

# Mobilná aplikácia s umelou inteligenciou pre deti

## Mobile application with artificial intelligence for children

Jaroslav Knežník  
Univerzita Mateja Bela  
Tajovského 40  
97401 Banská Bystrica  
Slovenská Republika  
jkneznik@student.umb.sk

Jarmila Škrinárová  
Univerzita Mateja Bela  
Tajovského 40  
97401 Banská Bystrica  
Slovenská Republika  
jarmila.skrinarova@umb.sk

Patrik Voštinár  
Univerzita Mateja Bela  
Tajovského 40  
97401 Banská Bystrica  
Slovenská Republika  
patrik.vostinar@umb.sk

### ABSTRACT

The goal of the paper is to present a methodology for creating artificial intelligence models for image recognition. These models can be used in programming environments suitable for teaching programming to children. We describe the "Search." methodology, by which we create a model of artificial intelligence for image recognition. Subsequently, we focus on the description of the methodology of creating a mobile application in the App Inventor environment, in which the model of artificial intelligence is used.

### Keywords

Programming for children. Teaching programming. Artificial intelligence. Artificial intelligence for kids. Programming with artificial intelligence. AppInventor.

### ABSTRAKT

Cieľom tohto článku je prezentovať metodiku tvorby modelov umelej inteligencie na rozpoznávanie obrázkov. Tieto modely je potom možné používať v programovacích prostrediach, vhodných na výučbu programovania pre deti. V príspevku opisujeme metodiku s názvom „Hľadaj.“, pomocou ktorej vytvoríme model umelej inteligencie na rozpoznávanie obrázkov. Následne sa zameriavame na opis metodiky tvorby aplikácie v prostredí App Inventor, v ktorom je použitý model umelej inteligencie.

### Kľúčové slová

Programovanie pre deti. Vyučovanie programovania. Umelá inteligencia. Umelá inteligencia pre deti. Programovanie s umelou inteligenciou. AppInventor.

## 1 ÚVOD

V súčasnosti veľmi často počujeme pojem: „umelá inteligencia“. Aplikácie a technológie, na báze umelej inteligencie, používame denne, bez toho, aby sme si to uvedomovali. Typickým príkladom je rozpoznávanie odtlačku prsta alebo scan tváre, ktorý používame pri prístupe do nášho telefónu. Ďalším príkladom je nasadenie „softvérových robotov“ na vyhľadávanie obsahu webových stránok, ktoré implementovala napríklad spoločnosť Google. Vďaka tejto technológii máme vo vyhľadávачi zobrazené odkazy na webové stránky za pár sekúnd.

Čo je inteligencia? Definícií pojmu inteligencia je veľa. Autori Shane Legg a Marcus Hutter v práci „A Collection of Definitions of Intelligence“, v roku 2007, uviedli 71 definícií inteligencie. Pričom 18 definícií predstavuje definície zo slovníkov a encyklopédií, 35 definícií od vedcov z oblasti psychológie a 18 od vedcov z oblasti umelej inteligencie. Jedna z nich definuje ľudskú inteligenciu ako: „Schopnosť využívať pamäť, vedomosti, skúsenosti, porozumenie, myslenie, predstavivosť a úsudok s cieľom vyriešiť problémy a prispôbiť sa novým situáciám“ [1]. Z definície je zrejmé, že ľudská inteligencia predstavuje

kompozíciu schopností, ktoré sa postupne rozvíjajú a navzájom sa ovplyvňujú. Tiež je zrejmé, že ani súčasný moderný svet so všetkými poznatkami a technológiami nie je pripravený vyvinúť takú komplexnú umelú inteligenciu ako je ľudská.

Umelá inteligencia sa zaoberá vývojom algoritmov a technológií, ktoré modelujú určité prvky inteligentného správania sa. Tieto prvky správania sa sú odpozorované od biologických systémov a sú podstatne zjednodušené.

Podľa [3] je umelá inteligencia: „...schopnosť systému primerane konať v neurčitom prostredí, ... je to činnosť, ktorá zvyšuje pravdepodobnosť úspechu na dosiahnutie čiastkových cieľov správania sa tak, aby systém splnil konečný cieľ.“ [2,3]

Medzi základné úlohy umelej inteligencie patrí:

- Rozpoznávanie obrazu, zvuku, textu.
- Klasifikácia, t.j. zatriedenie prvkov alebo objektov do určitých skupín.
- Predikcia, ktorá na základe predchádzajúcich znalostí predpovedá výstupy v ďalších krokoch.

Cieľom tohto článku je uviesť príklady, metodiku a primerané aktivity, ktoré môžu učitelia informatiky použiť pri vyučovaní programovania. Metodika, na základe praktického príkladu, rozvíja praktické zručnosti v programovaní a súčasne rozvíja porozumenie práce s umelou inteligenciou, zameranou na rozpoznávanie obrazu. Cieľom je, aby po použití metodiky, žiak chápal čo je potrebné urobiť, aby umelá inteligencia rozpoznala obraz a dokázal si toto poznanie zovšeobecniť na celú triedu systémov, určených na rozpoznávanie pomocou umelej inteligencie.

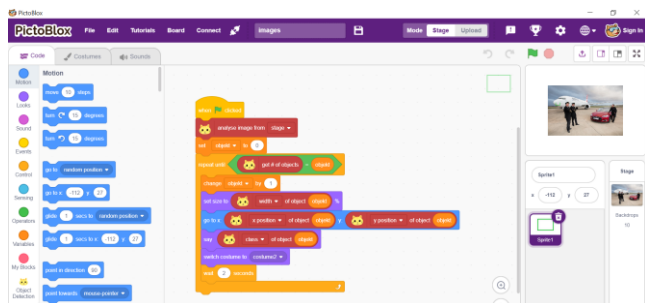
V kapitole 2 uvádzame niekoľko príkladov dostupných softvérových nástrojov a metodík, ktoré vytvárajú modely umelej inteligencie. Tieto modely je potom možné používať v programovacích prostrediach, vhodných na vyučovanie programovania pre deti. V kapitole 3 opisujeme metodiku tvorby príkladov s umelou inteligenciou. V tejto kapitole sme predstavili metodiku „Hľadaj“, pomocou ktorej vytvoríme model umelej inteligencie na rozpoznávanie obrázkov. V kapitole 4 sa zameriavame na opis metodiky tvorby programu v online prostredí App Inventor, v ktorom je použitý model umelej inteligencie vytvorený v kapitole 3.

## 2 UMEĽÁ INTELIGENCIA PRE DETI

V súčasnosti (najmä v zahraničí) patrí pojem umelá inteligencia na školách k čoraz viac skloňovaným témam. Je dôležité pracovať s umelou inteligenciou aj na slovenských školách. Široký pojem „umelá inteligencia“ je možné žiakom predkladať radšej formou čiastkových úloh a na príkladoch ich naučiť pracovať s umelou inteligenciou. Dobrým príkladom je rozpoznávanie obrazu. V súčasnosti existuje viacero aplikácií, ktoré využívajú princípy

umelej inteligencie a ich vývojové prostredia sú podobné tým, ktoré sa už dnes využívajú na základných a stredných školách.

Jednou z takýchto aplikácií je PictoBlox<sup>1</sup>. PictoBlox obsahuje blokové prostredie, ktoré je veľmi podobné Scratchu a umožňuje nám rozoznávať objekty, gestá, tváre, hudobné tóny a mnoho iného. Na obrázku 1 je zobrazené programovacie prostredie aplikácie PictoBlox.



Obrázok 1: Ukážka blokového prostredia Pictobloxu

Na obrázku 2 môžeme vidieť výsledok rozpoznania obrazu. Vľavo, na obrázku 2, aplikácia rozpoznala, že dievčatko nemá rúško a postavička hovorí: „prosím, nos rúško, aby si sa ochránila“. Vpravo, na obrázku 2, aplikácia rozpoznala, že dievčatko má rúško a postavička hovorí: „ďakujem, že nosíš rúško“.



Obrázok 2: Výstup z programu na rozpoznanie obrázku v Pictoblox [4]

MIT App Inventor<sup>2</sup> ponúka sériu 6 metodík vhodných pre predmet informatika na nižšom a vyššom stupni sekundárneho vzdelávania. Sú zamerané na spoznávanie základných konceptov a postupov typických pre aplikácie umelej inteligencie. Dôraz sa kladie na experimentovanie, objavovanie a tvorivý prístup pri programovaní vlastných mobilných aplikácií. Okrem podrobnej osnovy vyučovacej hodiny sú k dispozícii aj rôzne podporné materiály (prezentácie pre učiteľov, videá, sprievodcovia tvorbou mobilných aplikácií, vzorové riešenia) [5].

Ďalšie zaujímavé metodiky a príklady pre využitie umelej inteligencie na školách ponúkajú nástroje na webstránke Machine learning for kids<sup>3</sup>. Nástroj umožňuje, na základe našich vstupných dát, natrénovať model. Takto natrénovaný model použijeme v našom programe. Použitý model rozhodne, či sa objekt alebo tvar, ktorý bol na vstupných dátach zhoduje s tým, ktorý chceme rozlišovať. Ak sú vstupné dáta obrázky, potom model rozpozná, či sa na obrázku nachádza naučený objekt. Metodiky a nástroje, ktoré urobia učiteľom a študentom pri vyučovaní radosť vytvoril pán Dale Lane [6].

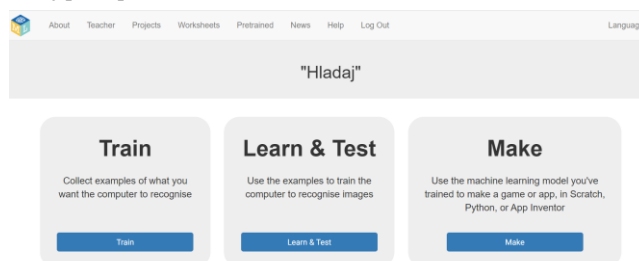
### 3 AKTIVITA HĽADAJ V „MACHINE LEARNING FOR KIDS“

V tomto článku opíšeme príklad a aktivitu „Hľadaj“, ktorú sme vytvorili pomocou nástroja na stránke Machine learning for kids. Najprv vytvoríme nový projekt, ten v sebe bude zahŕňať naše vstupné dáta a taktiež aj model, ktorý je potrebné na základe týchto dát natrénovať. Pri vytváraní projektu je možné zvoliť si z možností text, obrázky, čísla a zvuky. My sme zvolili „obrázky“. To znamená, že budeme rozpoznať obrázky.

Keď už máme projekt vytvorený zobrazia sa nám 3 kroky v postupe, ktoré je potrebné dodržať (pozri Obrázok 3.):

- Trénuj (angl. Train).
- Uč sa a testuj (angl. Learn & Test).
- Vlož (Make).

Tieto možnosti hovoria o tom, akým spôsobom sa má v projekte ďalej postupovať.



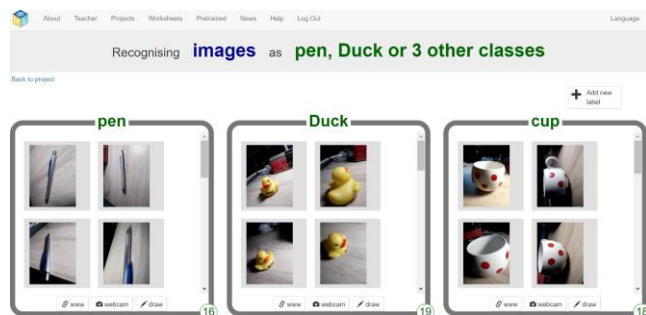
Obrázok 3: Kroky postupu

Časť *Train* je prvá časť, do ktorej je potrebné vstúpiť po vytvorení projektu a zároveň aj najdôležitejšia časť projektu. V tejto časti musíme poskytnúť nášmu projektu dostatočné množstvo vstupných dát, v našom prípade obrázkov, na základe ktorých sa natrénuje model. Dáta sú vložené do kategórií, ktoré sú posudzované osobitne. Model bude po natrénovaní schopný rozlišovať medzi týmito kategóriami. Počet vstupných dát, pre každú kategóriu musí byť väčší ako desať, aby bolo možné model úspešne natrénovať. Je však dobré poskytnúť modelu väčšie množstvo vstupných dát, pretože od množstva vstupných dát sa odvíja aj presnosť, s akou je model schopný rozoznať čo sa na obrázku nachádza. Vhodné je začať tak, že sa zvolené objekty budú od seba výrazne odlišovať. My sme sa rozhodli, že budeme rozlišovať objekty pero, kačička a šálka (pozri Obrázok 4). Fotografia získame kliknutím na ikonu „webcam“ v spodnej časti každej kategórie. My sme každý objekt odfoťili viackrát, konkrétne kačičku 19-krát, šálku 18-krát a pero 16-krát. Pre objekty, ktoré sú si viac podobné je potrebné vytvoriť a vložiť na trénovanie viac vstupných súborov (fotografií).

<sup>1</sup> <https://thestempedia.com/product/pictoblox/>

<sup>2</sup> <https://appinventor.mit.edu/>

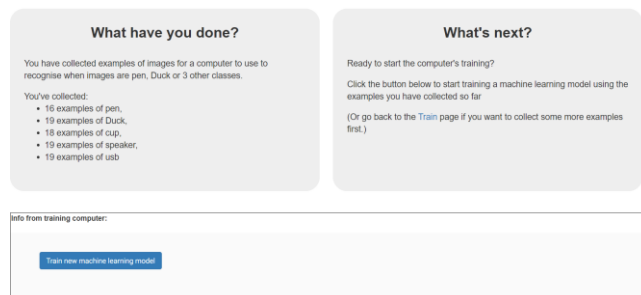
<sup>3</sup> <https://machinelearningforkids.co.uk/>



Obrázok 4: Kategórie vstupných dát

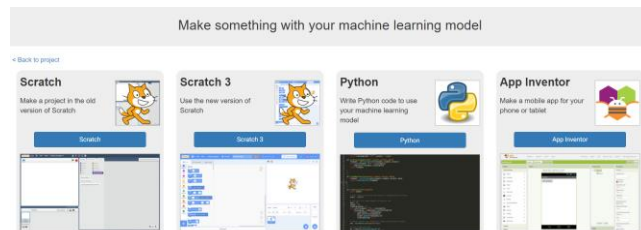
Ak už máme dostatočný počet obrázkov (fotografií) pre šálku, kačičku a pero môžeme sa presunúť do časti Learn & Test. My sme sa však rozhodli nášho projektu pridať ďalšie 2 objekty. (USB kľúč a reproduktor). A až potom sme sa presunuli do časti Learn & Test.

V časti Learn & Test sa model naučí rozpoznávať obrázky. Systém vypíše čo z celého procesu sme už urobili a čo je potrebné spraviť ďalej (pozri Obrázok 5.). Klikneme na tlačidlo Train new machine learning model a systém automaticky natrénuje model. Trvá to spravidla pár minút.



Obrázok 5: Natrénovanie modelu

Poslednou časťou postupu je vloženie modelu do aplikácie (pozri Obrázok 6.), tu je potrebné implementovať vytvorený model do jedného z dostupných programovacích prostredí. Na výber je z možností Scratch, Python a App Inventor. V našom prípade sme vybrali App Inventor. Podľa inštrukcií pridáme vytvorený model ako rozšírenie do vytvoreného programu v App Inventore. Na tento krok sa bližšie pozrieme v kapitole 4.



Obrázok 6 Programovacie jazyky (prostredia), do ktorých je možné vložiť model

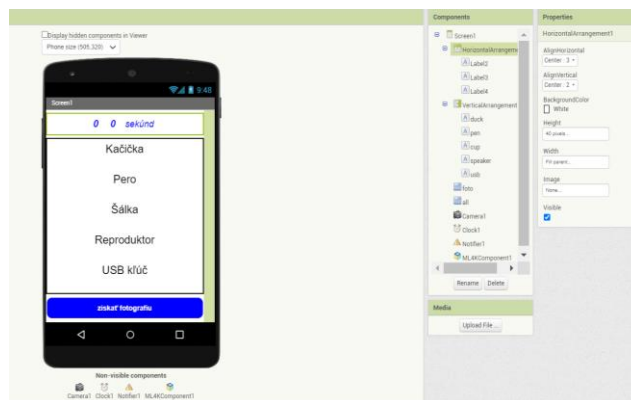
## 4 TVORBA WEBOVEJ APLIKÁCIE POMOCOU APPINVENTOR – TRETÍ KROK V METODIKE HĽADAJ

V predchádzajúcej časti sme opísali prvé 2 kroky metodiky „Hľadač“. V tejto kapitole opíšeme postup tvorby mobilnej aplikácie. Aplikácia bude využívať model vytvorený pomocou predchádzajúcej kapitoly. Model bude rozoznávať či sa na

fotografii získanej pomocou našej aplikácie nachádza kačička, pero, šálka, USB kľúč alebo reproduktor. Fotografie získame pomocou fotoaparátu priamo cez aplikáciu. V poslednej časti Make po výbere nástroja App Inventor dostaneme vygenerovanú URL adresu. Na tejto adrese sa nachádza naučený model, adresu vložíme do nášho novovytvoreného projektu v App Inventore ako rozšírenie. Toto rozšírenie tu nájdeme pod skratkou „ML4KComponent“. V Prostredí AppInventoru sa náš projekt skladá z grafickej časti (časť označovaná ako dizajnér) a z časti bloky, ktorá obsahuje programovacie prostredie.

### 4.1 Postup tvorby vzhľadu mobilnej aplikácie

V prostredí dizajnéra vytvárame vzhľad mobilnej aplikácie. Dizajnér obsahuje políčka, ktoré sa budú vyfarbovať, ak sa daný objekt nájde a súčasne sa bude zaznamenávať čas, nameraný pomocou stopiek. Pri vytváraní dizajnéra budeme pridávať komponenty z palety v ľavej časti obrazovky. Najprv vytvoríme hlavnú časť, v tejto časti sa budú nachádzať názvy našich kategórií (pozri Obrázok 7.).



Obrázok 7: Tvorba vzhľadu mobilnej aplikácie – dizajnér

Z časti „Extensions“ z palety komponentov pridáme model „ML4KComponent1“ a komponent „Vertical Arrangement“ (zvislé usporiadanie) z časti „Layout“ (rozloženie) a vložíme ho na obrazovku. Komponent „label“, ten vložíme do komponenty „Vertical Arrangement“.

V časti „Properties“ (vlastnosti), zmeníme text, jeho veľkosť a umiestnenie pomocou komponentu „label“. V časti „Components“ premenujeme „Label1“ na „cup“. Tento krok zopakujeme pre všetkých 5 kategórií obrázkov.

Následne vytvoríme tlačidlo tak, že vyberieme z palety „button“ (tlačidlo), vložíme ho pod „Vertical Arrangement“ a upravíme parametre tlačidla.

Teraz potrebujeme vytvoriť ukazovateľ stopiek, preto z palety pridáme: 1x „Horizontal Arrangement“ z časti „Layout“, 3x komponent „label“ a upravíme parametre každého „label“ komponentu

Nakoniec z palety pridáme aj neviditeľné komponenty „Camera“, „Clock“ a „Notifier“ a dizajnér máme hotový (pozri Obrázok 7).

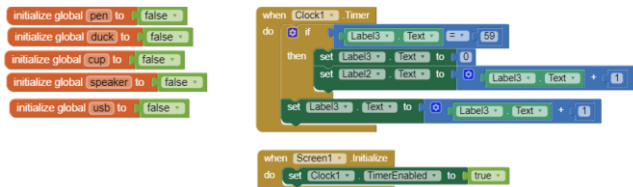
### 4.2 Postup tvorby programu s modelom z umelej inteligencie

V časti bloky naprogramujeme funkcionality jednotlivých komponentov, ktoré sme vytvorili v časti dizajnér. Cieľom je, aby aplikácia pracovala takto:

- Po spustení aplikácie sa spustia stopky, tie budú zobrazovať čas v hornej časti obrazovky v minútach a sekundách.
- Po stlačení tlačidla „získať fotografiu“ sa spustí fotoaparát,
- po získaní fotografie sa fotografia odošle na rozpoznávanie vloženému modelu,
- Model určí, ktorá z vecí (kačička, pero,...) sa nachádza na fotografii, a vyfarbí pozadie názvu danej veci zelenou farbou v mobilnej aplikácii.
- Po odfotení všetkých určených vecí sa zobrazí notificačná hláška spolu s dosiahnutým časom.

Postup tvorby programu:

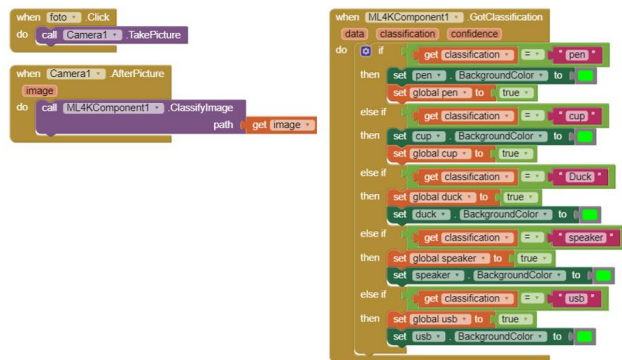
Najprv inicializujeme globálne premenné, ktoré si vytvoríme na hodnoty „false“ pre všetky kategórie obrázkov (pozri Obrázok 8 vľavo).



**Obrázok 8: Inicializácia premenných (vľavo), práca stopiek (vpravo)**

Pri spustení rozpoznávania sa spustia stopky, čas ktorý stopky zaznamenávajú sa zobrazuje. Postačí nám ukazovateľ minút a sekúnd (pozri Obrázok 8.).

Po stlačení tlačidla „foto“ sa na telefóne spustí fotoaparát, fotografia sa následne odošle do modelu. Z blokov nášho modelu vyberieme „Classify image“, tento blok zavolá funkciu kategórie, ktorá pošle fotografiu do nášho modelu (pozri Obrázok 9.).



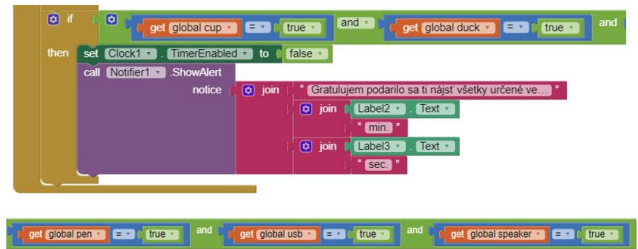
**Obrázok 9 Funkcia fotoaparát (vľavo), rozpoznávanie a výstup z modelu UI (vpravo)**

Použijeme model umelej inteligencie *ML4KComponent1*, ktorý sme vytvorili v kapitole 3. Model rozpozná, do ktorej kategórie fotografia patrí a vráti nám naspäť názov kategórie.

Pomocou podmienky „IF-ELSE“ zistíme názov kategórie. Pri kategórii, pre ktorú je podmienka splnená, zmeníme hodnotu príslušnej premennej na „true“ (pozri Obrázok 9). Tým sa farba pozadia príslušného názvu kategórie obrázku na mobilnej aplikácii na zelenú (pozri Obrázok 11.).

Na koniec tohto cyklu pridáme podmienku tak, aby sme zistili, či sú všetky premenné nastavené na hodnotu „true“ (či boli správne rozpoznané všetky veci). Ak sme našli všetky určené veci, zastavia

sa stopky a zavolá sa okno „notifier“ s výpisom: „Gratulujem, podarilo sa ti nájsť všetky určené veci“.



**Obrázok 10: Podmienka ukončenia rozpoznania všetkých objektov**

Na obrázku 11 je zobrazená obrazovka našej vytvorenej aplikácie.



**Obrázok 11: Ukážka bežiaciej aplikácie s umelou inteligenciou**

## 5 ZÁVER

Základ metodiky, ktorá je opísaná v tomto článku, pôvodne tvorila jedna z priebežných úloh v predmete Didaktika informatiky, ktorý je vyučovaný na Katedre informatiky, FPV UMB v Banskej Bystrici. Študenti druhého ročníka magisterského štúdia v programe Učiteľstvo informatiky v kombinácii dostali za úlohu vytvoriť programátorské aktivity, ktoré využívajú prvky umelej inteligencie a zároveň sú určené pre deti základných škôl. Študentom bolo predstavených niekoľko nástrojov a príkladov z oblasti umelej inteligencie pre deti. Niekoľko z nich sme uviedli v kapitole 2. Z pohľadu učiteľov môžeme konštatovať, že študenti úspešne splnili úlohu a vytvorili programy a aktivity, pričom využili nástroj machine learning for kids. Študenti, po splnení úloh v rámci spätnej väzby, uviedli:

- „...úloha skvelo prepája prvky umelej inteligencie a softvéru (napr. Scratch a App Inventor), ktorý žiaci dobre poznajú...“
- „Vybral som si App Inventor a vytvoril jednoduchú mobilnú aplikáciu. Táto aplikácia rozpoznáva určité veci zachytené na fotografii, pričom komunikuje s modelom umelej inteligencie a na základe odpovede, ktorú dostane zobrazuje

čo sa nachádza na fotografii. Domnievam sa, že tento nástroj má obrovský potenciál a je jedným z dôležitých krokov, ktoré umožňujú umelej inteligencii dostať sa už k žiakom základných škôl.“

- „Pred touto aktivitou som sa s umelou inteligenciou nestretol, no zaoberal som sa detským programovaním a preto sa domnievam, že pravé aktivity tohto druhu dokážu najmä u žiakov základných škôl vzbudiť veľký záujem. Pre mňa osobne bola práca s nástrojom machine learning for kids veľmi zaujímavá a myslím, že rovnaký efekt by to malo aj pre žiakov, ktorých zaujíma všetko nové a neprebádané. V neposlednom rade to prinesie oživenie programovania a hodín informatiky, nakoľko možnosti využitia tohto nástroja sú obmedzené len fantáziou žiakov.“

K tejto aktivite sme vypracovali metodiku, ktorú študenti pôvodne plánovali overiť na základnej škole, kde absolujú pedagogickú prax alebo na informatickom krúžku, ten navštevujú deti zo základných škôl v Banskej Bystrici. No, žiaľ kvôli nepredvídaným okolnostiam pandémie sa nám nepodarilo navštíviť školu a overiť túto metodiku. Ostáva nám len dúfať, že sa situácia v dohľadnej dobe zlepší a my sa opäť plnohodnotne vrátíme do našich životov.

## POĎAKOVANIE

Príspevok je podporený grantovým projektom Implementácia nových trendov v informatike do výučby algoritmického myslenia a programovania v predmete informatika v sekundárnom vzdelávaní, projekt KEGA 018UMB-4/2020.

## BIBLIOGRAFICKÉ ODKAZY

- [1] LEGG, S., HUTTER, M. A collection of Definitions of Intelligence. In *Frontiers in Artificial Intelligence and Applications*. 2007, vol.157, p. 17-24.
- [2] LEGG, S., HUTTER, M. Universal Intelligence: A Definition of Machine Intelligence. *Minds and machines*. 2007, vol. 17, no. 4, p. 391-44. ISSN: 0924-6495. DOI= 10.1007/s11023-007-9079-x.
- [3] ALBUS, J., S. Albus. Outline for a theory of intelligence. *IEEE Transaction of Systems, Man and Cybernetics*. 1991, vol. 21, no. 3, p. 473–509.
- [4] PICTOBLOX. *Thestempedia.com* [online]. 2021 Pictoblox. [cit. 2021-01-08]. Dostupné na: <https://thestempedia.com/shop/online-courses/artificial-intelligence-for-kids/>
- [5] MIT App Inventor. Artificial Intelligence with MIT App Inventor. Dostupné na internete: <https://appinventor.mit.edu/explore/ai-with-mit-app-inventor>
- [6] LANE, D. *Bringing AI into classroom* [online] [cit. 2021-01-27]. Dostupné na internete <https://dalelane.co.uk/blog/?m=202002>