
symbol_layer = symbol.symbolLayer(0)
symbol_layer.setDataDefinedProperty(
    QgsSymbolLayer.PropertyAngle,
    QgsProperty.fromField("azimuth")
)

# Aplikace rendereru na vrstvu
renderer = QgsSingleSymbolRenderer(symbol)
layer.setRenderer(renderer)

```

Ukázka kódu 2: Dynamická rotace symbolu v PyQGIS (zdroj: Vlastní zpracování autorů).

3. Generování webové mapy (Leaflet): Finální fází je export do HTML. Plugin provádí následující kroky:

1. Vytvoří exportní adresář s časovým razítkem.
2. Fyzicky zkopíruje fotografie do podsložky photos a v datech vygeneruje relativní cesty.

3. Načte šablonu template.html a pomocí zástupných symbolů do ní vloží data ve formátu GeoJSON.

Výsledná mapa je postavena na lehké knihovně Leaflet.js. Zajímavým detailem je vykreslování markerů pomocí dynamického SVG přímo v prohlížeči, což umožňuje zachovat rotaci šipek i ve webovém prostředí bez nutnosti generovat desítky statických ikon. Mapa je plně responzivní a je navržena pro offline použití – stačí otevřít soubor index.html.

```
# Vytvoření symbolu (červený trojúhelník)
symbol = QgsMarkerSymbol.createSimple({
    'name': 'triangle',
    'color': 'red',
    'size': '6',
    'outline_color': 'black'
})

# Získání vrstvy symbolu a nastavení rotace dle atributu
symbol_layer = symbol.symbolLayer(0)
symbol_layer.setDataDefinedProperty(
    QgsSymbolLayer.PropertyAngle,
    QgsProperty.fromField("azimuth")
)

# Aplikace rendereru na vrstvu
renderer = QgsSingleSymbolRenderer(symbol)
layer.setRenderer(renderer)
```

Ukázka kódu 3: Příprava dat a injektáž do HTML šablony (zdroj: Vlastní zpracování autorů).

Výsledky a diskuze

Nasazení pluginu výrazně zefektivnilo pracovní postup (workflow) při zpracování terénní dokumentace.

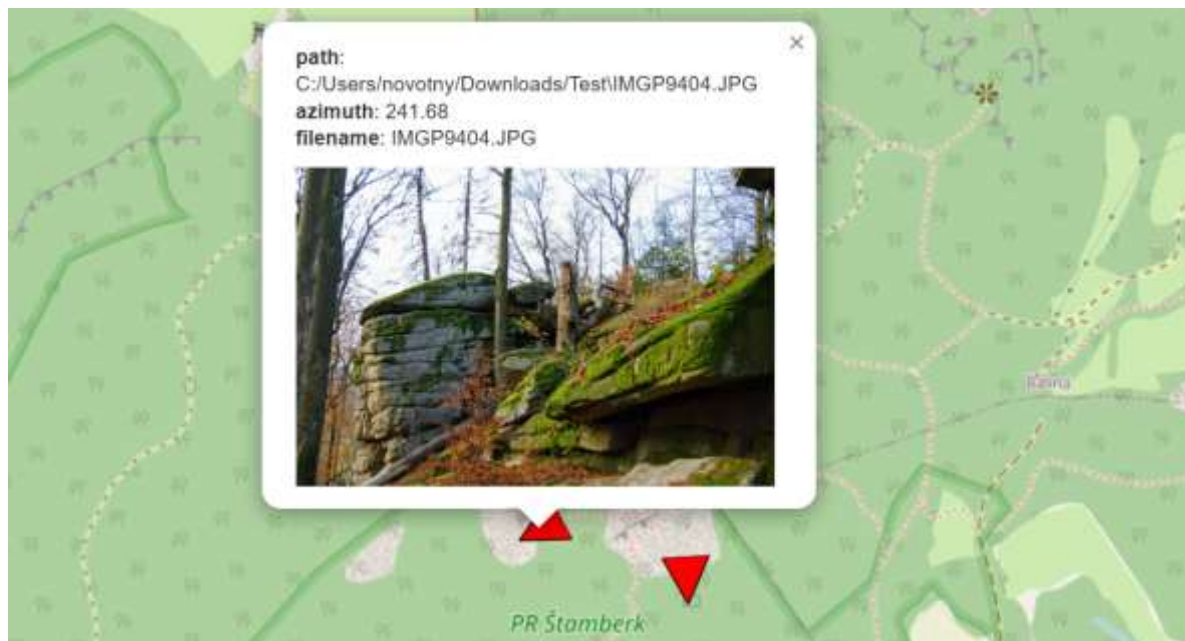
Tabulka 1: Srovnání efektivity pracovního toku

Krok procesu	Tradiční postup (manuální)	Postup s SurveyWebMap Exporter
Import dat	Externí nástroje, problémy s kódováním	Jedním kliknutím v QGIS, přímé čtení EXIF
Vizualizace	Manuální nastavování rotace a stylu	Automatická symbologie a rotace
Export	Složitější konfigurace qgis2web	Okamžitý export do HTML šablony
Distribuce	Závislost na serveru / absolutní cesty	ZIP archiv (HTML a složka s daty), funkční offline (relativní cesty)

Zdroj: Vlastní zpracování autorů.

Jako limitující faktor se v praxi ukazuje přesnost samotných GPS senzorů a digitálních kompasů v mobilních zařízeních, zejména v husté městské zástavbě, kde dochází k odrazům signálu

a magnetickému rušení. Uživatelé musí být s tímto limitem seznámeni – mapa slouží jako orientační schéma, nikoliv jako geodetické zaměření.



Obrázek 1: Ukázka výsledné mapové prezentace (zdroj: Vlastní zpracování autorů).

Závěr

Vývoj zásuvného modulu „SurveyWebMap Exporter“ potvrdil, že i v době existence komplexních GIS systémů jsou smysluplné jednoduché, specializované nástroje, které řeší konkrétní potřeby komunity – v tomto případě efektivní prezentaci památkové fotodokumentace. Automatizace procesu od stažení dat po vytvoření webové mapy šetří výzkumníkům čas a minimalizuje chybovost. V rámci dalšího vývoje je plánováno rozšíření funkcionality o shlukování bodů (clustering) pro lepší čitelnost rozsáhlejších datových sad a implementace pokročilých filtrů, které uživatelům umožní selekci snímků na základě časových či atributových metadat přímo ve webovém rozhraní.

Z pedagogického hlediska projekt demonstroval úspěšnost zapojení studentů do řešení reálných výzkumných úkolů. Již studenti bakalářských programů jsou schopni pod odborným vedením vytvářet softwarová řešení, která mají trvalou užitnou hodnotu. Tento model spolupráce mezi VŠPJ a ÚTAM AV ČR slouží jako inspirativní příklad pro propojování výuky s praxí.

Poděkování

Tento příspěvek vznikl za podpory Strategie AV21, konkrétně projektu *Moc předmětů: Materialita mezi minulostí a budoucností* (StrategieAV21/36), a programu Regionální spolupráce AV ČR, konkrétně projektu *Multidisciplinární výzkum zříceniny hradu Janštejn v geologických souvislostech Geoparku Vysočina* (R100712501).

Literatura

BONACCHI, C. et al. Crowd-sourced archaeological research: The MicroPasts project. *Archaeology International*, 2014, 17, 61–68. DOI: 10.5334/ai.1705.

KERSKI, J. *Using the Photos with Locations Tool in Education* [online]. Esri Community (Education Blog), 10. 3. 2020 [cit. 2025-11-28]. Dostupné z: <https://community.esri.com/t5/education-blog/using-the-photos-with-locations-tool-in-education/ba-p/883663>.

QGIS2WEB Development Team. qgis2web: Export QGIS projects to webmaps [software]. 2024 [cit. 2025-12-02]. Dostupné z: <https://plugins.qgis.org/plugins/qgis2web/>.

KYRIAKOU, M. S. et al. ImportPhotos [software]. QGIS Plugins Repository. 2019 [cit. 2025-12-02]. Dostupné z: <https://plugins.qgis.org/plugins/ImportPhotos/>.

LEAFLET TEAM. Leaflet: a JavaScript library for interactive maps [online]. 2025 [cit. 2025-12-03]. Dostupné z: <https://leafletjs.com/>.

Kontaktní údaje

Ing. Jakub Novotný, Ph.D.

Ústav teoretické a aplikované mechaniky AV ČR, v. v. i.

Prosecká 809/76, 190 00 Praha 9

e-mail: novotny@itam.cas.cz

Alžběta Frýdlová

Vysoká škola polytechnická Jihlava (studentka Aplikované informatiky)

Tolstého 16, 586 01 Jihlava

e-mail: frydlova@student.vspj.cz

JAK STUDENTI VYUŽÍVAJÍ GENERATIVNÍ UMĚLÉ INTELIGENCE PŘI ŘEŠENÍ PROGRAMOVACÍCH ÚLOH

HOW STUDENTS USE GENERATIVE ARTIFICIAL INTELLIGENCE TO SOLVE PROGRAMMING TASKS

Radim Remeš, Ladislav Beránek, Josef Milota

Abstrakt

Článek se zabývá využitím generativní umělé inteligence vysokoškolskými studenty ve výuce programování. Cílem je identifikovat a analyzovat různé kvalitativní přístupy, které studenti uplatňují při řešení zadaných programovacích úloh s asistencí nástrojů generativní umělé inteligence.

Klíčová slova: umělá inteligence, programování, výuka

Abstract

The article deals with the use of generative artificial intelligence by university students in programming lessons. The aim is to identify and analyze various qualitative approaches that students apply when solving assigned programming tasks with the assistance of generative artificial intelligence tools.

Keywords: Artificial Intelligence, programming, teaching

Úvod

V několika posledních letech prošla výuka programování značným didaktickým posunem. Nástup velkých jazykových modelů (LLM) transformoval výuku informatiky rychleji než jakákoliv dosud známá technologie. Tradiční model, kdy studenti hledají řešení zadaných úkolů na diskusních fórech a syntaxi upravují dle dokumentace programovacího jazyka, ustupuje a je nahrazován interakcí studentů s agenty generativní umělé inteligence (GenAI), které generují celé kódy v reálném čase. (Prather et al., 2023; Lau a Guo, 2023)

V tomto článku bychom se rádi podělili o naše zkušenosti a poznatky získané z výuky předmětu úvodního kurzu programování určeného pro studenty, budoucí učitele, na Jihočeské univerzitě.

Teoretický rámec

V tradičním způsobu vzdělávání (revidovaná Bloomova taxonomie podle Kraftwohl, 2002) se studenti vzdělávají postupně dle jednotlivých úrovní kognitivních dovedností, 1) od základní úrovně se základními dovednostmi zapamatovat si (základní pojmy, pamatování si syntaxe programovacího jazyka), přes 2) pochopit (programovací konstrukce, např. cykly nebo větvení), 3) aplikovat (používání naučených konceptů k řešení konkrétních problémů), 4)

analyzovat (rozdělení problému, nalezení chyby či nalezení vzorů v řešení), 5) hodnotit (dovednost kriticky posoudit více řešení a zdůvodnit volbu vhodnějšího z nich) až po 6) tvořit (navrhnout a realizovat komplexní řešení nebo celý projekt). Tento postup od nižších po vyšší kognitivní úrovně zajišťuje, že student nejprve buduje mentální modely problémů, následně je schopen porozumět jejich struktuře a až poté umí využít své schopnosti k efektivnímu vytváření komplexních řešení.

S nástupem generativní umělé inteligence se však tradiční učební postupy začínají narušovat. GenAI je schopna generovat funkční kód včetně kompletních řešení dříve, než student zvládne pochopit jeho funkci nebo osvojit si syntaxi jazyka. To znamená, že zadaný úkol, jehož řešení by mělo probíhat od nižších po vyšší kognitivní úrovně, může být zkratkovitě vyřešen externím nástrojem, aniž by student budoval základní znalostní schémata nebo porozuměl klíčovým konceptům. Tato možnost rychlého řešení přináší do procesu učení nové dynamiky, které mohou významně ovlivnit mentální reprezentaci problému, způsob řešení úloh i kvalitu osvojovaných dovedností.

Z hlediska kognitivní psychologie lze takové chování popsat teorií kognitivní zátěže (Cognitive Load Theory, CLT), která zdůrazňuje omezení pracovní paměti a potřebu optimalizovat rozložení zátěže při učení. CLT rozlišuje různé složky mentální zátěže (vnitřní, zevní a podstatná) a poukazuje na to, že navržené učební prostředí vyučujícími by mělo cílit na minimalizaci vnější zátěže, aby zbyla kapacita pro zpracování důležitých aspektů učiva. (Duran et al., 2022)

Pokud GenAI přebírá příliš mnoho elementárních kognitivních úloh (např. generování samotného kódu), studenti mohou přijít o příležitost postupně budovat vlastní mentální schémata, která jsou nezbytná pro hlubší porozumění a řešení komplexních úloh.

Kognitivní psychologie definuje *odlehčení* (cognitive offloading) jako využívání externích zdrojů ke snížení mentální zátěže, tedy přesun části kognitivní práce mimo jednotlivce, aby se uvolnila kognitivní kapacita pro jiné procesy. V kontextu programování to znamená, že delegování odpovědnosti za syntaktickou správnost, návrh struktury programu nebo řešení konkrétního úkolu na GenAI může teoreticky uvolnit kognitivní kapacitu studenta pro zaměření se na hlubší porozumění architektury programu, jeho logiku a další souvislosti. (Mehta et al., 2025)

Takové předčasné delegování může být kontraproduktivní: i když krátkodobě snižuje pracovní zátěž a umožní rychlé dosažení výsledku, omezuje hluboké zpracování učiva, které je nezbytné pro přenos znalostí do nových situací a zvládnutí vyšších kognitivních úrovní. Výzkumy ukazují, že časté spoléhání se na externí asistenci může snižovat aktivní využívání vyšších úrovní myšlení (analýzování, hodnocení, nebo tvoření) a místo toho může podporovat pasivní přijímání odpovědí nebo principů návrhu bez reflexe jejich kvality. (Moşoi et. al., 2025)

Tento posun může vést k povrchnímu osvojování znalostí, kde GenAI funguje jako nástroj nahrazující kognitivní práci studenta, místo aby sloužila jako podpůrný prostředek při budování hlubšího porozumění.

Z těchto důvodů integrace GenAI do výuky musí být pečlivě navržena tak, aby nástroj fungoval jako podpora učebních procesů, nikoliv jako jejich náhrada. Optimální využití GenAI v didaktice spočívá ve vytvoření prostředí, kde GenAI pomáhá studentovi budovat konceptuální rámce, které pak sám aktivně verifikuje a doplňuje, čímž podporuje proces seberegulace a hlubšího zapojení mentálních modelů. Takový přístup nejen že snižuje zevní kognitivní zátěž, ale zároveň maximalizuje podstatnou zátěž aktivním zapojením studenta do procesu učení.

Metodologie

Pro účely našeho zkoumání byla zvolena kombinovaná metodika založená na principech metodologické triangulace. Tento přístup umožňuje nahlížet na interakci studenta s generativní umělou inteligencí z několika perspektiv, od objektivního sledování technické aktivity přes subjektivní kognitivní procesy, až po finální evaluaci výkonu. Výzkumný design se skládal z následujících tří pilířů:

1. Analýza práce studenta v IDE,
2. Kvalitativní sonda metodou *Think-aloud*,
3. Kontrolované komparativní testování.

První fáze se zaměřila na kvantitativní sběr dat prostřednictvím sledování práce v integrovaném vývojovém prostředí (IDE) u cílové skupiny studentů úvodního kurzu programování. Cílem bylo monitorovat přirozenou práci respondentů, frekvenci kompilací, strukturu psaného kódu a dynamiku opravování syntaktických či sémantických chyb. Tato metoda poskytla objektivní záznam o tom, jakým způsobem studenti GenAI nástroje využívají při samotné technické realizaci úlohy.

Na technickou analýzu navázala *Think-aloud*, která sloužila k identifikaci kognitivních procesů, jež nejsou z pouhého sledování kódu patrné. Studenti byli požádáni, aby v reálném čase nahlas komentovali svůj proces uvažování a rozhodování při řešení úlohy za pomoci modelu ChatGPT (OpenAI, 2025). Tento postup umožnil výzkumníkům porozumět strategiím při tvorbě promptů, kritickému (či nekritickému) hodnocení generovaných odpovědí a způsobu, jakým studenti integrují doporučení GenAI do svého mentálního modelu problému.

Závěrečný pilíř metodiky tvořilo komparativní testování, jehož účelem byla objektivizace vlivu GenAI na výslednou kvalitu řešení. V přísně kontrolovaném prostředí proběhlo porovnání výsledků úloh řešených s podporou GenAI a bez ní. Sledovanými parametry byly zejména:

- algoritmická správnost a efektivita výsledného kódu,
- časová náročnost vypracování,
- schopnost studentů následně interpretovat a obhájit zvolené řešení.

Kombinace těchto metod umožnila eliminovat jednostrannost pohledu a poskytla komplexní data pro vyhodnocení dopadu používání velkých jazykových modelů (LLM) na pedagogický proces v oblasti výuky programování.

Výsledky a diskuze

Výzkumná data umožnila identifikovat tři typy užívání GenAI studenty. Tato typologie reflektuje nejen technickou zdatnost respondentů, ale především jejich kognitivní přístup k řešení problémů a míru jejich kritického myšlení.

Pasivní kopírovač

Tento typ studenta vnímá umělou inteligenci jako všeobsahující zdroj výsledného řešení. Proces interakce postrádá analytickou fázi; student do promptu vkládá zadání v nezměněné podobě, často nahrává nezměněný soubor se zadáním úkolu v podobě, jak jej získal z výukového systému. Dochází zde k situaci, kdy je proces dekompozice úlohy zcela přenechán na GenAI.

Typickým znakem tohoto typu přístupu je velmi krátký čas mezi zadáním úlohy a prvním odevzdáním. Při následné verbální obhajobě však student vykazuje neschopnost vysvětlit logiku použitých konstrukcí, volbu datových struktur nebo význam specifických instrukcí v generovaném kódu.

Primárním nebezpečím je vytvoření tzv. *křehkých znalostí*. Studenti získávají falešný pocit kompetence založený na úspěšném odevzdání kódu, avšak jeho porozumění selhává při následné modifikaci zadání, která by vyžadovala vlastní logický zásah bez asistence GenAI.

Ladič metodou pokus-omyl

Studenti náležející do této kategorie vykazují vyšší míru vlastní iniciativy, postrádají však hlubší porozumění sémantiky jazyka. Typicky napíší základní kostru kódu, ale jakmile narazí na chybu kompilace nebo logickou chybu, namísto analýzy chybového hlášení provedou prosté zkopírování chyby zpět do rozhraní GenAI s požadavkem na opravu.

Studenti nekontrolují, zda navržená oprava dává smysl v kontextu celého programu, a mechanicky aplikují doporučení GenAI, dokud program náhodou nezačne fungovat.

Dochází k rozvoji pouze povrchní kompetence. Studenti se sice naučí identifikovat, kde se chybová hlášení zobrazují, ale zcela rezignují na pochopení jejich důsledků. Neví, proč chyba vznikla a jak jí v budoucnu předcházet. Výsledný kód je často nejednotný, jednotlivé části jsou značně heterogenní.

Architekt a validátor

Tento přístup představuje ideální model integrace GenAI do výuky. Studenti nevyužívají model jako náhradu za vlastní myšlení, ale jako kognitivní lešení. Úlohu si nejprve vnitřně dekomponují a nástroj GenAI využívají ke generování dílčích, technicky náročných komponent nebo k vysvětlení teoretických konceptů, které následně sami kriticky hodnotí a integrují do svých řešení.

Prompty jsou vysoce specifické, kontextuální a často obsahují explicitní omezení (např. „implementuj daný algoritmus bez použití modulu X, abych pochopil, jak algoritmus funguje“). Studenti v rámci sledování Think-aloud kriticky posuzují části kódu navržené GenAI a upravují je podle svých potřeb.

Dochází k maximálnímu učebnímu efektu. GenAI zde neslouží jako generátor hotových řešení, ale jako vysoce efektivní osobní tutor, který urychluje rutinní kroky a umožňuje studentům soustředit se na vyšší kognitivní úroveň programování, jako jsou návrh struktury a optimalizace kódu.

Závěr

Výsledky našeho pozorování potvrzují rozdílný vliv GenAI na výuku: zatímco u motivovaných studentů dochází k akceleraci učební křivky, u ostatních hrozí úspěšné absolvování studia bez hlubšího porozumění problematice. Spoléhání se na asistenci GenAI v základech oboru vytváří deficit znalostního fundamentu, který se kriticky projevuje v momentě, kdy komplexita projektů přesáhne aktuální limity modelů a studenti již nejsou schopni samostatné analytické práce.

Vzhledem k integraci GenAI do praxe musí pedagogika transformovat své priority. Těžiště hodnocení se přesouvá z finálního kódu na validaci myšlenkových procesů. Programování se v současné době definuje spíše jako schopnost kritického čtení, ověřování a systémového designu než jako samotná tvorba zdrojového kódu.

Literatura

Anderson, L. W., Krathwohl, D. R., et al (Eds..) (2001) *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*. Allyn & Bacon. Boston, MA (Pearson Education Group).

Duran, R., Zavgorodniaia, A. & Sorva, J. (2022). [online], [2025-12-05]. Cognitive Load Theory in Computing Education Research: A Review. In *ACM Trans. Comput. Educ.* 22, 4, Article 40 (December 2022), 27 s. DOI: 10.1145/3483843. Dostupné z: <https://dl.acm.org/doi/10.1145/3483843>.

Lau, S. & Guo, P. (2023). [online], [2025-12-05]. From "Ban It Till We Understand It" to "Resistance is Futile": How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot. In *Proceedings of the 2023 ACM Conference on International Computing Education Research - Volume 1 (ICER '23)*, Vol. 1. Association for Computing Machinery, New York, NY, USA, 106–121. DOI 10.1145/3568813.3600138. Dostupné z: <https://dl.acm.org/doi/10.1145/3568813.3600138>

Mehta, N., Benjamin, J., Agrawal, A., Valanci, S., Masters, K., & MacNeill, H. (2025). [online], [2025-12-05]. Addressing educational overload with generative AI through dual coding and cognitive load theories. In *Medical teacher*, 1–3. Advance online publication. DOI 10.1080/0142159X.2025.2543548. Dostupné z: <https://pubmed.ncbi.nlm.nih.gov/40779508/>.

Moşoi, A. A., Maican, C. I., Cazan, A.-M. & Sumedrea, S. (2025). [online], [2025-12-05]. Do students need to think hard? The interplay of AI and cognitive abilities in solving problems. *Educ Inf Technol* 30, 24337–24364. DOI 10.1007/s10639-025-13738-8. Dostupné z: <https://link.springer.com/article/10.1007/s10639-025-13738-8>.

OpenAI. (2023). [online], [2025-12-05]. *ChatGPT [Large language model]*. Dostupné z: <https://chat.openai.com/chat>

Prather, J., Denny, P., Leinonen, J. & al. (2023). [online], [2025-12-05]. The Robots Are Here: Navigating the Generative AI Revolution in Computing Education. In *Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education* (Turku, Finland) (ITICSE-WGR '23). Association for Computing Machinery, New York, NY, USA, 108–159. DOI: 10.1145/3623762.3633499. Dostupné z: <https://doi.org/10.1145/3623762.3633499>

Kontakní údaje

Mgr. Radim Remeš, Ph.D.

Jihočeská univerzita v Českých Budějovicích

Fakulta zemědělská a technologická, Katedra datových věd a výpočetních systémů

Studentská 1668, 370 05 České Budějovice

e-mail: inrem@jcu.cz

Prof. Ing. Ladislav Beránek, CSc., MBA
Jihočeská univerzita v Českých Budějovicích
Fakulta zemědělská a technologická, Katedra datových věd a výpočetních systémů
Studentská 1668, 370 05 České Budějovice
e-mail: beranek@jcu.cz

RNDr. Josef Milota
Jihočeská univerzita v Českých Budějovicích
Rektorát, Centrum informačních technologií
Branišovská 1645/31a, 370 05 České Budějovice
e-mail: milota@jcu.cz

VÝUKA PHP FRAMEWORKU SYMFONY

TEACHING PHP FRAMEWORK SYMFONY

František Smrčka

Abstrakt

Článek ukazuje návrh obsahu předmětu webové technologie a praktické zkušenosti s jeho výukou. První lekce předmětu se zabývají JavaScript frameworky jQuery a jQuery UI a AJAX, který umožňuje načítat nebo odesílat data na server asynchronně. Pro tvorbu Back Endu je navržen PHP framework Symfony. Ten pracuje s MVC architekturou, je modulární, má vysokou bezpečnost a dobrou dokumentaci. Součástí výuky je i tvorba, testování a používání REST API. Před výukou PHP frameworku je vhodné studenty naučit s OOP v PHP dále práci s Composerem, což je správce závislostí pro PHP. Pro vytváření projektů PHP je použito profesionální vývojové prostředí PhpStorm. Předmět obsahuje 14 výukových lekcí, kde poslední 2 lekce se zaměřují na tvorbu závěrečné práce. Celý předmět je zakončen zkouškou. Článek zároveň hodnotí přínosy zvolené koncepce výuky z pohledu připravenosti studentů na praxi.

Klíčová slova: PHP framework Symfony, výuka, webové technologie, MVC

Abstract

The article shows the design of the content of the web technology course and practical experience with its teaching. The first lessons of the course deal with the JavaScript frameworks jQuery and jQuery UI and AJAX, which allows you to load or send data to the server asynchronously. The PHP Framework Symfony is designed for creating the Back End. It works with the MVC architecture, is modular, has high security and good documentation. The course also includes the creation, testing and use of REST API. Before teaching the PHP framework, it is advisable to teach students about OOP in PHP and how to work with Composer, which is a dependency manager for PHP. The professional development environment PhpStorm is used to create PHP projects. The course contains 14 lessons, where the last 2 lessons focus on creating a final thesis. The entire course ends with an exam.

Keywords: PHP framework Symfony, education, web technologies, MVC

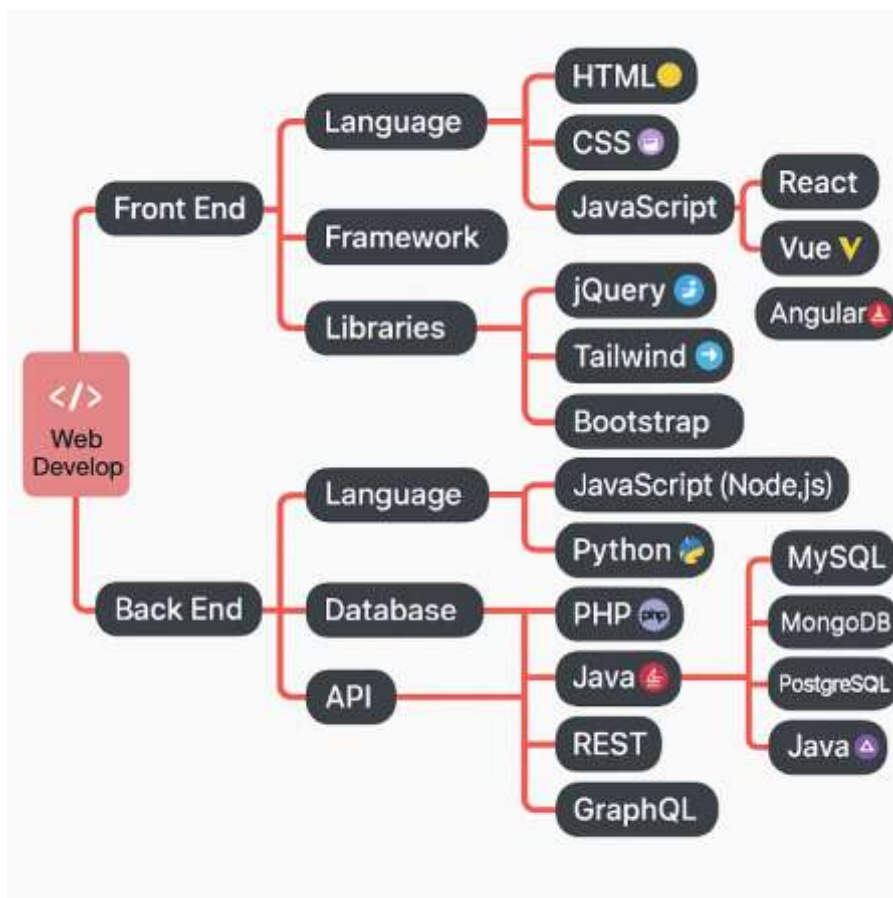
Úvod

Moderní webové technologie umožňují vytvářet dynamické, interaktivní a efektivní webové aplikace. Mezi často používané nástroje patří knihovna jQuery, asynchronní komunikace pomocí AJAXu a architektura REST API pro výměnu dat mezi klientem a serverem. Na straně serveru hrají důležitou roli PHP frameworky, například Symfony, které usnadňují vývoj přehledných a bezpečných aplikací. Tyto technologie jsou důležité ve výuce studentů, protože odpovídají aktuálním požadavkům praxe a běžně se používají při vývoji moderních webových

aplikací. Studenti si díky nim osvojují principy klient–server komunikace, práci s API a strukturovaný vývoj aplikací pomocí frameworků. Znalost jQuery, AJAXu, REST API a frameworku Symfony tak zvyšuje uplatnitelnost absolventů na trhu práce a připravuje je na reálné projekty. Výběr správných témat do jednotlivých výukových lekcí je důležitý pro efektivní a systematické osvojení znalostí. Díky správnému obsahu výuky mohou studenti získané znalosti použít v praxi a navazovat na ně v dalších předmětech či projektech. V dalších kapitolách budou vysvětleny důvody používání jednotlivých technologií, především však PHP frameworku Symfony.

Jaké technologie zařadit do výuky

Výukový předmět Webové technologie navazuje na předmět Tvorba internetových stránek. Zde se studenti naučili základy vytváření struktury webových stránek v nejnovější verzi značkovacího jazyka html5 (HyperText Markup Language). Vzhled a rozložení webových stránek nastavovali pomocí stylovacího jazyka CSS (Cascading Style Sheets). Pro tvorbu responzivních a vizuálně jednotných webových stránek je využíván front-endový framework Bootstrap. Dále byly v tomto předmětu probrány základy programovacího jazyka JavaScript, který umožňuje vytváření interaktivních a dynamických webových stránek. Důležitou částí předmětu Tvorba internetových stránek je výuka skriptovacího programovacího jazyka PHP (PHP: Hypertext Preprocessor), určeného především pro vývoj webových aplikací na straně serveru. Na předmět Tvorba internetových aplikací navazuje předmět Webové technologie. Při vytváření jeho obsahu se vycházelo z potřeb podle obrázku.



Obrázek 1: Technologie používané pro Front End a Back End. Zdroj: vlastní

Obsah předmětu Webové technologie

Základem předmětu jsou tyto kapitoly: jQuery, jQuery UI, Ajax, PHP OOP, REST API, MVC, PHP framework Symfony. jQuery je rychlá a jednoduchá JavaScriptová knihovna, která usnadňuje práci s HTML, událostmi, animacemi a AJAXem napříč všemi prohlížeči. jQuery UI je oficiální rozšiřující knihovna pro jQuery, která nabízí hotové widgety (např. kalendář, dialogy, posuvníky), animace a interakce připravené k okamžitému použití. (jQuery Team, 2025)

AJAX (Asynchronous JavaScript and XML) je technika, která umožňuje webovým stránkám načítat nebo odesílat data na pozadí, aniž by se musela znovu načíst celá stránka. (MDN contributors, 2025)

Pro zvládnutí PHP frameworku je zde kapitola věnována objektovému programování v PHP (PHP OOP).

REST API (Representational State Transfer Application Programming Interface) je způsob, jakým spolu komunikují aplikace přes internet – pomocí jednoduchých, jasně definovaných pravidel založených na architektuře REST. Umožňuje klientovi (např. webu nebo mobilní aplikaci) pracovat se „zdroji“ na serveru pomocí standardních HTTP metod. (IBM, 2025)

MVC je architektonický vzor, který rozděluje aplikaci na tři části: Model, View a Controller. Model se stará o data a logiku aplikace, View zobrazuje informace uživateli a Controller řídí komunikaci mezi nimi. Toto rozdělení pomáhá udržet kód přehledný, lépe testovatelný a snadno rozšiřitelný. (ITnetwork, 2025)

Další částí výuky, která byla řešena je výběr vhodného PHP frameworku.

Jaký framework vybrat

Samotný PHP lze použít pro Jednoduché weby s několika statickými stránkami s minimální logikou. Dále je vhodné ho použít pro jednorázové interní skripty, exporty dat, automatizaci nebo napojení na API, které nemají veřejné rozhraní ani složité prostředí. Malé mikro služby velmi jednoduchých backendových služeb se čisté PHP spustí rychleji, bez zbytečné zátěže. Pro větší projekty se používají frameworky.

Framework je v podstatě „stavebnice“ pro vývojáře. Místo toho, aby každý projekt psali od nuly, využívají předem připravenou strukturu, nástroje a osvědčené postupy. To zjednodušuje práci, šetří čas a snižuje riziko chyb. Dobrý framework umožní vývojáři soustředit se na logiku konkrétní aplikace, místo toho, aby musel řešit stále dokola základní věci, jako je přihlašování, směrování požadavků (routing) nebo ochrana proti útokům.

Mezi nejčastěji používané PHP frameworky v roce 2025 patří: 1. Laravel, 2. Symfony, 3. Laminas (Formerly Zend), 4. CodeIgniter, 5. CakePHP. (GeeksforGeeks, 2025)

Laravel

Laravel patří mezi nejpoužívanější PHP frameworky současnosti. Nabízí moderní přístup k vývoji, přehlednou syntaxi a množství nástrojů, které zrychlují práci vývojářů a tím i celý vývoj projektu. (Expert Dev, 2025)

Symfony

Symfony je robustní PHP framework, který si získal oblibu hlavně u rozsáhlých a dlouhodobých projektů. Používají ho banky, pojišťovny nebo globální e-shopy. Symfony nabízí maximální kontrolu, flexibilitu a důraz na čistou architekturu.

Na rozdíl od Laravelu je Symfony technicky náročnější. Vývoj v něm bývá na začátku pomalejší, ale dává uživateli do rukou stabilní základ, který zvládne i velmi složité projekty. Je to PHP framework, který se nehodí pro rychlý start, ale spíše pro projekty s dlouhou životností a vysokými nároky na výkon, bezpečnost a škálovatelnost. (Expert Dev, 2025)

Nette

Nette je český PHP framework, oblíbený pro svou srozumitelnost, čistý kód a důraz na bezpečnost. Hodí se pro firemní weby, interní systémy nebo menší až středně velké projekty, kde je potřeba mít vývoj plně pod kontrolou. (Expert Dev, 2025)

Na základě analýzy a potřeb firem byl zvolen pro výuku PHP framework Symfony.

Praktické zkušenosti s výukou PHP frameworku Symfony.

Obsah výuky PHP frameworku Symfony

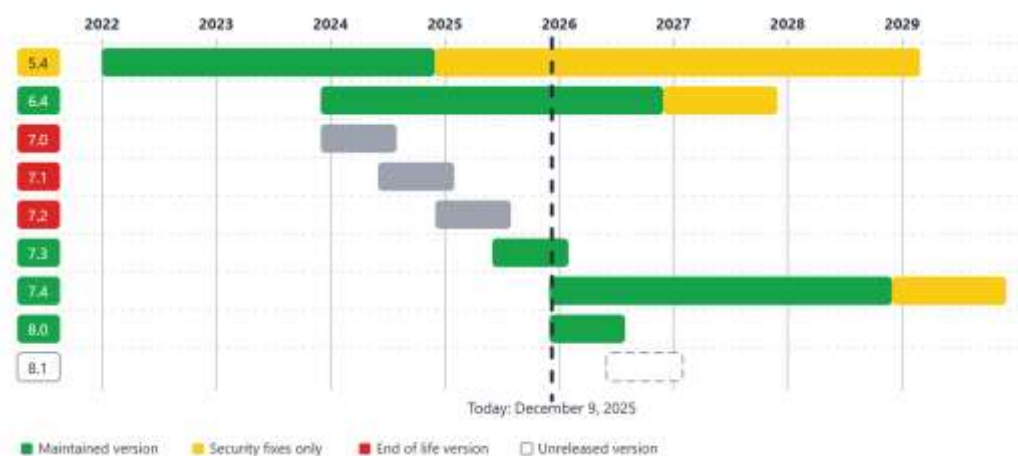
První lekce se soustředí na založení projektu se Symfony, práci s routováním a kontrolery a vykreslování šablon v Twigu. Následně přichází kontejner služeb a dependency injection, konfigurace prostředí (.env). Důležitá je práce s databází přes Doctrine ORM (entity, migrace) a tvorba formulářů s validací. Na malém projektu (například seznam studentů) si pak studenti procvičí autentizaci a autorizaci (Security).

Object–Relational Mapping (ORM) představuje softwarovou techniku, která zprostředkovává převod mezi objektovým modelem aplikace a relační databází, čímž umožňuje práci s databázovými daty prostřednictvím objektů programovacího jazyka.

Doctrine je open-source knihovna pro programovací jazyk PHP, která implementuje principy Object–Relational Mapping a poskytuje nástroje pro mapování objektových entit na relační databázové tabulky. (Symfony, 2025)

Problémy s verzemi a udržitelností

Symfony má velmi specifický a přísně řízený systém verzování, který ovlivňuje plánování vývoje, kompatibilitu i dlouhodobou údržbu projektů. Framework vydává nové verze podle pevného časového plánu a každá verze má přesně definovanou dobu podpory, což potvrzují oficiální stránky Symfony. Tyto skutečnosti jsou studentům vysvětlovány s důrazem na dlouhodobou udržitelnost projektů.



Obrázek 2: Symfony – kalendář verzí. Zdroj: <https://symfony.com/releases#symfony-releases-calendar>

Pro tvorbu projektů v PHP framework Symfony byl vybrán editor PhpStorm. PhpStorm je profesionální vývojové prostředí (IDE) pro PHP od české firmy JetBrains, navržené tak, aby výrazně zrychlilo programování. Patří mezi nejpoužívanější PHP IDE a nabízí chytré doplňování kódu, ladění, práci s databázemi i podporu moderních frameworků. (JetBrains. 2025)

Celý projekt se vytváří v XAMPP, nainstalovaném na operačním systému Windows. (Apache Friends, 2025).

Pro správu závislostí je nainstalovaný Composer, který zajišťuje automatické stahování, instalaci a aktualizaci knihoven potřebných pro daný projekt. Umožňuje definovat požadované balíčky v souboru composer.json a následně řeší jejich kompatibilitu i verzování, takže vývojáři pracují s konzistentní sadou knihoven. Composer také generuje autoloader, který usnadňuje načítání tříd a podporuje modulární a udržitelný vývoj aplikací. (Composer, 2025)

Pro splnění zápočtu je studenti vypracují 12 cvičení a vytvoří závěrečný projekt, který bude puštěný na školním serveru a jehož výsledky budou konzultovány s vyučujícím. Předmět končí zkouškou, která obsahuje otázky z přednášek.

Závěr

Předmět Webové technologie poskytuje studentům ucelený základ moderního webového vývoje. Kombinuje klientské technologie (jQuery, jQuery UI, AJAX) se serverovými principy (PHP OOP, REST API, MVC) a připravuje studenty na práci s profesionálními nástroji. Pro výuku byl zvolen PHP framework Symfony, který díky své robustnosti a dlouhodobé udržitelnosti odpovídá potřebám praxe.

Praktická část probíhá v profesionálním prostředí PhpStorm a na lokálním serveru XAMPP, což studentům umožňuje pracovat s nástroji běžně používanými ve firmách. Celková koncepce předmětu propojuje teorii s praxí a zvyšuje uplatnitelnost studentů při vývoji moderních webových aplikací.

V návaznosti na současný obsah předmětu lze výuku dále rozšířit o práci s nástrojem Docker, který umožňuje studentům pracovat s jednotným a snadno přenositelným vývojovým prostředím. Základní seznámení s CI/CD procesy by studentům pomohlo pochopit automatizaci testování a nasazování aplikací, což jsou běžné postupy v týmovém vývoji softwaru.

Literatura

Apache Friends. XAMPP – Apache + MariaDB + PHP + Perl [online], [2025-12-30]. Dostupné z: <https://www.apachefriends.org/>

Composer. Composer – Dependency Manager for PHP [online], [2025-12-30]. Dostupné z: <https://getcomposer.org/>

Expert Dev. Jak vybrat ten správný PHP framework pro váš web? [online], [2025-12-30]. Dostupné z: <https://www.expert-dev.cz/jak-vybrat-ten-spravny-php-framework-pro-vas-web/>

GeeksforGeeks. PHP Frameworks [online], [2025-12-30]. Dostupné z: <https://www.geeksforgeeks.org/blogs/php-frameworks/>

IBM What Is a REST API (RESTful API)? [online] [2025-12-30] Dostupné z: <https://www.ibm.com/think/topics/rest-apis>

ITnetwork. MVC architektura – návrhový vzor [online], [2025-12-30]. Dostupné z: <https://www.itnetwork.cz/navrh/mvc-architektura-navrhovy-vzor>

JetBrains. PhpStorm [online], [2025-12-30]. Dostupné z: <https://www.jetbrains.com/phpstorm/>

jQuery Team. jQuery: JavaScript Library. Version 3.x. [online], [2025-12-30]. Dostupné z: <https://jquery.com/>

MDN contributors. AJAX – Glossary. Mozilla Developer Network [online], [2025-12-30]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Glossary/AJAX>

Symfony. Symfony [online], [2025-12-30]. Dostupné z: <https://symfony.com/>

Kontaktní údaje

PaedDr. František Smrčka, Ph.D.
Vysoká škola polytechnická Jihlava
Tolstého 16, 586 01 Jihlava
e-mail: smrcka@vspj.cz

OD KANCELÁŘSKÝCH BALÍKŮ K INFORMATICKÉMU MYŠLENÍ: PRAKTICKÁ ZKUŠENOST S VÝUKOU NOVÉ INFORMATIKY NA ZÁKLADNÍ ŠKOLE

FROM OFFICE SOFTWARE TO COMPUTATIONAL THINKING: PRACTICAL EXPERIENCE WITH TEACHING THE NEW INFORMATICS AT PRIMARY SCHOOL

Pavel Ševčík, Jiří Vojtěšek

Abstrakt

Článek se věnuje proměně výuky informatiky od původního přístupu založeného především na používání kancelářských programů k moderní „nové informatice“, která rozvíjí informatické myšlení, tvořivost a praktické dovednosti žáků. Autor vychází ze své zkušenosti učitele informatiky na základní škole v roce 2024/2025. Zpočátku žáci vnímali předmět jako málo atraktivní, avšak po zavedení nových metod typu 3D modelování, robotiky, týmových projektů a herně orientované výuky se jejich zájem i aktivita výrazně zvýšily. Text nabízí porovnání obou přístupů a naznačuje, že atraktivní výuka je klíčem k rozvoji skutečných digitálních kompetencí.

Klíčová slova: nová informatika, projektová výuka, motivace žáků, informatické myšlení

Abstract

The paper explores the transformation of informatics education from a traditional approach focused on office software to the modern concept of “new informatics,” which develops student computational thinking, creativity, and practical skills. Based on the author’s experience as a primary school informatics teacher in 2024/2025, the lessons initially seemed unattractive to students. However, after introducing new methods – 3D modeling, robotics, team projects, and game-based learning – their motivation and engagement increased significantly. The paper compares both approaches and suggests that engaging, creative teaching is essential for developing students’ genuine digital competences.

Keywords: new informatics, project-based learning, student motivation, computational thinking

Úvod

V dnešní digitální době je na výuku informatiky kladen větší důraz než kdykoliv dříve. Technologie nás obklopují na každém kroku a požadavky na uživatele s každým dnem rostou. Tento trend se bude i nadále stupňovat. Současná generace dětí na základních i středních školách – a stejně tak generace po nich nastupující – jsou již dětmi digitální doby. Proto je nezbytné klást mnohem větší důraz na změnu metodiky výuky informatiky, než tomu bylo v letech předešlých.

Ať už se jedná o digitální gramotnost, bezpečnost na internetu, kde trávíme většinu času, nebo o rozvoj nových dovedností, jako je práce s umělou inteligencí, je klíčové, jak tento trend

uchopíme. Doba, kdy se vyučovala převážně jen práce s kancelářskými balíky, je nenávratně pryč. Je na čase učit informatiku moderním stylem, který reaguje na aktuální trendy, a nezůstávat několik let pozadu.

Tento příspěvek nevznikl pouze na základě teoretického studia, ale vychází z přímé pedagogické zkušenosti autora ve školním roce 2024/2025. Autor je začínajícím učitelem a zároveň studentem doktorského studia, který se ve své disertační práci věnuje právě analýze výuky nové informatiky.

Poznatky prezentované v textu byly získány na spádové základní škole v obci s přibližně 2000 obyvateli. Vzhledem ke spádovosti školy dojíždějí do zařízení i děti z okolních měst a vesnic, což vytváří široký kolektiv. Výuka a pozorování probíhaly na druhém stupni, konkrétně v 6., 7. a 8. ročníku, přičemž do ověřování nových metodik bylo zapojeno přibližně 150 žáků. Cílem bylo v praxi ověřit, jak přechod na obsah „Nové informatiky“ ovlivní motivaci a dovednosti dětí.

Současný stav na poli zavádění Nové informatiky

Problematika zavádění „nové informatiky“ rezonuje v odborné pedagogické komunitě velmi silně, zejména v souvislosti s povinným náběhem revidovaného Rámcového vzdělávacího programu (RVP ZV). Řada prací publikovaných v posledním roce (2024–2025) se věnuje především teoretickým aspektům implementace a mapování připravenosti škol.

Významným zdrojem dat je tematická zpráva České školní inspekce z roku 2024, která konstatuje, že ačkoliv ředitelé škol vnímají změny pozitivně, narážejí na personální limity a nedostatečnou sebedůvěru učitelů v technických dovednostech (Česká školní inspekce, 2024). Podobně i diplomové práce a akademické texty z poslední doby, se zaměřují spíše na zjišťování aktuálního stavu implementace a identifikaci bariér než na konkrétní metodické postupy ověřené v praxi (Minářová, 2024).

V rovině metodické podpory se objevují iniciativy zaměřené na dílčí nástroje. Wagner a platforma Pedagogické.info popularizují online kurzy a využití nástrojů jako je Canva, což představuje přístup k modernizaci výuky bez nutnosti hlubokých programovacích znalostí (Wagner, 2024). Na druhé straně spektra stojí praktické workshopy a setkání v rámci center Elixíru do škol či Institutu Nové informatiky, které v roce 2024 představovaly konkrétní pomůcky jako vláčky Intelino nebo drony (Institut nové informatiky, 2024). Existují rovněž případové studie zaměřené čistě na technické zvládnutí 3D tisku, jako například zkušenosti s tiskárnami MakerBot, které však často řeší technologii izolovaně od širšího kontextu motivace žáků (MakerBot, 2024).

Zatímco výše uvedené zdroje se soustředí buď na plošné mapování situace, nebo na návody k dílčím technologiím, tento článek přináší ucelenou reflexi procesu transformace. Nezaměřuje se pouze na to, jak technologii použít, ale analyzuje dopad radikální změny výuky – od stereotypních kancelářských balíků k robotice a 3D tisku – na vnitřní motivaci žáků a změnu jejich postoje k předmětu. Jde o praktické ověření toho, že atraktivní obsah dokáže překonat i počáteční odpor žáků k technicky náročnějším tématům.

Historický kontext a revize výuky informatiky

Pro pochopení současného stavu je nutné podívat se i na chronologický vývoj předmětu. V průběhu let docházelo k různým revizím, které se snažily přizpůsobit výuku novým trendům.

Období před rokem 2000

První systematické ukotvení přinesly Učební osnovy základní školy z roku 1991, které zavedly předmět Informatika pro 7. a 8. ročník (MŠMT, 1991). V polovině 90. let byla výuka realizována v rámci tří vzdělávacích programů schválených MŠMT: Základní škola (1996), Obecná škola (1997) a Národní škola (1997). V těchto programech byla informatika často řazena do pracovních činností nebo jako volitelný předmět, přičemž důraz byl kladen na základní seznámení s technikou (MŠMT, 1996; MŠMT, 1997).

Období 2000–2010

Zlomový bod přišel v roce 2000, kdy vláda ČR usnesením č. 351 schválila Koncepti státní informační politiky ve vzdělávání (SIPVZ) (Vláda ČR, 2000). Tato strategie cílila na plošné vybavení škol a zvýšení informační gramotnosti. Následně, se zavedením Rámcového vzdělávacího programu pro základní vzdělávání (RVP ZV) v roce 2005, vznikla samostatná vzdělávací oblast Informační a komunikační technologie (ICT), která definovala závazné výstupy zaměřené především na uživatelské dovednosti (VÚP Praha, 2005).

Období 2014–2020

S rychlým vývojem technologií se ukázalo, že pouhé uživatelské ovládání softwaru (kancelářských balíků) je nedostatečné. MŠMT proto v roce 2014 přijalo Strategii digitálního vzdělávání do roku 2020, která poprvé explicitně formulovala potřebu rozvíjet u žáků tzv. informatické myšlení a otevřít vzdělávání novým metodám (MŠMT, 2014).

Období 2021– současnost

Tento vývoj vyvrcholil aktuální revizí RVP ZV, která zavádí „Novou informatiku“. Dochází k navýšení hodinové dotace a zásadní proměně obsahu směrem k práci s daty, algoritmizaci a programování (MŠMT, 2021). Cílem již není jen obsluha počítače, ale pochopení digitálního světa. Od školního roku 2024/2025 je start této výuky na základních školách povinný.

Výchozí stav: Informatika jako "nuda"

Při nástupu na pozici učitele informatiky na 2. stupni ZŠ bylo jednou z prvních věcí zmapování dovedností dětí a jejich přístupu k předmětu.

Očekávání autora, začínajícího učitele informatiky, v tomto případě nebylo veliké, ovšem realita byla ještě horší.

Většina dětí (ze vzorku přibližně 150 žáků) uvedla, že je informatika nebaví. Hodiny si chodily pouze „odsedět“. Výuka v předešlých letech stereotypně opakovala práci s kancelářskými balíky a grafickými editory. Pouze několik jedinců projevilo zájem o programování a robotiku, se kterými se však setkali jen velmi okrajově. Bylo zřejmé, že je čas tento trend změnit.

Hlavní problémy současného stavu lze shrnout do tří bodů: nedostatečná aprobovanost učitelů, nízká motivace žáků a často i zastaralé technické vybavení škol.

Problematika aprobace vyučujících

Alarmujícím prvkem je nedostatek aprobevaných učitelů informatiky, což potvrzují i závěry České školní inspekce (Česká školní inspekce, 2024). Na většině škol je informatika stále vnímána jako okrajový předmět a k obsazení výuky se přistupuje podobně. I když škola disponuje moderním vybavením, které by mohlo žákům otevřít dveře do světa technologií, bez kvalifikovaného pedagoga je toto vybavení nevyužité. To nás vrací zpět do zaběhlých kolejí starého stylu výuky.

Nízká motivace žáků

Žáci často nevidí smysl v opakovaném formátování textu, což vede k pasivitě. Pokud výuka nereflktuje dynamický svět, který děti znají z domova (hry, sociální sítě, aplikace), ztrácí předmět atraktivitu.

Technické vybavení a rozdíly mezi školami

Zásadní bariérou je obrovská disparita ve vybavení jednotlivých škol. Zatímco některé instituce disponují moderními učebnami, jiné školy se potýkají se zastaralými počítačovými učebnami, které stěží stačí na výuku kancelářského software. Specifickou roli hraje i velikost školy a její rozpočet. Pro menší školy může být nákup sady robotů nebo 3D tiskáren velkou finanční zátěží, kterou si nemohou dovolit bez externích grantů či projektů. Naopak velké městské školy mohou mít lepší přístup k financím. Tato nerovnost vytváří riziko „digitální propasti“ mezi žáky z různých regionů a typů škol.

Změna a nástup Nové informatiky

Tato část popisuje nasazení nových postupů z konceptu Nové informatiky na vybrané základní škole, kde autor článku působil ve školním roce 2024/2025, které vedly ke změně vnímání informatiky mezi dětmi. I když se často říká, že změna bolí, pro posun vpřed je nezbytná.

Původní koncept výuky, který byl praktikován doposud na této škole se změnil od základů. Ačkoliv dovednosti jako psaní textu, tvorba prezentací či práce s tabulkami jsou stále potřebné, těžiště výuky se přesunulo. Na řadu přišly nové technologie: 3D modelování, základy algoritmizace, robotika, umělá inteligence a týmové projekty.

3D modelování a rozvoj představivosti

3D modelování je fenoménem posledních let. Ačkoliv výuka začínala v základních programech jako je TinkerCad. Žáci rychle získali přehled o fungování této sféry. Modelování rozvíjí představivost a kreativitu – děti dokáží své myšlenky přenést do virtuálního prostoru a následně je pomocí 3D tiskárny zhmotnit. Mnoho žáků bylo v úžasu z toho, jak jednoduše lze transformovat myšlenku do reality během několika hodin.

Tato činnost má navíc přesah do dalších předmětů. Elementární znalost modelování rozšiřuje prostorovou představivost v matematice nebo pomáhá v chemii při vizualizaci struktury atomů a vazeb. Zájem byl tak velký, že mnozí žáci konzultovali i pořízení vlastní tiskárny domů.

Základy algoritmizace a robotiky

Při prvním setkání s programováním bylo mnoho dětí vyděšeno. Obávaly se, že je to příliš obtížné, nebo měly zkušenost pouze s prostředím Scratch. Klíčem k úspěchu je správné podání látky.

Jakmile se dětem ukázalo, co může být výstupem – například že robot dokáže dovést předmět z druhé strany místnosti přímo k nim – jejich postoj se změnil. Velký úspěch, zejména u chlapců, zaznamenala disciplína RoboSumo – velmi oblíbená disciplína, kdy se žáci lehce naučí základní postupy algoritmizace a následně si program mohou vyzkoušet na fyzickém zařízení. Robot je sestaven z Lego dílků, které děti velmi dobře znají od malička a mohou tak rozvíjet svoji kreativitu různými typy konstrukcí. Následně svůj program nahrají do robota a vidí, jak jejich součástky ožívají a stává se z nich plnohodnotný stroj, který může zápasit proti ostatním robotům. Hmatatelná forma algoritmizace (práce s fyzickými roboty) slouží k popularizaci oboru lépe než pouhá vizuální výuka na monitoru.

Týmové projekty a herně orientovaná výuka

Dalším populárním prvkem se staly týmové projekty. Ty pomáhají dětem v komunikaci a umožňují jim pracovat ve skupině, kde vyniknou jejich individuální schopnosti. Pokud se každý člen týmu specializuje na to, co umí nejlépe, efektivita a kvalita výsledku roste.

Příkladem z praxe byla tvorba třídního časopisu. Děti využily své zkušenosti s grafickými editory (tvorba grafiky, koláže, úprava fotek) i práci s textem a formátováním. Tento projekt se napříč třídami těšil velké oblibě.

Zhodnocení: Je to správná cesta?

Po roce testování metodiky Nové informatiky došlo k velkému posunu v tom, co se děti naučily a jak předmět vnímají. Na konci roku byl s žáky veden stejný rozhovor jako na začátku roku. Výsledky potvrdily, že u dětí dramaticky vzrostl zájem o obor.

Pro mnoho dětí se informatika stala nejoblíbenějším předmětem, což je v porovnání s populárním tělocvikem markantní úspěch. Žáci začali předmět vnímat jako prostředek k získání dovedností, které jim pomohou v reálném životě. Postupem času se začali sami chodit ptát i o přestávkách, jak vylepšit své projekty, a experimentovali s technologiemi i doma nad rámec výuky.

Významným zjištěním je také nárůst popularity informatiky u dívek. Ty předmět dříve považovaly za nezajímavý, ale díky novému přístupu k němu našly cestu.

Závěr

Přechod od výuky zaměřené na kancelářské balíky k Nové informatice se ukázal jako nezbytný a vysoce efektivní krok. Zkušenosti ze školního roku 2024/2025 potvrzují, že pokud je látka podána moderní, praktickou a zábavnou formou, přirozená zvědavost žáků a jejich motivace k učení výrazně rostou.

Zavedení prvků jako 3D tisk, robotika a projektová výuka nejenže zvýšilo atraktivitu předmětu, ale také prokazatelně rozvinulo logické myšlení, kreativitu a schopnost týmové spolupráce. Aby však tento trend mohl pokračovat, je nezbytné řešit nejen technické vybavení škol, ale především odbornou připravenost a aprobaci pedagogů, kteří jsou nositeli této změny. Nová

informatika tak není jen změnou osnov, ale cestou k přípravě žáků na život v digitální budoucnosti.

Literatura

Česká školní inspekce (2024). [online], [2025-11-10]. *Tematická zpráva – Digitální kompetence a informatické myšlení žáků ZŠ a SŠ*. Praha: ČŠI. Dostupné z: https://www.csicr.cz/CSICR/media/Prilohy/2024_p%C5%99%C3%ADlohy/Dokumenty/TZ_digitalni_kompetence_informaticke_mysleni_final.pdf

Institut nové informatiky (2024). [online], [2025-11-28]. *Zprávy ze setkání DIGI centra Elixíru*. Dostupné z: <https://novainformatika.cz/elixir/zpravy-ze-setkani/>

MakerBot (2024). [online], [2025-11-28]. *Jak žáci základní školy pracují s 3D tiskárnou MakerBot SKETCH Classroom*. Dostupné z: <https://www.makerbot.cz/pripadove-studie/jak-zaci-zakladni-skoly-pracuji-s-3d-tiskarnou-makerbot-sketch-classroom/>

Minářová, N. (2024). [online], [2025-11-10]. *Implementace inovované informatiky do výuky na základních školách*. ČR: Theses. Diplomová práce. Dostupné z: https://theses.cz/id/zade3q/minarova_DP.pdf

MŠMT (1991). *Učební osnovy základní školy: Matematika 5.–8. ročník, Informatika 7.–8. ročník*. Praha: Fortuna. ISBN 80-85298-25-2.

MŠMT (1996). *Vzdělávací program Základní škola*. Čj. 16 847/96-2. Praha: Fortuna.

MŠMT (1997). *Vzdělávací program Obecná škola*. Čj. 12 035/97-20. Praha: Portál.

MŠMT (2014). [online], [2025-11-10]. *Strategie digitálního vzdělávání do roku 2020*. Praha: MŠMT. Dostupné z: <https://edu.gov.cz/dokumenty/strategie-msmt/strategie-digitalniho-vzdelavani-do-roku-2020/>

MŠMT (2021). [online], [2025-11-10]. *MŠMT spustilo web na podporu moderní výuky informatiky*. Dostupné z: <https://edu.gov.cz/msmt-spustilo-web-na-podporu-moderni-vyuky-informatiky/>

Vláda ČR (2000). *Usnesení vlády České republiky ze dne 10. dubna 2000 č. 351 ke Koncepci státní informační politiky ve vzdělávání*. Praha: Úřad vlády ČR.

VÚP Praha (2005). [online], [2025-11-10]. *Rámcový vzdělávací program pro základní vzdělávání*. Praha: VÚP. Dostupné z: https://www.npi.cz/images/RVP_ZV_2005.pdf

Wagner, J. (2024). [online], [2025-11-10]. *Nová informatika bez obav!* Pedagogicke.info. Dostupné z: <https://www.pedagogicke.info/2024/05/nova-informatika-bez-obav.html>

Kontaktní údaje

Ing. Pavel Ševčík

Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky

Nad Stráněmi 4511, 760 05 Zlín

e-mail: p_sevcik@utb.cz

ITERATIVNÍ GENEROVÁNÍ A LADĚNÍ MOODLE TESTŮ PRO VÝUKU PROGRAMOVÁNÍ S PODPOROU LLM

ITERATIVE GENERATION AND DEBUGGING OF MOODLE TESTS FOR PROGRAMMING EDUCATION SUPPORTED BY LLMS

Lenka Šroubová, Petr Kropík

Abstrakt

Příspěvek představuje model efektivní tvorby Moodle testů pro výuku programování, postavený na spolupráci mezi pedagogem a velkým jazykovým modelem (LLM). Díky schopnosti porozumět kontextu lze vytvářet testy, které reflektují studijní materiály. Příspěvek popisuje proces od generování různorodých typů otázek včetně validních distraktorů až po iterativní ladění exportních formátů jako GIFT a Moodle XML. Zároveň je zdůrazněna nová role pedagoga jako architekta zadání (prompt engineering). Klíčová je také odborná kontrola pedagogem, která zajistí správnost a didaktickou hodnotu. Výsledkem je udržitelný pracovní postup, který výrazně zefektivňuje přípravu a potvrzuje LLM jako výkonný nástroj pro podporu výuky, nikoli jako její náhradu.

Klíčová slova: E-learning, Moodle, Testování, LLM, Generování otázek, Formát GIFT/XML, CodeRunner, Prompt Engineering.

Abstract

The paper presents a model for the effective creation of Moodle tests for programming education based on collaboration between a teacher and a Large Language Model (LLM). Thanks to its ability to understand context, tests reflecting study materials can be created. The paper describes the process, from generating diverse question types that include valid distractors, to the iterative debugging of export formats such as GIFT and Moodle XML. Simultaneously, the new role of the teacher as an assignment architect (prompt engineering) is emphasized. Expert control by the educator is crucial to ensure correctness and didactic value. The result is a sustainable workflow that significantly streamlines preparation and confirms the LLM as a powerful tool for supporting teaching, not replacing it.

Keywords: E-learning, Moodle, Testing, LLM, Question Generation, GIFT/XML Format, CodeRunner, Prompt Engineering.

Úvod

Hodnocení znalostí v programovacích předmětech přináší specifické výzvy. Kromě teoretických znalostí je nutné ověřovat schopnost číst kód, odhalovat chyby a doplňovat syntaktické konstrukce. Tvorba takových otázek v prostředí LMS Moodle je tradičně časově náročná, zejména kvůli nutnosti generovat velké množství variant pro zajištění objektivit testování. S přechodem výuky z jazyka C++ na jazyk Python vyvstala potřeba vytvořit zcela

novou banku úloh. Tento článek popisuje využití velkých jazykových modelů (LLM) k automatizaci a zefektivnění tohoto procesu.

1 Metodika

Proces tvorby testů nebyl pojat jako plná automatizace, ale jako iterativní dialog. Role pedagoga se posunula od psaní otázek k navrhování struktury otázek (prompt engineering), pedagog zde vystupuje v roli architekta, LLM potom slouží jako stavitel projektu.

1.1 Generování obsahu a distraktorů

Jednou z nejsilnějších stránek LLM se ukázala být schopnost generovat validní distraktory (nesprávné odpovědi). Při zadání typu „Vytvoř otázku na cyklus while, kde student musí odhalit nekonečnou smyčku“, model nejen vygeneroval správný kód, ale navrhl i tři nesprávné varianty, které odpovídají typickým chybám začátečníků (chybějící inkrementace, špatná podmínka). Tím se výrazně zvyšuje didaktická hodnota testu, neboť distraktory nejsou náhodné, ale cílené na mylné představy studentů.

2 Technická realizace a formáty importu

Klíčovou částí naší zkušenosti bylo hledání efektivního způsobu, jak dostat obsah z chatu s AI do systému Moodle.

2.1 Limity formátu GIFT

Zpočátku byl pro svou jednoduchost zvolen formát GIFT. Pro textové otázky (Multiple Choice) fungoval bezchybně. Problémy nastaly při pokusu o import otázek obsahujících bloky kódu v jazyce Python. Znaky jako složené závorky {} nebo dvojtečky :, které jsou v Pythonu i v syntaxi GIFT klíčové, způsobovaly konflikty při parsování (Obrázek 1).

Stále je to problém; zdá se, že nefunguje importní filtr Moodle pro cloze. Mohl bys zkusit tyto úlohy přegenerovat do formátu Moodle xml – strukturu přikládám v přiloženém souboru.

Model

+ Thoughts

Expand to view model thoughts

Rozumím, situace je naprosto jasná. Děkuji za poskytnutí Moodle XML souboru. Analýza tohoto souboru je klíčová a odhalila několik důležitých věcí:

- Formát bodování:** Váš Moodle (nebo jeho importní filtr pro XML) skutečně používá **absolutní bodové hodnoty** (`{20:MULTICHOICE:...}`), nikoliv procentuální váhy, jak naznačovala předchozí chybová hláška z GIFT importu. To znamená, že můj úplně první návrh bodování byl správný pro tento formát.
- Struktura:** Vidím přesnou strukturu, kterou Moodle očekává – `<quiz>`, `<question type="cloze">`, `<name>`, `<questiontext format="html">` a hlavně použití `<![CDATA[ ... ]]>` pro bezpečné vložení HTML a kódu.
- Formátování kódu:** Pro zachování odsazení a formátování kódu (což je pro Python kritické) je nejlepší použít tagy `<pre><code>...</code></pre>`

Na základě této analýzy jsem kompletně přegeneroval všech 5 CLOZE otázek do Moodle XML formátu. Tento soubor by měl být 100% kompatibilní s vaším systémem.

Stačí zkopírovat celý obsah níže, uložit ho do souboru s příponou `.xml` (např. `python_cloze_otazky.xml`) a nainportovat ho do banky úloh v Moodle.

Moodle XML soubor pro import

```
<? Xml>

<?xml version="1.0" encoding="UTF-8"?>
<quiz>
  <!-- category definition -->
  <question type="category">
```

Obrázek 1: Ukázka dialogu s LLM – prompt pedagoga s požadavkem na opravu navrženého řešení – nefunkčního formátu pro import CLOZE úloh do Moodle a odpověď LLM.

Ještě větším problémem se ukázaly být otázky typu doplňovačka – CLOZE úloha. Moodle vyžaduje specifický zápis vah odpovědí (např. `~%25%odpověď`). LLM měl tendenci generovat absolutní body místo procentuálních vah, což vedlo k chybám při importu typu „Tento typ úlohy vyžaduje, aby součet vah byl 100%“.

2.2 Přejít na Moodle XML

Jako robustnější řešení se ukázal formát Moodle XML (Obrázek 1). Ten umožňuje uzavřít kód do bloků `<![CDATA[ ... ]>`, čímž je chráněn před interpretací XML parserem.

Zde jsme však narazili na specifické úskalí při spolupráci s AI. Model, trénovaný na mnoha programovacích jazycích, měl tendenci vkládat do XML tagů pro Python kód nadbytečně složené závorky (zvyk z C++/Java), což by mátló studenty. Bylo nutné model explicitně instruovat: „V kódu Pythonu nepoužívej blokové závorky, pouze odsazení“ a opakovaně zpřesňovat zadání, zadávat chybové hlášky a vzory importních formátů (Obrázek 2).



Obrázek 2: Ukázka stavu po 4. iteraci – zpřesňovacích promptech pedagoga, kdy LLM navržené řešení již bylo následně funkční.

3 Specifická úskalí CLOZE úloh a CodeRunner

Nejkomplexnější fází byla tvorba úloh typu CLOZE, kde studenti doplňují části kódu (např. klíčová slova `def`, `return`).

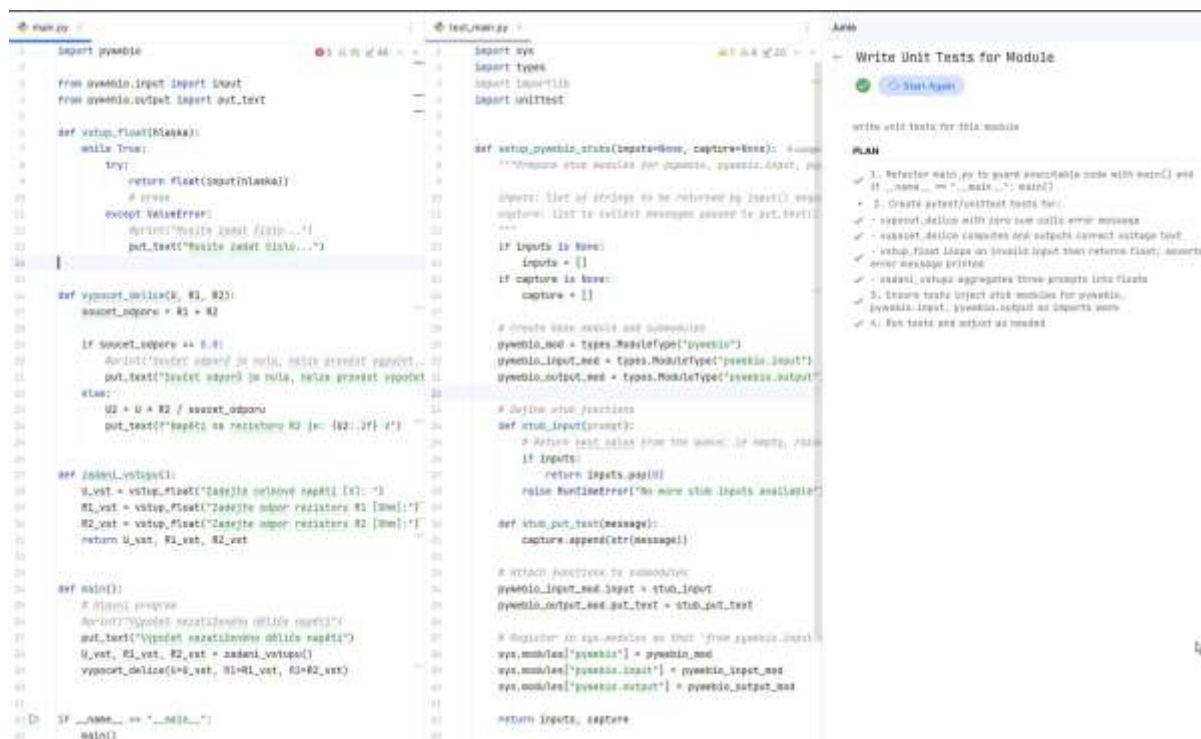
3.1 Halucinace parseru

Narazili jsme na situaci, kdy i syntakticky správné XML Moodle odmítl importovat. Příčinou byla kombinace Python syntaxe f-stringů (např. `f"Výsledek: {x}"`) uvnitř CLOZE pole. Moodle parser nedokázal rozlišit složené závorky f-stringu od závorek definujících doplňovačku.

Řešení bylo nalezeno v iterativním dialogu s LLM – nikoliv opravou syntaxe, ale zjednodušením kódu. Model byl požádán o přepsání příkladů tak, aby se problematickým konstrukcím vyhnul (např. nahrazení f-stringu klasickým `print`em s čárkami), čímž se zvýšila kompatibilita importu.

3.2 Úlohy CodeRunner

Pro pokročilejší testování (psaní celých funkcí) byl využit plugin CodeRunner. Zde LLM sloužil jako generátor testovacích případů (test cases). Ruční vymýšlení deseti různých vstupů a očekávaných výstupů např. pro funkci faktorial je zdouhavé; AI to zvládne vteřiny. Pedagogova role zde byla v kontrole hraničních případů (např. faktoriál 0 nebo záporného čísla). Pro tuto úlohu lze také s výhodou využít AI asistenty ve vývojových prostředích (Kilo Code, GitHub Copilot, Junie, apod.), zvláště jejich schopnost generovat unit testy, což je vhodné pro tvorbu test case pro CodeRunner (Obrázek 3).



Obrázek 3: Generování unit testů pomocí Junie.

4 Kritické zhodnocení a „Human-in-the-loop“

Je nutné otevřeně přiznat, že proces není plně automatický a bezrizikový.

Nutnost manuálních úprav: I přes pokročilé prompty bylo často nutné vygenerované XML soubory ručně otevřít a provést drobné korekce (např. smazání nadbytečné hlavičky, kterou model přidal).

Falešný pocit správnosti: Kód vygenerovaný AI vypadá na první pohled správně. Při importu do Moodle však může selhat na banalitě (např. špatně nastavená penalizace za chybnou odpověď). Proto je nezbytný proces: Generování -> Import -> Testovací průchod studentem (náhled) -> Korekce.

Odpovědnost pedagoga: AI není autorita. Pokud model navrhne testovou otázku, kde je jako správná odpověď označena sporná varianta, je na pedagogovi, aby to odhalil. Slepá důvěra v „strojovou neomylnost“ je největším rizikem.

Závěr

Využití LLM pro tvorbu Moodle testů představuje významnou úsporu času a umožňuje vytvářet variabilnější a didakticky bohatší testy. Není to však „magické tlačítko“. Vyžaduje od pedagoga novou kompetenci – schopnost přesně formulovat zadání pro AI a technickou zdatnost pro řešení integračních problémů mezi výstupem AI a LMS systémem. Pokud je tento proces zvládnut, stává se z LLM nepostradatelný asistent, který pedagogovi uvolňuje ruce pro to nejdůležitější – práci se studenty.

Poděkování

Tento příspěvek vznikl za podpory projektu SGS-2024-025 a Fakulty elektrotechnické Západočeské univerzity v Plzni.

Literatura

Savelka, S., Davis, J. C., Denny, P., Leinonen, J., Whalley, J. (2023). [online], [2025-11-28]. Generative AI for Programming Education: Benchmarking the Ability of Large Language Models to Generate Multiple Choice Questions. In Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2023). New York: ACM, 1–7. Dostupné z: <https://dl.acm.org/doi/10.1145/3587102.3588796>

Liu, T., Xia, X., Wang, Y., Zhang, X. (2023). [online], [2025-11-28]. Is Your Code Generated by ChatGPT Really Correct? Rigorous Evaluation of Large Language Models for Code Generation. arXiv preprint arXiv:2305.01210. Dostupné z: <https://arxiv.org/abs/2305.01210>

CodeRunner. CodeRunner: A Moodle quiz question type that runs student code [online], [2025-11-28]. Dostupné z: <https://coderunner.org.nz/>

Moodle. GIFT format [online], [2025-11-28]. Dostupné z: https://docs.moodle.org/403/en/GIFT_format

Moodle. Moodle XML format [online], [2025-11-28]. Dostupné z: https://docs.moodle.org/403/en/Moodle_XML_format

OpenAI. Prompt engineering [online], [2025-11-28]. Dostupné z: <https://platform.openai.com/docs/guides/prompt-engineering>

Kontaktní údaje

Ing. Lenka Šroubová, Ph.D., Ing. Petr Kropík, Ph.D.

Katedra elektrotechniky a počítačového modelování
Západočeská univerzita v Plzni, Fakulta elektrotechnická
Univerzitní 26, 301 00 Plzeň

e-mail: lsroubov@fel.zcu.cz, pkropik@fel.zcu.cz

MODERNÍ METODY VÝUKY OBJEKTOVÉHO MODELOVÁNÍ A ROZVOJ DIGITÁLNÍCH KOMPETENCÍ STUDENTŮ

MODERN METHODS OF TEACHING OBJECT MODELING AND DEVELOPMENT OF STUDENTS' DIGITAL COMPETENCES

David Velner, Petr Tučník

Abstrakt

Tento článek představuje inovativní přístup k výuce předmětu „Úvod do objektového modelování“. Ve výuce jsou používány moderní metody kombinující teoretické principy, praktické scénáře a případové studie inspirované praxí. Tento přístup podporuje zkvalitnění digitálních dovedností studentů a zvyšuje jejich uplatnitelnost na trhu práce. Akademici mají možnost rozvíjet své pedagogické zkušenosti v oblasti ICT. Článek představuje možnosti, jak propojení akademické sféry a praxe přináší inovace do výuky a vytváří prostor pro vzájemnou spolupráci. Tento inovativní didaktický přístup proměňuje tradiční roli univerzit, které se tak stávají platformami pro spolupráci a inovace, kde se setkávají studenti, pedagogové a odborníci z praxe.

Klíčová slova: objektové modelování, inovativní přístup k výuce, moderní metody výuky, digitální dovednosti studentů, pedagogické zkušenosti, spolupráce akademie a praxe, inovace ve vzdělávání

Abstract

This article presents an innovative approach to teaching the course Introduction to Object Modeling. Modern methods are applied in the teaching process, combining theoretical principles, practical scenarios, and case studies inspired by real-world practice. This approach supports the improvement of students' digital skills and enhances their employability in the labor market. Academic staff are given the opportunity to further develop their teaching experience in the field of ICT. The article demonstrates how linking academia and practice brings innovations into teaching and creates space for mutual collaboration. This innovative didactic approach transforms the traditional role of universities, turning them into platforms for cooperation and innovation where students, educators, and professionals meet.

Keywords: object modeling, innovative approach to teaching, modern teaching methods, students, digital skills, teaching experience, academia–practice collaboration, innovation in education

Úvod

Objektové modelování v rámci studia oboru zaměřených na informační a komunikační technologie (ICT) je významnou částí výuky (Blaha, Rumbaugh, 2004). Pro výuku se většinou používají teoretické koncepty a abstraktní příklady (Budgen, 2003). Dle Budgena (2003) je

k získání praktických dovedností zásadní, aby studenti porozuměli souvislostem objektového modelování s reálnou praxí. Zde mají informační systémy již reálné uživatele, konkrétní data z mnoha zdrojů a stanovená omezení. Právě proto je třeba poukázat na fakt, že výuka má stále prostor k modernizaci a zavedení inovativních metod. Jejich využitím mohou studenti rozšířit své odborné znalosti a zároveň to přispěje ke zvýšení jejich digitálních kompetencí.

Propojení studentů s praxí řeší také nejnovější verze koncepce Smart cities, která se v jednom ze svých pilířů věnuje podpoře spolupráce vysokých škol s veřejnou správou (Coughlan, 2020). Právě zde se nabízí velká možnost jejich praktického uplatnění. Subjekty veřejné správy disponují velkým množstvím datových zdrojů a informačních systémů. Tyto digitální aktiva poskytují nepřehledné varianty uplatnění inovativních přístupů k výuce studentů.

Tento článek si klade za cíl představit inovativní didaktický model, jehož účelem je propojení výuky objektového modelování s praktickými scénáři z praxe. Konkrétně se zaměří na prostředí veřejné správy dle doporučení koncepce Smart cities. Tímto způsobem dochází k podpoře aktivního učení ze strany studentů a zároveň se tím do značné míry obohacují o pedagogické zkušenosti samotní vyučující. Studenti tak mohou získat vyšší šanci k úspěšnému uplatnění na trhu práce.

1 Teoretická východiska

Objektové modelování ve vysokoškolské výuce

Objektové modelování učí studenty více hovořit jazykem informačních systémů, umožní jim nahlédnout pod jejich pomyslnou pokličku a lépe se zorientovat ve způsobech jejich chování, vztahů a vazeb. Při výuce se využívají známé koncepty, jako Use-Case diagramy, analytické a návrhové modely tříd (Blaha, Rumbaugh, 2004). To vše probíhá v logické posloupnosti a studentům je umožněno pochopit požadavky zadavatele, přetransformovat je do modelu případů užití. Na tomto základě vypracují analytický a návrhový model tříd. Studenti v rámci výuky mohou tento postup vnímat jako příliš abstraktní (Budgen, 2003). Jednotlivé diagramy na ně mohou působit izolovaně, jelikož jim často chybí kontext reálného světa. Tím se výuka objektového modelování může stát teoretickou disciplínou (Kayuni, 2025). Propojení s praxí je tak výzvou a inovativním krokem k zajištění moderní výuky, jelikož umožní studentům nastínit způsob, jak se mohou jednotlivé objekty v rámci informačních systémů chovat a jaké problémy mohou nastat.

Moderní metody výuky ICT

Pro komplexní rozvoj digitálních dovedností studentů daný dynamickým trendem technologického vývoje je naprosto nezbytné, aby studenti byli zapojeni do praktického řešení problémů a fungování informačních systémů (Sindre et al., 2018). Podstatnou rolí tohoto rozvoje je práce se skutečnými daty (Sindre et al., 2018), případovými studiemi, projektově orientovanou výukou. To vše přináší vzhled do praxe, která odpovídá skutečným situacím.

Studentům je umožněno analyzovat procesy, datové zdroje, cíle a reálné modely (Huang et al., 2022). Rozdíl mezi praxí a akademickým prostředím je zřejmý z hlediska metodického zázemí ze strany univerzit a vysokých škol, které disponují výzkumným potenciálem a hlubokými teoretickými znalostmi v oblasti architektury systémů. Praxe umožňuje studentům dovednosti získané z tohoto zázemí přenést do reálných projektů s konkrétními uživatelskými požadavky a autentickými daty. Tato zkušenost jim dává ambici prohloubit své znalosti a digitální

dovednosti v co největší míře tak, aby byli schopni se následně uplatnit na trhu práce s co největším potenciálem. Propojení akademické sféry a praxe je nezbytným krokem k přeměně klasické výuky směrem k modernímu vzdělávání, zejména pro oblast ICT, kde dochází k exponenciálnímu vývoji.

2 Inovativní didaktický přístup

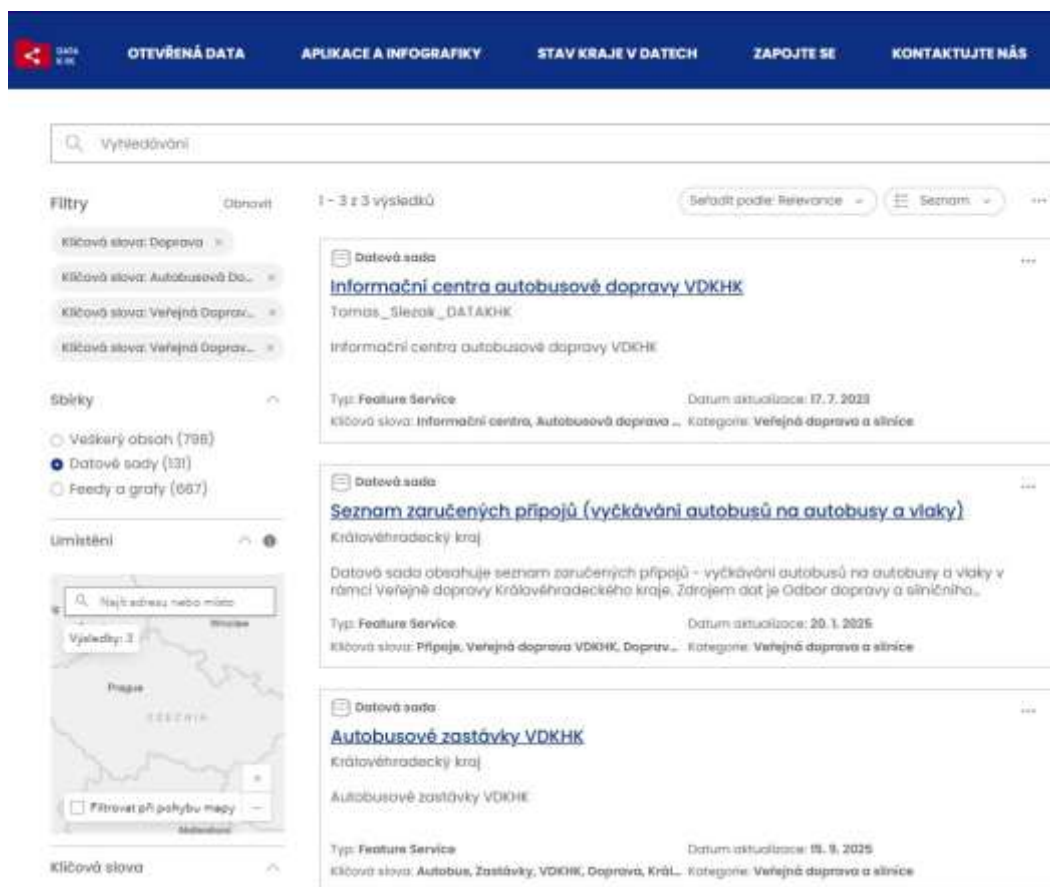
Propojení teorie s praxí

Pokud dojde k propojení teorie, praktických scénářů a případových studií na základě reálných dat a informačních systémů, vznikne prostředí, které je inspirující jak pro studenty, tak i pro vysokoškolské učitele. Umožní jim pochopit konkrétní význam v praxi.

Modelové příklady

Modelovým příkladem pro toto inovativní řešení je spolupráce akademické sféry se subjekty veřejné správy. Oblastí, která se nabízí k využití je téma otevřených dat. Toto téma poskytuje bezpečné zázemí datových zdrojů, které neobsahují citlivé údaje a jsou poskytovány zdarma bez licenčních zatížení (Coughlan, 2020). Studenti tak mohou pracovat s reálnými strukturami dat, které mohou nést znaky skutečných informačních systémů, jako jsou nepřesnosti, inkonzistence, chybné atributy, chybějící hodnoty a různé formáty dat. Rozdílnost mezi daty, které jsou určeny primárně pro výuku a těmito skutečnými je markantní. Studenti se tak dostávají do situací, kdy je třeba datové zdroje očistit od prvotních odchylek od ideálního stavu a až následně mohou využívat metodické postupy stanovené pro výuku objektového modelování (Coughlan, 2020).

Konkrétním příkladem využití otevřených dat ve výuce může být práce s lokálními katalogy otevřených dat veřejné správy. Tyto katalogy poskytují širokou škálu datových sad, například informace o dopravních zastávkách, autobusových a vlakových linkách nebo územním členění krajů, okresů a obcí. Studenti si mohou vybrat jako podklad pro modelování systému například datovou sadu obsahující informace o autobusových zastávkách, viz obrázek 1.



Obrázek 1: Ukázka datové sady otevřených dat týkající se autobusových a vlakových linek.

Zdroj: Data KHK (2025)

Na základě průzkumu datové sady poté mohou využít získané poznatky pro první úvahu o návrhu informačního systému. Dalším krokem by pak byla identifikace potenciálních uživatelů, formulace případů užití (např. vyhledání zastávky podle názvu, zobrazení linek obsluhujících zastávku). Následně poté vytvořit analytický model tříd, kde budou zachyceny základní entity, atributy a vztahy. V návrhové fázi poté přizpůsobit model technickým aspektům implementace, jako je přesná definice vztahů a operací mezi třídami.

Uvedený příklad demonstruje možnost poskytnutí otevřené datové domény studentům. Tato inovace, která koresponduje se strategií Smart cities, umožňuje rozvoj analytického a zároveň kritického myšlení studentů. Výuka tak dostává dynamičtější tempo.

3 Výhody pro rozvoj studentů

Autentické otevřené datové zdroje, zejména z oblasti veřejné správy, výrazně podporují rozvoj studentů a výuku objektového modelování. Jejich využití představuje výrazný potenciál pro to, aby studenti nebyli pouze pasivními příjemci teoretických a předpřipravených scénářů, ale aktivně pracovali s doménami v reálném prostředí. Díky tomu, že studenti mohou pracovat s autentickými daty, si rozvíjí své schopnosti převodu skutečných jevů do Use-Case diagramů, ve kterých identifikují byznysové aktéry, jejich cíle a interakce se systémem. Skutečná data vytvářejí situace, které mohou být velice komplexní, složité a nejasné (Blaha, Rumbaugh, 2004). Tyto skutečnosti nutí studenty k hlubšímu zamyšlení nad omezeními systémů, existencí

výjimek a dalších situací z reálného světa. Tímto způsobem dochází k rozvoji jejich datové a digitální gramotnosti, která je základním pilířem pro práci v ICT profesích (Huang et al., 2022).

Schopnosti studentů kvalifikovaně pracovat s daty, vytvářet objektové modely ve vysoké úrovni kvality a odůvodnit jejich strukturu, je důležitá dovednost dnešní doby, kterou zaměstnavatelé velmi často vyhledávají. Výukový proces vedení tímto inovativním způsobem si klade za cíl přiblížit studentům styl práce profesionálních týmů v oblasti ICT (Bonetti et al., 2025).

4 Rozvoj pedagogických zkušeností a dopad na akademickou sféru

Zapojení otevřených datových zdrojů do výuky objektového modelování, kterými disponují zejména subjekty veřejné správy formou otevřených dat, přináší benefity nejen studentům, ale významným způsobem ovlivňují také akademické prostředí. Naskytuje se tak možnost práce s reálnými daty, která poskytuje vyučujícím získat zpětnou vazbu o tom, jakým způsobem dokážou studenti interpretovat struktury skutečných informačních systémů. A nejen to (Bonetti et al., 2025). Vyučující dostávají reálnou představu, jakým způsobem studenti porozuměli datovým vztahům a zvládli přechod od analýzy až ke konečnému návrhu. Tyto zkušenosti poskytují pedagogům se lépe zaměřit na metodické posílení ve výkladu během výuky.

Podklady pro výuku, založené na otevřených datech, poskytují nezměrné množství inovativních přístupů pro pedagogy. Nemusí tak přemýšlet o volbě vhodných datasetů, postupů a zadání. Pro vyučující se otevírá příležitost pro práci s moderními nástroji a technologiemi, čímž se posilují jejich odborné a pedagogické kompetence.

Je důležité podotknout, že otevřená data jsou převážně datovými zdroji od veřejných institucí, a tím se stávají mostem pro vzájemnou spolupráci mezi akademickou sférou a veřejnou správou. Nabízí se tak k řešení nová výzkumná témata, která mohou nalézt praktické uplatnění v oblasti široké veřejnosti. Do budoucna to poskytuje další rozvoj nejen v automatizaci modelování procesních analýz nebo optimalizaci informačních systémů, ale i užití umělé inteligence. Pro pedagogy to představuje přístup k aktuálním trendům a reálným problémům v oblasti veřejné správy, které mohou být tématy pro vědeckou činnost.

Závěr

Uvedený inovativní přístup ve výuce objektového modelování poskytuje možnost výrazného zvýšení kvality výukového procesu za využití otevřených dat veřejné správy. Pro objektové modelování to přináší zvýšenou motivaci studentů k hlubšímu porozumění vzájemných vazeb v reálných informačních systémech. Tento způsob moderní výuky posiluje digitální kompetence studentů, učí je kriticky hodnotit skutečné datové zdroje a pracovat s nástroji používanými v profesionálním prostředí. Velkou předností je zejména otevřenost výuky směrem k praxi a možnost rozvíjet spolupráci mezi institucemi.

Poděkování

Autor tímto děkuje vedoucímu své disertační práce doc. RNDr. Petru Tučníkovi, Ph.D. za odborné vedení, cenné konzultace a dlouhodobou podporu při vzniku tohoto příspěvku. Tento příspěvek byl podpořen specifickým výzkumným projektem Fakulty informatiky a managementu UHK 2108 – Řešení aktuálních výzev v oblasti inteligentních měst a kybernetické bezpečnosti.

Literatura

- BLAHA, Michael a James RUMBAUGH. *Object-Oriented Modeling and Design with UML*. 2. Pearson, 2004. ISBN 978-0130159205.
- BONETTI, Tiago P., Silva WILLIAMSON a Thelma E. COLANZI. Example-Based Learning in Software Engineering Education: A Systematic Mapping Study. *Cornell University* [online]. 2025, **2025** [cit. 2025-11-26]. Dostupné z: doi:<https://doi.org/10.48550/arXiv.2503.18080>
- BUDGEN, David. *Software Design*. 2. Addison-Wesley, 2003. ISBN 978-0201722192.
- COUGHLAN, Tim. *The use of open data as a material for learning* [online]. 2020, **2020**, 383–411 [cit. 2025-11-26]. Dostupné z: doi:<https://doi.org/10.1007/s11423-019-09706-y>
- KAYUNI, Moses. *TEACHING OBJECT-ORIENTED DESIGN THROUGH IN TED DESIGN THROUGH INTERACTIVE UML: A DUAL-APPROACH FRAMEWORK FOR CODE-BASED GENERATION AND DIREC TION AND DIRECT DIAGRAM MANIPULATION*. Kentucky, 2025. Diplomová práce. Western Kentucky University.
- SINDRE, Guttorm, Michail GIANNAKOS, Birgit R. KROGSTIE, Robin MUNKVOLD a Trond AALBERG. Project-Based Learning in IT Education: Definitions and Qualities. *SCUP* [online]. 2018, 147–163 [cit. 2025-11-26]. Dostupné z: doi:<https://doi.org/10.18261/ISSN1893-8981-2018-02-06>
- HUANG, Wen, Jeremi S. LONDON a Logan A. PERRY. Project-Based Learning Promotes Students' Perceived Relevance in an Engineering Statistics Course: A Comparison of Learning in Synchronous and Online Learning Environments. *Taylor and Francis Online* [online]. 2022, 179-187 [cit. 2025-11-26]. Dostupné z: <https://www.tandfonline.com/doi/full/10.1080/26939169.2022.2128119>

Kontaktní údaje

Ing. David Velner LL.M.

Fakulta informatiky a managementu Univerzity Hradec Králové
Hradecká 1249/6, 500 03 Hradec Králové 3

david.velner@uhk.cz

OTEVŘENÁ DATA KE ZVYŠOVÁNÍ DIGITÁLNÍCH DOVEDNOSTÍ STUDENTŮ

OPEN DATA FOR ENHANCING STUDENTS' DIGITAL SKILLS

David Velner, Petr Tučník

Abstrakt

Otevřená data disponují významným potenciálem pro rozvoj digitálních dovedností studentů. Tento článek přibližuje, jak zapojení otevřených dat do výuky podporuje datovou gramotnost a schopnost práce s reálnými daty. Na příkladech z praxe zejména prostřednictvím studentských hackathonů pracujících s otevřenými daty a umělou inteligencí, přibližuje inovativní způsob přípravy studentů po odborné stránce do praxe. Poukazuje na to, že propojení reálných dat, AI nástrojů a týmové spolupráce vede k řešení reálných společenských výzev. Tím může být vývoj aplikací na základě otevřených dat, které slouží široké veřejnosti prostřednictvím datových portálů. Začlenění otevřených dat do výuky je jednoznačně inovativním prostředkem pro rozvoj studentských talentů.

Klíčová slova: otevřená data, hackathon, datový portál, umělá inteligence, digitální dovednosti

Abstract

Open data have significant potential for developing students' digital skills. This article illustrates how integrating open data into teaching supports data literacy and the ability to work with real data. Using practical examples, particularly through student hackathons working with open data and artificial intelligence, it presents an innovative way of preparing students professionally for practice. It points out that combining real data, AI tools, and teamwork leads to solving real societal challenges. This can include developing applications based on open data that serve the wider public through data portals. Incorporating open data into teaching is clearly an innovative means of fostering students' talents.

Keywords: open data, hackathon, data portal, artificial intelligence, digital skills

Úvod

Proces digitalizace, který je v současnosti běžnou praxí, probíhá i v oblasti vzdělávání, a to velice intenzivní cestou (Digitální a informační agentura, 2025). Mění se nároky na získávání kompetencí a digitálních dovedností nejen u studentů středních škol, ale i absolventů všech typů škol. Schopnost správně využít data, převést je do správných informací formou digitálních nástrojů a umět je správně interpretovat, to je dnes již velice žádanou dovedností v různých oblastech a odvětvích, ať jsou to technické obory, ekonomie, zdravotnictví nebo školství, veřejná správa. A právě pro tyto oblasti získávají stále větší význam a důležitost otevřená data. Je to z toho důvodu, že se jedná o volně dostupný datový zdroj, s kterým se dá významně pracovat s vysokou mírou inovativnosti, k zajištění chytrých řešení pro širokou veřejnost.

Téma otevřených dat se šíří významnou pozitivní energií napříč společností, má vysoký potenciál k rozvoji a je spouštěčem inovací. Otevřená data tak nemusí sloužit pouze vybraným oblastem, ale mohou podstatně přispět také ke zvyšování digitálních dovedností studentů. Dávají jedinečnou příležitost k propojení teoretické výuky s reálnými daty, která jsou poskytnuta veřejnými institucemi v rámci veřejné správy nebo vědeckých a výzkumných organizací. Prostřednictvím otevřených dat získávají studenti zkušenosti jednak v oblasti kvality zpracování dat, ve využití různých nástrojů i s umělou inteligencí (AI). Otevřená data mohou být zdrojem pro týmovou spolupráci, dokážou propojit studentské projekty přímo na potřeby veřejnosti k jejich následnému užití.

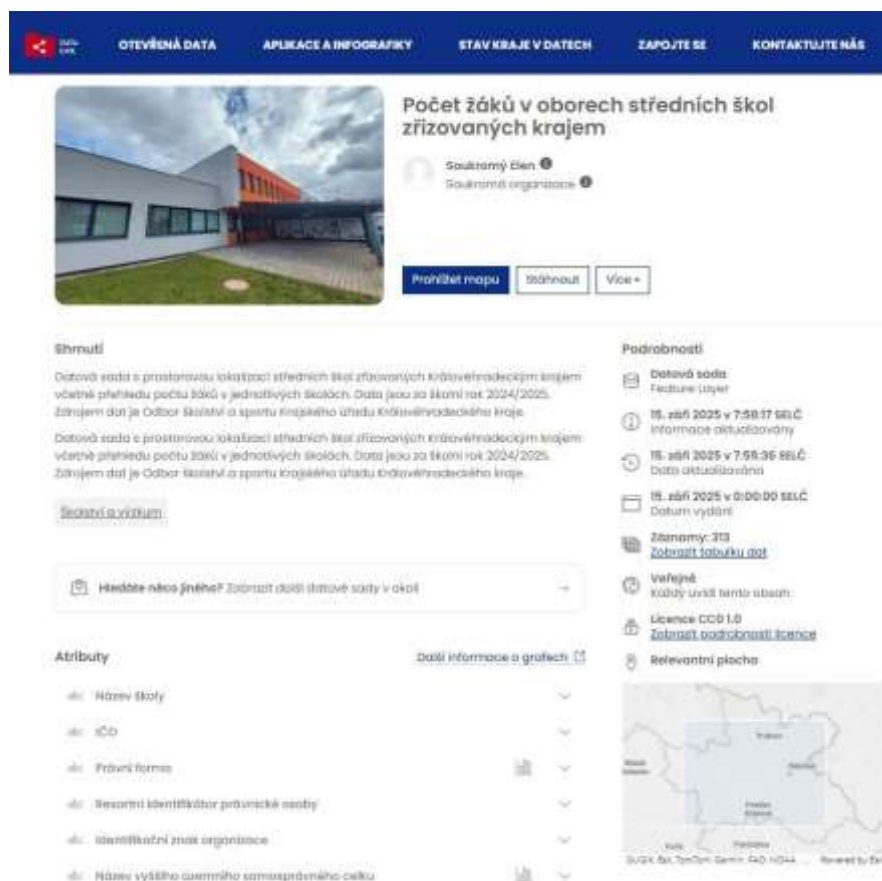
Významnou aktivitou v oblasti otevřených dat a inovačních procesů během výuky se dnes stávají studentské soutěže tzv. hackathony nad otevřenými daty (Velnerová, Poulová, 2025). Ty dokážou připravit studenty po odborné stránce do praxe a jsou dalším inovačním trendem pro rozvoj studentských talentů a zvyšování digitálních dovedností.

1 Otevřená data

Otevřená data jsou jedním ze základních datových zdrojů pro vývoj aplikací, které mohou sloužit široké veřejnosti. Tyto volně dostupná data jsou příčinou vzniku průkopnických služeb, podporují spolupráci mezi dalšími subjekty (Úřad pro publikace Evropské unie, 2025). Jelikož je prostřednictvím otevřených dat umožněn přístup ke kvalitním výukovým materiálům, inovují se také vzdělávací metody v oblasti pedagogické výuky (Digitální a informační agentura, 2025).

Téma otevřených dat je součástí zákona č. 106/1999 Sb. o svobodném přístupu k informacím, kde jsou tato data specifikována právními formulacemi. Právní předpis řeší formáty otevřených dat, způsob jejich zveřejnění v otevřeném a strojově čitelném formátu, jejich povinnou evidenci v Národním katalogu otevřených dat (Zákon č. 106/1999 Sb., 1999). K větší přehlednosti a lepší využitelnosti je nutné datové sady otevřených dat katalogizovat nejlépe ve vlastním lokálním katalogu otevřených dat. Tím, že jsou datové sady otevřených dat centralizovány do Národního katalogu otevřených dat, nabízí se možnost propojitelnosti více datových zdrojů pro různé aplikace (Velnerová, Poulová, 2025).

Proces přípravy datových sad otevřených dat je velice náročný z hlediska jejich kvality zpracování. Jelikož v současné době dochází k elektronizaci procesů napříč celou veřejnou správou, je nutné v první fázi metodicky upravit zadávání dat do informačních systémů, aby se následně získaná data dokázala zpracovat jednotným postupem, kde finálním výstupem jsou otevřená data připravená pro jejich další užití v jednotném standardu (Velnerová, Poulová, 2024). Ukázka již zpracované datové sady otevřených dat s jejím popisem v lokálním katalogu je na obrázku 1. Prostřednictvím otevřených dat je tak zajištěn trend s požadavkem na transparentnost veřejné správy, jelikož informace přichází veřejnosti nejen z hlediska kvantitativního, ale zejména kvalitativního s využitím současných moderních nástrojů k jejich vizualizaci (Velnerová, Poulová, 2024). Mnoho lokálních katalogů otevřených dat je součástí datových portálů, které významným způsobem zasahují taktéž do možností inovativní výuky ze strany pedagogů, jelikož jsou centrálními místy pro získání informací přehlednou a jednoduchou formou.



Obrázek 1: Ukázka popisu datové sady otevřených dat v lokálním katalogu otevřených dat.

Zdroj: Data KHK (2025)

2 Digitální dovednosti studentů

K současným trendům digitálních dovedností, kterými musí studenti v dnešní době disponovat, je umět pracovat s digitálními technologiemi, mít jistou úroveň informační gramotnosti, schopnost porozumět datům, umět je analyzovat a využít pro rozhodovací procesy. Pracovní trh přichází s těmito požadavky a výuka, která je založená pouze na teoretických příkladech užití dat nedokáže naplnit tyto požadavky budoucích zaměstnavatelů. Zapojením otevřených dat do výuky dává možnost studentům získat praktické zkušenosti s reálnými datovými sadami. Studenti se prakticky seznamují s nedokonalostmi dat, musí řešit různé možnosti jejich nápravy, aby je dokázali efektivně využít při použití různých technologických nástrojů, čímž získávají zkušenosti, které jsou v běžné praxi zcela běžným jevem (Velnerová, Poulová, 2025).

3 Hackathony

Dalším stupněm pro získání digitálních dovedností na základě práce s otevřenými daty je zařazení do pedagogické praxe soutěží tzv. hackathonů nad otevřenými daty, které jsou zároveň společenskou akcí, kde programátoři a analytici se sejdou za účelem vytváření softwaru (Collins, 2025). Je to významná událost pro nadané studenty a odborníky k řešení reálných potřeb, při které jsou užita otevřená data. Týmy studentů programují a navrhuji

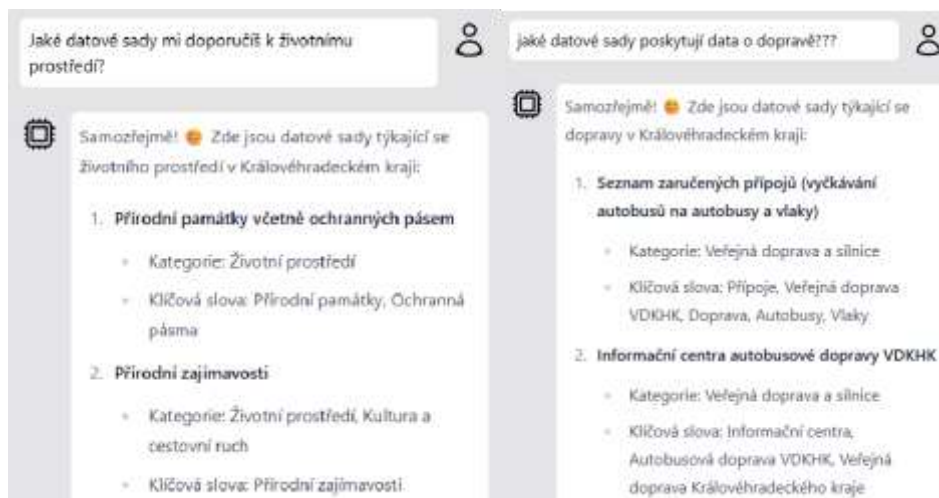
aplikace, které mohou mít praktické využití. Výsledkem mohou být aplikace, které může využívat široká veřejnost nebo najdou svoje uplatnění ve vzdělávání při výuce formou e- learning nebo online učení (Velnerová, Poulová, 2025). Během vyhodnocení výsledných návrhů aplikací studenti musí prokázat prezentaci svého řešení před týmem odborníků z praxe, čímž dostávají další velmi cennou zkušenost.

Přínosem těchto inovačních hackathonů nad otevřenými daty je možnost získání znalostí a dovedností během skutečné praxe. Hackathony jsou inspirací pro vytváření nových nástrojů ze strany studentů, rozvíjí jejich kreativitu, soutěživost a přímo vybízí k hledání inovativních postupů, aplikací a návrhů řešení. To napomáhá ke zlepšení a inovaci vzdělávacích procesů, podporuje se forma inteligentního učení s otevřenými vzdělávacími zdroji, rozvíjí se inteligentní vzdělávací prostředí, podporují se nové technologie ve výuce, jako je aplikace umělé inteligence (AI) (Velnerová, Poulová, 2025). Zároveň se tyto hackathony stávají silným a efektivním nástrojem pro navázání spolupráce mezi jednotlivými institucemi, kterými mohou být univerzity, ministerstva nebo územně samosprávné celky (Velnerová, Poulová, 2025). Užití otevřených dat a realizace hackathonů v rámci vzdělávání dynamickým tempem podporuje růst digitálních dovedností studentů, umožňuje rozvoj technologických inovací a výzkumných aktivit. A nejen to, studenti se rozvíjí i v oblasti měkkých dovedností jako je týmová spolupráce, vzájemná komunikace, práce pod časovým limitem a prezentace vlastních výsledků.

4 Příklad dobré praxe

Příkladnou ukázkou dobré praxe z krajských hackathonů nad otevřenými daty, je realizace návrhu aplikace, která slouží k efektivnímu vyhledávání datových sad otevřených dat nad lokálním katalogem otevřených dat. Návrh byl realizován formou výzkumných aktivit ve spolupráci s akademickou institucí a subjektem veřejné správy s užitím AI. Kromě zjednodušení práce s více datovými sadami najednou, aplikace umožňuje zobrazení dat ve formátu 3 D v denním a nočním režimu (Kubišta et al., 2024). Studenti mimo jiné zavedli do aplikace jazykové modely AI pro generování odpovědí na základě dotazů uživatelů. Tento systém je rovněž schopen vést rozšířenou konverzaci o dostupných datech a poskytovat relevantní informace uživatelům aplikace (Kubišta et al., 2024). AI byla ošetřena studenty inovační bariérou, která dokáže zabránit odpovědím na nerelevantní otázky (Kubišta et al., 2024). Ukázka této komunikace je na obrázku 2.

Uvedený příklad dobré praxe je inspirací, jak je možné pracovat s nadanými studenty formou inovativních metod, podpořit rozvoj jejich kreativního myšlení a využít jejich digitálních dovedností v praxi, což je pro studenty nesmírně cenná zkušenost. Tento příklad je ukázkový, jelikož tato aplikace získala ocenění na národní úrovni ze strany špičkových IT specialistů. Odborná veřejnost tak podpořila inovaci a smysluplnost pořádání hackathonů nad otevřenými daty. Prostřednictvím této studentské aplikace byla podpořena výzkumná činnost s využitím nejmodernějších technologií jako je AI. Aplikace byla uvedena do provozu, slouží veřejnosti, přispívá dále jako forma výukové pomůcky pro další studenty při práci s otevřenými daty, geografickými daty a mapami (Velnerová, Poulová, 2025). Studenti si navíc odnesli velice významnou zkušenost do svého profesního života.



Obrázek 2: Ukázka komunikace s uživateli aplikace na téma otevřených dat v rámci aplikace, jejíž návrh byl realizován v rámci hackathonu. Zdroj: katalog.dataahk.tech (2025)

5 Závěr

Otevřená data jsou fenoménem dnešní doby. Dávají jedinečnou příležitost studentům pracovat s reálnými daty, umožňují jim získat digitální dovednosti, které jsou významným přínosem pro jejich další uplatnění v praxi. Podpora veřejných institucí k otevírání dat napomáhá transparentnosti celé veřejné správy, a navíc má spoustu dalších přidaných hodnot. Těmi jsou možnosti jejich využití při pedagogické činnosti.

Pořádání hackathonů nad otevřenými daty efektivním a inovativním způsobem dokáže zapojit talentované a kreativní studenty do oblasti vývoje moderních nástrojů a aplikací, výzkumných aktivit. To vše podporuje vznik inteligentního vzdělávacího prostředí a inteligentního učení na konkrétních produktech (Velnerová, Poulová, 2025). Ty mohou být v praxi uplatněny pro efektivnější služby veřejnosti, ale i pro výuku mezi studenty. Výhodou dnešní doby je možnost nad těmito daty využívat AI jako nástroj k jejich pokročilé analýze, predikčním modelům, klasifikaci dat nebo vizualizacím.

Při hackathonech mohou vzniknout zajímavé mobilní aplikace, které se užívají jako učební materiály pro studenty bez omezení, a tím se může podpořit vzdělávání bez hranic. Možnosti uplatnění otevřených dat, hackathonů je velice rozmanité, nejen pro rozvoj a zkvalitnění služeb široké veřejnosti, ale i vývoj vzdělávacích nástrojů (Velnerová, Poulová, 2025). V neposlední řadě otevřená data a hackathony jsou inovativní cestou pro rozvoj dovedností studentů, posilují jejich schopnost prezentovat výsledky své práce, obhajovat vlastní řešení a napomáhají efektivní výuce k získání praktických zkušeností (Velnerová, Poulová, 2025).

Poděkování

Autor tímto děkuje vedoucímu své disertační práce doc. RNDr. Petru Tučníkovi, Ph.D. za odborné vedení, cenné konzultace a dlouhodobou podporu při vzniku tohoto příspěvku. Tento příspěvek byl podpořen specifickým výzkumným projektem Fakulty informatiky

a managementu UHK 2108 – Řešení aktuálních výzev v oblasti inteligentních měst a kybernetické bezpečnosti.

Literatura

- Collins Collins English English Dictionary [online], Dictionary. [cit. 2025-11-30]. Hackathon. Dostupné z: <https://www.collinsdictionary.com/dictionary/english/hackathon>
- Digitální a informační agentura ČR. Kontext otevřených dat. Digitální a informační agentura ČR [online], [cit. 2025-11-30]. Dostupné z: <https://opendata.gov.cz/informace:kontext:start>
- KUBIŠTA, Tomáš, Jakub DOLEŽAL a Jakub KYZR. *Výzkumná aplikace Mapa dat s AI ke katalogu otevřených dat* [online]. Praha: TRIADA, spol., 2024 [cit. 2025-11-30]. ISBN 978-80-907164-6-9. Dostupné z: https://2024.isss.cz/wp-content/uploads/sites/4/2024/05/ISSS2024_sbornik_opt.pdf
- ÚŘAD PRO PUBLIKACE EVROPSKÉ UNIE. Co jsou „otevřená data“? *Oficiální portál evropských dat* [online]. 2020 [cit. 2025-11-30]. Dostupné z: <https://data.europa.eu/cs/dataeuropa-academy/what-open-data>
- VELNEROVÁ, Radmila a Petra POULOVÁ. OPEN REGIONAL DATA AND THE USE OF AI. *Business Trends (Trendy v podnikání)* [online]. 2024, **2024**(2), 47-59 [cit. 2025-11-30]. Dostupné z: doi:https://doi.org/10.24132/jbt.2024.14.2.47_59
- VELNEROVÁ, Radmila a Petra POULOVÁ. Hackathons in Practical Education. *International Symposium on Educational Technology (ISET)* [online]. Institute of Electrical and Electronics Engineers, 2025, **2025**(1) [cit. 2025-11-30]. ISSN 2766-2144. Dostupné z: doi:10.1109/ISET65607.2025.00051
- Zákon č. 106/1999 Sb. ze dne 8. června 1999 o svobodném přístupu k informacím.

Kontaktní údaje

Ing. David Velner LL.M.

Fakulta informatiky a managementu Univerzity Hradec Králové

Hradecká 1249/6, 500 03 Hradec Králové 3

david.velner@uhk.cz

DIGITALIZACE SCINTILAČNÍCH SPEKTER: OD SKENOVANÝCH TABULEK K JSON POMOCÍ GENERATIVNÍ AI

DIGITIZATION OF SCINTILLATION SPECTRA: FROM SCANNED TABLES TO JSON USING GENERATIVE AI

Jan Zbirovský, Michal Jeřábek, Nikolaj Gribakin

Abstrakt

Článek prezentuje metodu automatizované extrakce dat z historických katalogů scintilační gama spektrometrie. Využitím Google AI Studio byla vyvinuta aplikace schopná převádět skenované tabulky obsahující závislosti četnosti detekce na energii jednotlivých izotopů do strukturovaného JSON formátu. Takto připravená data poslouží k trénování modelu schopného detekovat přítomnost konkrétních izotopů z naměřených spektrálních hodnot. Práce demonstruje praktické nasazení velkých jazykových modelů (LLM) a provádí srovnávací studii mezi cloudovým řešením a lokální inferencí. Výsledky ukazují, že zatímco cloudový model Google Gemini 3 Pro dosáhl 100% úspěšnosti, lokálně nasazené modely na hardwaru starší generace (architektura Pascal) selhaly vlivem nutné agresivní kvantizace, což poukazuje na limity "edge" nasazení pro úlohy vizuálního uvažování.

Klíčová slova: scintilační spektrometrie, generativní AI, datová extrakce, OCR

Abstract

This paper presents a method for automated data extraction from historical scintillation gamma-ray spectrometry catalogues. Using Google AI Studio, an application was developed capable of converting scanned tables containing count rate versus energy dependencies for individual isotopes into structured JSON format. The prepared data will be used to train a model capable of detecting the presence of specific isotopes from measured spectral values. The work demonstrates practical deployment of large multimodal models for digitizing archival scientific data and compares cloud-based SOTA models with local open-source alternatives. Results show that while Google Gemini 3 Pro achieved perfect accuracy, local models running on legacy hardware suffered from significant performance degradation due to quantization artifacts, highlighting the hardware constraints in modern VLM deployment.

Keywords: scintillation-spectrometry, generative AI, data extraction, OCR

Úvod

V rámci realizace projektu DURECO (CZ.02.01.01/00/23_021/0012564) zaměřeného na vývoj systémů pro detekci radionuklidů, jsme již v rané fázi řešení identifikovali zásadní překážku – nedostupnost referenčních spektrometrických dat v adekvátní digitální podobě. Tento deficit představuje kritický limitující faktor z hlediska návrhu a implementace řešení založených na algoritmech strojového učení, které vyžadují strukturované trénovací datové sety.

V první fázi realizace projektu bylo nezbytné tyto archivní zdroje transformovat do strukturovaného formátu vhodného pro další algoritmické zpracování. Vznikla tak potřeba nalézt rychlé a spolehlivé řešení pro automatizovanou extrakci dat. Tradiční metody optického rozpoznávání znaků (OCR), jako je Tesseract, se v tomto kontextu ukázaly jako nedostatečné. Jejich limitací je absence sémantického porozumění struktuře dokumentu – nedokáží efektivně rozlišit mezi datovými sloupci, metadaty a šumem. Naproti tomu moderní velké multimodální modely nabízejí schopnost tzv. "vizuálního uvažování" (visual reasoning), kdy model nahlíží na obraz jako na celek a interpretuje data v kontextu jejich fyzikálního významu. Tato práce prezentuje komparativní analýzu možných přístupů k řešení uvedené výzvy a hodnotí jejich efektivitu z hlediska přesnosti extrakce dat i hardwarové náročnosti.



Obrázek 1: Grafické znázornění celého procesu. Zdroj: Vlastní zpracování (2025)

1 Shrnutí dosavadních poznatků

Historické katalogy scintilační gama spektrometrie (Heath, R.L., 1957; Heath, R.L., 1964), obsahující referenční spektra jednotlivých izotopů, vznikaly převážně v 50. a 60. letech 20. století a v současnosti jsou dostupné výhradně ve formě naskenovaných archivních dokumentů. Tyto materiály, ačkoliv z vědeckého hlediska stále hodnotné, postrádají strojovou čitelnost. Pro naši úlohu byl zvolen dokument, kde jsou data k dispozici v tabulkové formě (Heath, R.L., 1964), jelikož první vydání obsahuje pouze grafické znázornění v nízkém rozlišení.

2 Data a metody

Celý datový set se skládá z 263 stránek naskenovaných tabulek obsahujících závislosti četnosti detekce na energii pro jednotlivé izotopy. Kvalita vstupních skenů odpovídala běžným archivním materiálům s typickými artefakty jako mírné geometrické zkreslení a degradace tisku.

Jako demonstrační testovací datový set byl použit vzorek 5 naskenovaných stran, kde byla vyhodnocena kvalita převodu dat na formát JSON. Pro knihovnu Tesseract byl proveden pouze

základní test v aplikaci dostupné na internetu (ScribeOCR). Vzhledem k vysoké četnosti chyb při rozpoznávání tabulkové mřížky bylo od dalších testů s klasickým OCR upuštěno (Obrázek 2).

4x5 DAY CALCIUM 47

DETECTOR 3X3-2 DATE 4-2-64

PLATE NO. 20-47-1 SOURCE DISTANCE 10 CM.

ENERGY SCALE 1.0 KEV/PHU ABSORBER 1.18 G/CM SQ. BE

9.666 X 10⁷ (1.300 MEV) GAMMAS EMITTED

	0	1	2	3	4	5	6	7	8	9
000	100000	025000	023000	021500	020292	019835	019465	018556	017400	016968
010	026112	016094	015048	015041	014631	014698	013902	013822	013609	013236
020	013485	014102	014570	014279	013222	012635	012255	012624	012417	012586
030	022406	012211	011781	051363	010844	010444	010456	010545	010252	010112
040	010039	009943	010170	010103	010721	012047	015430	020838	027500	031769
050	031142	025				010365	009892	009780	009722	009687
060	009511	009				008991	008854	008789	008935	008975
070	009066	009				011090	012572	014628	017153	019338
080	020597	020				012715	011534	011011	010629	010906
090	010670	010				011334	011311	011658	011833	011874
100	011881	011705	011525	010967	010298	009337	008317	007279	006186	005220
110	004352	003859	003292	002360	002623	002491	002709	003150	003808	005515
120	008640	013201	020067	028846	038513	047503	054335	056805	054533	049447
130	041022	031937	023168	016022	010418	006404	003922	002271	001386	000847
140	000534	000296	000252	000158	000122	000070	000049	000036	000049	000028
150	000029	000030	000027	000016	000032	000037	000022	000025	000021	000030
160	000026	000000	000000	000000	000000	000000	000000	000000	000000	000000

Obrázek 2: Příklad rozpoznání dat v dokumentu s pomocí knihovny Tesseract.

Zdroj: Vlastní zpracování (2025)

2.1 Konfigurace modelů a hardwaru

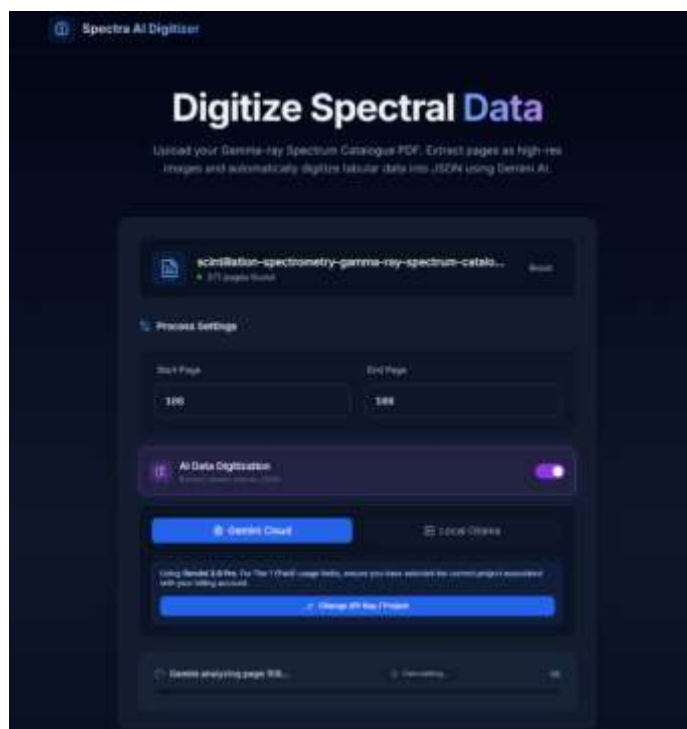
V rámci experimentu byly porovnány tři odlišné přístupy:

1. **Klasické OCR:** Open-source knihovna Tesseract.
2. **Lokální open-source modely:** Gemma 3 27B (Google), Qwen 3 VL 32B (Alibaba Cloud), Llava 34B (Microsoft/Open source).
3. **Cloudový SOTA (State-Of-The-Art) model:** Google Gemini 3 Pro.

Lokální inference probíhala na pracovní stanici vybavené grafickou kartou NVIDIA Quadro P6000 s 24 GB VRAM (architektura Pascal). Je důležité poznamenat, že tato architektura nepodporuje nativně instrukce BF16 (Brain Float 16), které jsou standardem pro moderní LLM. Pro spuštění modelů o velikosti cca 30 miliard parametrů v limitu 24 GB VRAM bylo nutné použít framework **Ollama** a agresivní **4-bitovou kvantizaci (Q4_K_M)**. Tato konfigurace představuje "stress-test" schopností provozovat nejnovější AI modely na starším, avšak profesionálním hardwaru.

2.2 Vývoj aplikace metodou „vibe coding“

Vývoj uživatelského rozhraní a extrakčních skriptů byl realizován s využitím metodiky **AI-assisted development**, v průmyslové praxi populárně označované jako **"vibe coding"** (termín zpopularizoval Andrej Karpathy v roce 2025). Tento přístup spočívá v delegování implementačních detailů (syntaxe kódu) na LLM (v tomto případě Gemini 3 Pro), zatímco vývojář se soustředí na definici logiky, vstupů a validaci výstupů. Aplikace byla vytvořena v prostředí Google AI Studio.



Obrázek 3: Aplikace na extrakci dat z PDF. Zdroj: Vlastní zpracování (2025)

3 Výsledky a diskuze

Vyhodnocení proběhlo zvlášť pro hlavičku i pro data samotná (Tabulka 1). V případě shody s originálním obsahem byl započítán platný převod.

Tabulka 1: Příklad porovnání LLM modelů při zpracování scintilačních dat (Heath, R.L., 1964)

Poznámka: Celkový počet položek/správně převedených; tučným písmem označen nejlepší výsledek

Strana	Gemini 3 Pro		Gemma3:27b		Qwen3-vl:32b		Llava:34b	
	Hlavička	Data	Hlavička	Data	Hlavička	Data	Hlavička	Data
108	8/8	70/ 70	8/3	70/1	8/0	70/0	8/0	70/1
109	8/ 7	186/ 186	8/1	186/0	8/0	186/0	8/0	186/0
110	8/8	229/ 229	8/2	229/0	8/0	229/0	8/0	229/0
111	8/8	310/ 310	8/2	310/0	8/0	310/0	8/0	310/0
112	8/8	120/ 120	8/1	120/0	8/0	120/0	8/0	120/0

Zdroj: Vlastní zpracování (2025)

Tabulka 2: Procentuální vyhodnocení přesnosti jednotlivých LLM na datech z Tabulky 1

Model	Gemini 3 Pro		Gemma3:27b		Qwen3-vl:32b		Llava:34b	
	Hlavička	Data	Hlavička	Data	Hlavička	Data	Hlavička	Data
Celkem	40	915	40	915	40	915	40	915
Správně	39	915	9	1	0	1	0	0
Přesnost	97,50%	100,00%	22,50%	0,11%	0,00%	0,11%	0,00%	0,00%

Zdroj: Vlastní zpracování (2025)

3.1 Analýza selhání lokálních modelů

Výsledky v Tabulce 2 ukazují drastický rozdíl mezi cloudovým a lokálním řešením. Zatímco Gemini 3 Pro exceloval, modely Qwen 3 VL a Gemma 3 dosáhly téměř nulové úspěšnosti. Tento výsledek je na první pohled v rozporu s globálními benchmarky, kde Qwen 3 VL typicky dosahuje špičkových výsledků v OCR úlohách.

Detailní analýza však ukazuje na kritický vliv hardwarových limitací naší testovací sestavy.

1. **Ztráta přesnosti kvantizací:** Aby bylo možné vměstnat modely s parametry 27B/32B do 24 GB VRAM, byla použita 4-bitová kvantizace. Vizuální enkodéry (např. SigLIP používaný v modelech Gemma nebo Qwen VL) jsou na tuto kompresi extrémně citlivé.
2. **Architektura Pascal:** Absence Tensor Cores pro BF16 na kartě P6000 si vyžádala neefektivní převody datových typů během výpočtu, což dále degradovalo výkon.
3. **Prompt:** Ve všech scénářích s LLM, byl použit prompt s technikou „one-shot“ (jeden příklad formátu výstupu), který může být také důvodem selhání lokálních modelů. Možnou optimalizací promptu (prompt engineering) a použitím „few shot“ techniky (zpravidla 3-5 příkladů výstupů), by mělo být možné výsledky zlepšit.

Tento výsledek je tedy nutné interpretovat nikoliv jako selhání architektury modelů Qwen/Gemma, ale jako demonstraci toho, že pro efektivní provoz moderních VLM (Vision-Language Models) v plné kvalitě je nezbytný hardware nové generace s podporou BF16 a vyšší kapacitou VRAM (např. NVIDIA RTX 6000 Ada nebo A100).

Naproti tomu model Gemini 3 Pro, běžící na nativní infrastruktuře Google (TPU), nebyl těmito limity svázán a mohl plně využít své kapacity pro bezchybnou rekonstrukci dat.

Závěr

Předložená práce demonstrovala úspěšnou aplikaci generativní AI pro digitalizaci historických katalogů scintilační gama spektrometrie. Komparativní analýza potvrdila, že pro extrakci komplexních vědeckých dat z nekvalitních skenů je v současnosti (rok 2025) nejspolehlivějším řešením využití proprietárních cloudových modelů (Gemini 3 Pro), které dosáhly 100% přesnosti.

Experiment s lokálními modely přinesl cenný negativní výsledek: prokázal, že nasazení velkých multimodálních modelů "on-premise" na starším hardwaru s sebou nese riziko kritické ztráty přesnosti vlivem nutné kvantizace. Vyvinutá aplikace využívající API Gemini 3 Pro tak umožňuje efektivní transformaci archivních dat do JSON formátu pro další strojové zpracování v projektu DURECO.

Poděkování

Autor děkuje Vysoké škole polytechnické Jihlava za možnost podílet se na řešení projektu DURECO (CZ.02.01.01/00/23_021/0012564) a členům řešitelského týmu za dosavadní přínosnou spolupráci.

Literatura

- Alibaba Cloud (2025). Qwen 3 VL: A Versatile Vision-Language Model. [online], [2025-12-12]. Dostupné z: <https://github.com/QwenLM/Qwen3-VL>
- Google AI Studio [online]. Dostupné z: <https://aistudio.google.com/>
- Heath, R.L., 1957. SCINTILLATION SPECTROMETRY--GAMMA-RAY SPECTRUM CATALOGUE (No. IDO-16408, 4305154). <https://doi.org/10.2172/4305154>
- Heath, R.L., 1964. SCINTILLATION SPECTROMETRY. GAMMA-RAY SPECTRUM CATALOGUE. 2nd Edition. Volume 1 of 2 (No. IDO-16880-1, 4033555). Dostupné z: <https://doi.org/10.2172/4033555>
- Karpathy, A. (2025). There's a new kind of coding I call "vibe coding". [online], [2025-03-07]. X (formerly Twitter). Dostupné z: <https://x.com/karpathy/status/1886192184808149383>
- Ollama [online]. Dostupné z: <https://ollama.com/search?c=vision>
- ScribeOCR [online]. Dostupné z: <https://scribeocr.com/>

Kontaktní údaje

Ing. Jan Zbirovský
Katedra technických studií
Vysoká škola polytechnická Jihlava, Tolstého 16, 586 01 Jihlava
e-mail: Jan.Zbirovsky@vspj.cz
<https://orcid.org/0009-0001-3738-9262>

Ing. Jan Zbirovský
Katedra mechaniky a materiálů
České vysoké učení technické v Praze, Fakulta dopravní, Konviktská 20, 110 00 Praha 1
e-mail: zbirovj@cvut.cz