

DALL'ESTERO

Dr. Scratch: supportare gli insegnanti nella valutazione del pensiero computazionale

Jesús Moreno-León¹, Gregorio Robles¹ e Marcos Román-González²

¹Universidad Rey Juan Carlos. Madrid, Spain.

²Universidad Nacional de Educación a Distancia, Spain

traduzione di Mara Masseroni, masseroni.mara@gmail.com

Vi sono molte tecnologie create con l'obiettivo di aiutare gli studenti nello sviluppo delle loro capacità di pensiero computazionale, ad esempio Logo, Alice e, in particolare, Scratch, che è diventato il linguaggio di programmazione standard per insegnare informatica nelle scuole. Tuttavia, vi è una mancanza di strumenti che supportano sia gli insegnanti sia gli studenti nella valutazione di tale competenza.

[Dr. Scratch](#) è uno strumento open source che analizza i progetti Scratch e offre un feedback sia agli educatori che ai discenti assegnando un punteggio ai vari progetti che sviluppano il pensiero computazionale.

Questo punteggio, che va da 0 a 21 punti, si basa sul grado di sviluppo delle diverse dimensioni della competenza di pensiero computazionale, in particolare astrazione e scomposizione dei problemi, il pensiero logico, la sincronizzazione, il parallelismo, nozioni algoritmiche di controllo del flusso, interattività con l'utente e la rappresentazione dei dati, che vengono valutati esaminando il codice sorgente del progetto analizzato. Dr. Scratch rileva anche alcune cattive abitudini di programmazione o potenziali errori che si trovano comunemente nei progetti presenti nella repository Scratch, come ad esempio nomi sprite non significativi, la ripetizione del codice, o un codice che non viene mai eseguito e/o la scorretta inizializzazione degli attributi degli oggetti, come illustrato nella figura 1.

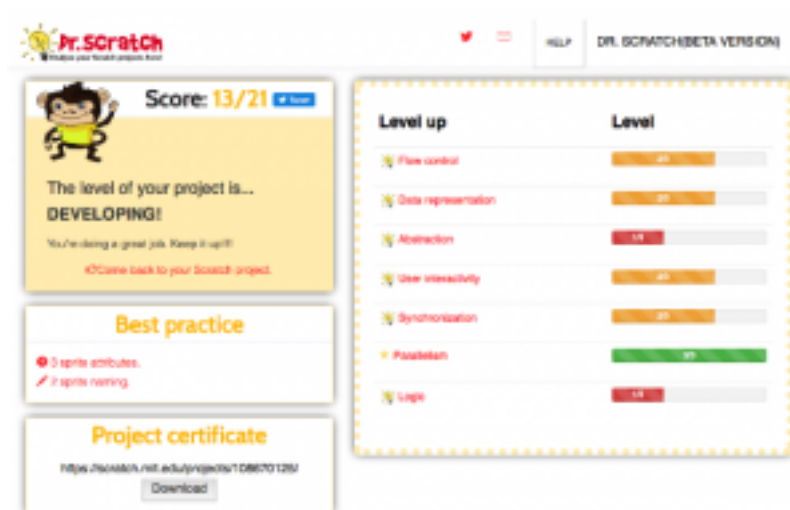


Figura 1 – rapporto di feedback di Dr. Scratch che assegna un punteggio al pensiero computazionale e rileva alcune cattive abitudini di programmazione.

Per ciascuna delle cattive abitudini di programmazione rilevate nel codice e per ciascuna delle dimensioni del pensiero computazionale in cui vi è spazio per miglioramenti, lo strumento fornisce collegamenti a informazioni che possono essere utilizzate per migliorare i progetti.

Ad esempio, se un progetto include un codice ripetuto, Dr. Scratch fornisce un collegamento ad un esempio di codice sorgente, una spiegazione del perché questa situazione dovrebbe essere evitata e come gli utenti potrebbero migliorare il loro progetto sviluppando loro stessi i propri blocchi (vedi figura 2).

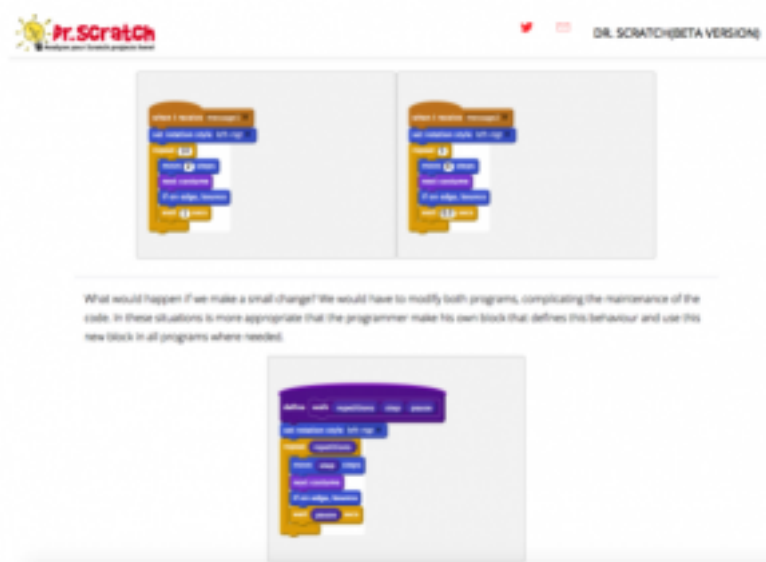


Figura 2 – Idee e suggerimenti forniti da Dr. Scratch per evitare la ripetizione del codice mediante la creazione di nuovi blocchi

Pensato per non scoraggiare i programmatori alle prime armi, il rapporto di feedback fornito da Dr. Scratch dipende dal punteggio assegnato al pensiero computazionale. In questo modo, se il punteggio è basso, lo strumento fornirà solo informazioni di base

relative ai miglioramenti più importanti da apportare al codice, ma appena il punteggio aumenta, Dr. Scratch fornirà maggiori informazioni circa i progetti analizzati.

Al fine di valutare l'efficacia del Dr. Scratch come strumento per aiutare i discenti nella programmazione, abbiamo organizzato una serie di workshop con 109 studenti tra i 10 e i 14 anni provenienti da 8 diverse scuole spagnole e che hanno già avuto esperienza di codifica con Scratch. Gli studenti hanno analizzato uno dei loro progetti Scratch con Dr. Scratch, hanno letto le informazioni contenute nel rapporto di feedback fornito dallo strumento, hanno cercato di migliorare il loro codice seguendo le istruzioni date e, infine, hanno analizzato di nuovo i loro progetti.

I risultati mostrano che, in media, gli studenti hanno migliorato il loro CT Score di 1,45 punti, da 12.00 / 21 al 13.45 / 21, il che rappresenta un miglioramento statisticamente significativo. La dimensione complessiva dell'effetto, $d = 0,47$, indica un effetto piuttosto moderato ma, tenendo conto che è stato generato durante una sola ora di workshop, esso evidenzia il reale impatto che l'uso di Dr. Scratch ha avuto sulle capacità di codifica e di sviluppo del Pensiero Computazionale (CT) dei partecipanti.

I risultati indicano che il feedback fornito da Dr. Scratch è stato particolarmente utile per quegli studenti della scuola secondaria con punteggio CT iniziale medio (developing). Tuttavia, lo strumento non sembra essere utile per studenti con un alto punteggio iniziale (proficient), per lo meno in un laboratorio di una sola ora. Dedicheremo il nostro lavoro futuro a verificare come migliorare il feedback fornito dallo strumento. Inoltre, una nuova ricerca potrebbe aiutarci a scoprire se le differenze di prestazioni tra gli studenti delle scuole secondarie e primarie sono dovute allo strumento stesso o sono invece legate alla maturazione dello sviluppo meta-cognitivo dei discenti.

Dr. Scratch è utilizzato da insegnanti e organizzazioni di tutto il mondo come strumento di supporto in attività di valutazione. Ad esempio, lo strumento può essere utilizzato per rilevare gli studenti che non utilizzano determinate istruzioni nei loro progetti, come ad esempio le operazioni logiche o cloni, in questo modo gli insegnanti potrebbero preparare compiti specifici per aiutarli a capire l'importanza di tali operazioni.

"Fondamentalmente, mi piace molto la possibilità di dare agli studenti una prospettiva totalmente soggettiva (ed esterna al loro insegnante) sul loro coding e dove questo funziona e dove no. Apprezzo anche la riduzione del mio tempo per la correzione e l'assegnazione dei voti, e anche per la ricerca di eventuali errori nel loro codice se non sta facendo quello che dovrebbe.

Di solito ho poi ancora bisogno di verificare e valutare se i criteri che ho fissati sono stati rispettati, quindi il dottor Scratch è sicuramente di grandissimo aiuto! "

Paul Herring, St Peters Lutheran College, Australia.

"Gli studenti spesso credono che se il loro progetto funziona ", questo non possa andare meglio. Io uso Dr. Scratch per mostrare quelle debolezze del codice che può essere migliorato evidenziando tutti quei concetti di informatica che gli studenti stanno imparando attivamente, ma che sono spesso sottaciuti. Dr.

Scratch aiuta i miei studenti a migliorare le loro capacità di codifica in un modo che può essere trasferito ai linguaggi di programmazione professionali. Mi capita spesso di avere un feedback da miei studenti che sostengono che quando usano Dr. Scratch il loro lavoro sembra essere più rilevante per il corso di informatica e che migliora la loro percezione di realizzazione. "

Dimitris Nikolos, Model Experimental Junior High School of University of Patras, Greece.

Vi sono alcune lacune che abbiamo in programma di affrontare nel prossimo futuro. Infatti l'analisi di un singolo progetto di uno studente non fornisce un quadro completo del suo sviluppo di pensiero computazionale. La ragione è che ci sono meravigliosi semplici progetti che non richiedono di essere modificati giusto per includere strutture più complesse che sono quelle che danno un alto punteggio di pensiero computazionale.

Nel prossimo futuro il team di sviluppo di Dr. Scratch includerà una nuova funzionalità per consentire la creazione di account utenti. Di conseguenza, l'analisi del portfolio dei progetti degli utenti fornirà un quadro più ricco, e i punteggi aggregati consentiranno di rilevare le lacune di conoscenza in modo più accurato.

Vale la pena ricordare che, anche se il dottor Scratch sta supportando con successo gli insegnanti nei compiti di valutazione, lo strumento non deve essere inteso come un sostituto di valutatori o mentor, in quanto vi sono importanti capacità di pensiero computazionale, come ad esempio il debugging, che non vengono valutate. Inoltre, la funzionalità, l'originalità o la creatività, aspetti chiave di progetti di programmazione, non sono neanche presi in considerazione nel punteggio fornito da Dr. Scratch.

Riconoscimenti

Il lavoro di entrambi gli autori è stato in parte finanziato dalla Regione di Madrid nell'ambito del progetto "eMadrid – Investigación y Desarrollo de tecnologías para el-learning en la Comunidad de Madrid" (S2013 / ICE-2715). Il lavoro di Gregorio Robles è stato in parte finanziato dal governo spagnolo nell'ambito del progetto SobreSale (TIN2011- 28110). Siamo anche molto grati a Mari Luz Aguado e Eva Hu per il loro supporto tecnico.

Per saperne di più su Dr. Scratch

Moreno, J., & Robles, G. (2014). Automatic detection of bad programming habits in scratch: A preliminary study. *Proceedings of the Frontiers in Education Conference (FIE), 2014 IEEE* (pp. 1-4). IEEE.

Moreno-León, J., Robles, G., & Román-González, M. (2015). Dr. Scratch: Automatic analysis of Scratch projects to assess and foster computational thinking. *Revista de Educación a Distancia*, (46).

Moreno-León, J., & Robles, G. (2015). Dr. Scratch: a Web Tool to Automatically Evaluate Scratch Projects. In *Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 132-133). ACM.