



Citation: M. Giampaolo, C. Garofano (2021) Il coding nei percorsi di tirocinio degli studenti di Scienze dell'educazione e della formazione. Uno studio di caso. *Media Education* 12(2): 73-82. doi: 10.36253/me-11366

Received: June, 2021

Accepted: November, 2021

Published: December, 2021

Copyright: © 2021 M. Giampaolo, C. Garofano. This is an open access, peer-reviewed article published by Firenze University Press (<http://www.fupress.com/me>) and distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All relevant data are within the paper and its Supporting Information files.

Competing Interests: The Author(s) declare(s) no conflict of interest.

Best Practices

Il coding nei percorsi di tirocinio degli studenti di Scienze dell'educazione e della formazione. Uno studio di caso

Coding in curricular internship paths of students of education and training sciences. A case study

MARIO GIAMPAOLO, CATERINA GAROFANO¹

Università degli studi di Siena

mario.giampaolo@unisi.it; caterina.garofano@student.unisi.it

Abstract. The rapid changes that characterize the world of work ask school and university to critically rethink the way to facilitate creative skills. The coding and the development of computational thinking are declined in this context as facilitators of these skills in everyday life and in working contexts. The contribution presents a case study that describes peer tutoring laboratory activities focused on learning coding skills that involved students, enrolled in the Degree Course in Education and Training Sciences of the University of Siena, as a form of alternative curricular internship during the lockdown period. The study shows how students approached this form of internship, how they designed peer tutoring sessions and the gains in terms of learning and professional development.

Keywords: coding, creative learning, internship, professional development.

Riassunto. I veloci cambiamenti che caratterizzano il mondo del lavoro ci chiedono di ripensare criticamente alle nostre professionalità, mettendo in campo doti creative. L'attività di *coding* e lo sviluppo del pensiero computazionale si declinano in questo contesto come facilitatori dell'implementazione del pensiero creativo nella vita di tutti i giorni, allenando la capacità di progettare sistemi che permettono la risoluzione dei problemi. Il contributo presenta uno studio di caso che descrive attività laboratoriali di peer tutoring incentrate sull'apprendimento delle abilità di coding che hanno coinvolto studenti e studentesse, iscritte al Corso di Laurea in Scienze dell'educazione e della formazione dell'Università degli studi di Siena, come forma di tirocinio alternativo che ha concesso loro di svolgere l'attività formativa prevista nel piano di studi durante il periodo di lockdown. Lo studio mostra come studenti e studentesse si sono approcciati a questa forma di tirocinio, come hanno progettato le sessioni di peer tutoring e i guadagni in termini di apprendimento e sviluppo professionale.

Parole chiave: coding; apprendimento creativo; tirocinio; sviluppo professionale.

¹ Il lavoro è frutto di lavoro congiunto tra gli autori. Solo per ragioni di attribuzione scientifica si specifica che Caterina Garofano è autrice dei paragrafi 1 e 2, Mario Giampaolo è autore dei paragrafi 3, 4 e 5.

1. INTRODUZIONE

Le professioni e i contesti lavorativi sono oggi soggetti a forti cambiamenti (Butera, 2020). Diversi ruoli in altrettanti settori produttivi stanno scomparendo velocemente sostituiti da macchine che riescono a svolgere operazioni routinarie e non solo. Se queste superano l'uomo in velocità di esecuzione, in accuratezza e affidabilità, non riescono ancora ad eguagliarlo in termini di creatività. Qualunque sia il lavoro da svolgere, proporre e prendere decisioni creative rende questo più interessante, efficiente e produttivo. Il successo di individui, organizzazioni e comunità sulle macchine dipende da questa capacità (Schmidt et al., 2016). La creatività sarà ancora più necessaria per fronteggiare le sfide dell'era post Covid-19.

I sistemi educativi, però, non supportano adeguatamente lo sviluppo del pensiero creativo nei professionisti di domani (Robinson, 2010; Robinson & Aronika, 2016). Originariamente progettate per fornire lavoratori alle fabbriche dell'età industriale, scuola e università hanno trovato difficoltà ad adattarsi all'evolversi della società. Il sistema educa e forma ad eseguire istruzioni come ascoltare, memorizzare, rispondere, compilare l'esercizio, istruzioni che si traducono nei contesti lavorativi in processi lavorativi stile catena di montaggio. Oggi le catene di montaggio sono sempre più presidiate da macchine. Vi è una scarsa attenzione allo sviluppo di un pensiero creativo, ovvero quell'atteggiamento mentale che permette di prevedere i problemi, anticipare la loro occorrenza, essere consapevoli che la soluzione proposta è solo una delle possibili. È questo pensiero che permetterà di svolgere quei lavori che supervisioneranno il lavoro delle macchine e che ne saranno potenziati e non rimpiazzati da queste. Ad esempio, lavori che richiedono adattabilità, relazioni personali, capacità di applicare creativamente abilità motorie fini, capacità di utilizzare in maniera creativa le tecnologie e le informazioni e di contribuire al processo di design delle macchine che mediano la relazione tra utenti e servizi, richiedono lo sviluppo di abilità e conoscenze specifiche ma anche la creatività nell'utilizzo della tecnologia e nel lavorare insieme alle macchine (Schmidt et al., 2016).

I problemi fin qui evidenziati hanno spinto molti studiosi a interrogarsi su possibili soluzioni da adottare. Mitchel Resnick e i ricercatori del gruppo *Lifelong Kindergarten* del Media Lab del MIT di Boston, ad esempio, sostengono che i livelli più alti del sistema formativo dovrebbero assomigliare molto di più ai nidi e alle scuole dell'infanzia (Lodi, 2018; Papert, 1980; Resnick, 2007; 2014). Qui avvengono eventi di apprendimento spontaneo, dal basso, sebbene con il supporto di educatori e

insegnanti. Negli asili i bambini, sotto la guida dell'educatore, progettano, creano, sperimentano ed esplorano. Emblematico è uno dei casi che Resnick (2018) riporta nel suo testo:

Due bambini iniziano a giocare con dei blocchi di legno e costruiscono una serie di torri. Un compagno di classe viste le torri muove una macchinina per farla passare attraverso due di queste. Le torri sono troppo vicine tra loro e i bambini iniziano a muovere le torri per distanziarle e far posto alla macchina. Una delle due torri cade. Dopo un breve litigio su chi avesse la colpa, iniziano a parlare di costruire una torre più alta e più forte. L'educatore mostra loro immagini di grattacieli del mondo reale, e i bambini si accorgono che i primi piani degli edifici sono più larghi degli ultimi. Così decidono di ricostruire le loro torri di blocchi con una base più ampia.

Questo tipo di processo viene ripetuto più e più volte all'asilo. I materiali variano (pittura a dita, pastelli, ecc...) e le creazioni variano (immagini, storie, canzoni), ma il processo principale è lo stesso. Un processo in cui i bambini immaginano cosa vogliono fare, creano un progetto basato sulle loro idee, giocano con le loro creazioni, condividono le proprie idee e creazioni con gli altri, riflettono sulle loro esperienze - tutto ciò li porta a immaginare nuove idee e nuovi progetti. (Resnick, 2018)

La tesi di Resnick (2018), secondo cui la scuola e l'università dovrebbero trarre ispirazione dall'approccio all'apprendimento che caratterizza servizi e scuole dell'infanzia, non trova applicazione. Il pensiero creativo non è sufficientemente considerato come abilità fondamentale da facilitare; a questo sono preferite conoscenze e competenze tecniche e, inoltre, nel progredire dei livelli di istruzione, le risorse a disposizione sono spesso insufficienti per creare *setting* in cui facilitare processi creativi. Nei servizi e nelle scuole dell'infanzia blocchi di legno e pastelli colorati sono strumenti utili per creare contesti dove esercitare capacità di astrazione che alimentano il pensiero creativo; nei livelli successivi del sistema d'istruzione progetti d'apprendimento avanzati, potrebbero essere svolti in ambienti e laboratori virtuali creati mediante stringhe di codice in cui l'applicazione di un pensiero creativo potrebbe condurre a risultati inaspettati.

2. COMPONENTI E FASI PER UN APPRENDIMENTO CREATIVO: L'APPROCCIO DEL LIFELONG KINDERGARTEN GROUP DEL MIT MEDIA LAB

Nel seguente paragrafo sarà descritto un approccio all'apprendimento funzionale allo sviluppo della creatività (Resnick, 2014; 2018). L'approccio propone quattro componenti e cinque fasi che si succedono a spirale. Le

componenti sono: i progetti, le passioni, i pari e il gioco; le fasi consistono nell'immaginazione, creazione, gioco, condivisione e riflessione.

Resnick sostiene che imparare lavorando per progetti permette alle persone di generare idee, prototipi e prodotti. Avere un progetto in mente predispone ad un atteggiamento di ricerca delle informazioni su come fare qualcosa, agire, magari sbagliare e ripetere le azioni. Riflettendo sul processo di progettazione risolviamo problemi specifici e affiniamo le capacità di produrre soluzioni generalizzabili. I progetti più ambiziosi e i problemi più difficili non possono essere risolti da soli, per questo nella vita quotidiana e in quella professionale collaboriamo. Le capacità di coinvolgere gli altri nella nostra vita e nel nostro lavoro e collaborare con loro in modo costruttivo sono importanti. Coinvolgere, condividere e collaborare comporta spiegare agli altri ciò che pensiamo e quando spieghiamo le nostre stesse comprensioni su un argomento ne beneficiano.

I nostri progetti, quelli cui sentiamo il bisogno di realizzare, ci appassionano, vi lavoriamo di più a più a lungo, più duramente. Aiutare bambine, bambini, ragazze e ragazzi ad identificare le loro passioni e supportarli affinché le perseguano è la sfida educativa di insegnanti ed educatori. Infine, sperimentare situazioni e pratiche in contesti protetti, come all'interno di un gioco, consente di testare idee nuove, utilizzare diversi strumenti e materiali rispetto a quelli già conosciuti, assumersi rischi e ripetere tutto se si fallisce.

Queste componenti caratterizzano le diverse fasi della spirale creativa. L'immaginazione, la prima, è il punto da cui prendono vita le storie di personaggi inventati o idee per disegnare o costruire. Dall'immaginazione si passa all'azione, disegnando o costruendo bambine e bambini aggiungono dettagli ed elementi grazie ai

materiali che stanno utilizzando, pennarelli, mattoncini ed altri materiali e quando fanno questo insieme discutono su cosa aggiungere. Giocano e condividono, altre fasi della spirale, ad un vero e proprio progetto. La loro collaborazione li porta a riflettere su come fare le cose ancora meglio: una costruzione più alta e più stabile, un disegno con colori diversi. Riflettono quindi su ciò che hanno fatto e su come possono continuare a farlo immaginando nuovi sviluppi.

Resnick afferma che il punto di partenza della spirale, la possibilità di immaginare è connessa ai materiali che vengono utilizzati. Con i materiali di oggi, le tecnologie digitali, l'autore ritiene che sia possibile creare strumenti e spazi per soddisfare approcci all'apprendimento che lavorano su progetti e idee più avanzate. Scratch, ad esempio, così come altri ambienti e linguaggi di programmazione permettono di creare laboratori virtuali in cui i professionisti di domani possono sperimentare, esplorare e verificare l'esito delle loro azioni. Anche i videogiochi possono essere considerati uno strumento utile per apprendere poiché permettono di simulare contesti reali in cui sviluppare conoscenze e competenze. Spinti dalla straordinaria popolarità dei videogiochi nella cultura giovanile, un numero crescente di ricercatori (Lacasa, 2013; Prensky, 2001) ha iniziato a esaminare come e cosa impara chi videogioca. Si sviluppa un profondo senso di coinvolgimento, un pensiero strategico, abilità nella risoluzione dei problemi e forniscono opportunità per progettare e creare.

La condivisione è un'altra fase cruciale nella spirale della creatività, ma è stata meno enfatizzata dalla scuola e dall'università rispetto alla competizione. Ciò è cambiato da quando studiosi nel campo dell'educazione e della formazione hanno concentrato l'attenzione sulla natura sociale dell'apprendimento e sulle strategie per sostenere le comunità (Lave, 1996; Lave & Wenger, 2002; Resnick, 1991). Assistiamo, infine alla proliferazione di tecnologie interattive e a un ampio accesso alla rete che ha diffuso la cultura della partecipazione (Fabbri, 2019; Lave & Wenger, 2002). Le scuole e le università hanno adottato, negli ultimi anni, approcci più attivi, ma il focus è spesso spostato sulla creazione di un artefatto piuttosto che sulla riflessione critica delle idee che hanno guidato la progettazione, sulle possibili strategie per affinare e migliorare il design, o sui collegamenti a concetti scientifici correlati ai fenomeni del mondo reale. L'iterazione di queste fasi caratterizza la spirale. Si immagina, si crea, si condivide e si riflette. Poi inevitabilmente appaiono nuove idee, che riconducono a nuove immaginazioni e all'inizio di un nuovo ciclo.

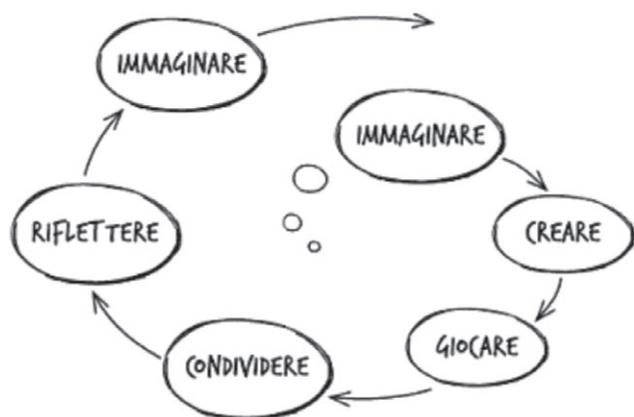


Figura 1. Un approccio all'apprendimento per l'asilo nido da Resnick, 2018.

3. AMBIENTI PER LO SVILUPPO DELLE ABILITÀ DI PROGRAMMAZIONE E DEL PENSIERO COMPUTAZIONALE

Le componenti e le fasi descritte nel precedente paragrafo rivestono un ruolo minore nella scuola e nell'università a causa della mancanza di risorse economiche e materiali che non permettono la sperimentazione di concetti complessi e astratti caratterizzanti i gradi più alti di istruzione. Oggi, però, linguaggi di programmazione *user friendly* permettono di realizzare laboratori digitali dove progettare, immaginare, condividere e sperimentare le leggi della fisica così come la scrittura di una storia, riportando la creatività al centro dell'educazione e della formazione. Secondo Resnick, apprendere a programmare non significa solo acquisire *skills* tecniche ma creare attraverso la tecnologia. Gli strumenti di programmazione rendono possibili ambienti di apprendimento costruzionisti che sostituiranno quelli formali dove vige ancora il paradigma trasmissivo.

Il coding ovvero l'atto di programmare le macchine è un esempio di come la teoria costruzionista dell'apprendimento può essere applicata nella realtà (Papert, 1980; 1991) ma molteplici sono i collegamenti che potrebbero essere fatti con altre teorie dell'apprendimento, prima tra tutte la teoria costruttivista di Piaget (1976; 2016). Non solo, le attività di programmazione permettono la definizione di un contesto in cui si affronta un problema reale, in cui si realizzano molteplici azioni risolutive (fase esecutiva), si modella un problema e si scompone in sottoproblemi (fase iconica), si esegue il processo creato mediante simboli e convenzioni condivise (fase simbolica) (Bruner, 1966). Inoltre, acquisire abilità di programmazione mediante una didattica laboratoriale permette a chi apprende di essere supportato da docenti o pari esperti e di aumentare le proprie conoscenze gradualmente, raggiungere nuovi traguardi e fare esperienza. In questo modo viene valorizzata la zona di sviluppo prossimale degli studenti (Vygotskij, 1997).

L'interesse multidisciplinare per la programmazione e l'aumento della presenza delle macchine nelle attività quotidiane stanno concentrando l'interesse dei sistemi educativi formali. Il valore aggiunto risiede nella possibilità di avere sistemi e ambienti virtuali in cui simulare problemi complessi. L'interazione in e con questi sistemi e ambienti virtuali mediante il codice di programmazione costringe a spaccettare un problema nelle sue componenti. Dovendo risolvere un problema particolare, potremmo chiederci: quanto è difficile risolverlo? E qual è il modo migliore per farlo? Potremmo inoltre chiederci se una soluzione approssimativa è abbastanza buona, quali sono i facilitatori e i distrattori del nostro ragiona-

mento. A queste domande è possibile rispondere facendo eseguire alle macchine delle istruzioni. Il pensiero computazionale, che sviluppiamo scrivendo le istruzioni per dialogare con le macchine, permette di riformulare un problema apparentemente difficile nelle sue parti, di utilizzare l'astrazione e la decomposizione per affrontare un compito complesso o per progettare un sistema volto alla soluzione di un problema. Utilizza la rappresentazione per modellare gli aspetti rilevanti di un problema e per renderlo trattabile. È pianificazione, apprendimento e programmazione in presenza di incertezze (Wing, 2006; 2017).

4. I LABORATORI DI CODING COME PERCORSI ALTERNATIVI DI TIROCINIO.

Lo studio di caso descritto di seguito riguarda la realizzazione di laboratori di peer-tutoring per l'apprendimento delle abilità di *coding*. Questi laboratori sono stati proposti come attività alternativa allo svolgimento del tirocinio curricolare nel Corso di Laurea in Scienze dell'educazione e della formazione (L-19) dell'Università di Siena durante il lockdown del marzo 2020. L'interesse era quello di superare le forme di tirocinio indiretto che spesso si riducono all'analisi e alla rielaborazione di testi e muovere verso metodologie attive e partecipative per lo sviluppo professionale (Fabbri, 2003; 2012; 2014; 2017; Fabbri & Giampaolo, 2019; Fabbri et al., 2019; Fabbri et al., 2017; Fabbri & Melacarne, 2016; Fabbri & Rossi, 2008). Le difficoltà incontrate dalle studentesse e dagli studenti nel trovare disponibilità da parte di organizzazioni come asili nido, centri di accoglienza, case-famiglia, agenzie formative e altri luoghi tradizionalmente scelti, hanno spinto la governance del Corso a ipotizzare un percorso alternativo (Fig. 2) che permettesse di sviluppare abilità di pianificazione, soluzione dei problemi, ricerca attiva delle informazioni per svolgere compiti, comunicazione e condivisione, creatività e produzione di artefatti. Questo percorso alternativo ha preso la forma di un laboratorio tra pari dove lo studente più esperto facilita l'apprendimento dei propri compagni su vari argomenti. Tra gli argomenti proposti quello che ha riscontrato l'interesse maggiore è stato il *coding*.

La procedura coinvolge gli studenti in una prima fase di preparazione. Gli studenti, supportati dal tutor universitario, hanno la possibilità di approfondire le tematiche del pensiero computazionale e della programmazione applicata ai contesti educativi. Le risorse in questa fase sono suggerite dal tutor universitario o ricercate autonomamente dallo studente. Esempi di risorse utilizzate dagli studenti sono i corsi sulla programmazione in

ambito educativo dei *massive on-line open courseware* Edx o Coursera e articoli scientifici reperiti sul database del sistema bibliotecario d'ateneo o in rete. Una seconda attività da svolgere in questa fase è la familiarizzazione con il *software* di programmazione per blocchi Scratch.

Scratch è sia un linguaggio di programmazione che una comunità online dove i ragazzi possono programmare e condividere con altri provenienti da qualunque parte del mondo i loro oggetti multimediali interattivi come ad esempio storie, giochi e animazioni. Quando i ragazzi creano con Scratch imparano a pensare in modo creativo, a lavorare in maniera collaborativa e a ragionare in modo sistematico. Scratch è progettato e supportato dal Lifelong Kindergarten group dei Media Lab del MIT (<https://scratch.mit.edu/>).

Infine, altri due importanti passaggi in questa fase sono la progettazione e la produzione dei materiali necessari allo svolgimento della sessione formativa. Il tutor universitario ha, durante questa fase, un ruolo di supervisione del tirocinante in termini di progettazione dell'attività formativa e di produzione dei materiali d'ausilio. Le tempistiche di questa fase stabiliscono un monte orario di 75 ore per il suo completamento.

Nella seconda fase, quella di realizzazione, il tirocinante deve organizzare incontri con i propri compagni di Corso mediante la piattaforma Google Meet. In questi incontri la scheda di progettazione (appendice 1) e i materiali prodotti permettono al tirocinante di facilitare l'apprendimento dei concetti studiati e delle abilità di

coding. Il monitoraggio e la valutazione della sessione formativa non sono in questa fase prerogative del tutor universitario ma degli studenti esperti che hanno un immediato riscontro del livello di interazione, di interesse e di soddisfazione suscitato nei pari.

Di seguito riportiamo un laboratorio condotto da una tirocinante. L'esperienza è descritta grazie all'analisi del suo elaborato di tesi triennale dove ne sono riportate nel dettaglio le caratteristiche. In particolare, ci concentreremo sulla fase di realizzazione e sull'incontro tutor-tutee.

Il laboratorio "Informatica creativa per gli educatori" ha una durata di circa 120 minuti e comprende la spiegazione di alcuni concetti relativi al *coding*, al software Scratch e la visione di tutorial che spiegano il modo con cui realizzare un progetto su Scratch. Terminata la parte teorica si procede con 60 minuti di pratica in cui ciascun partecipante realizza e poi presenta il suo progetto (Fig. 3, 4 e 5).

È mio interesse spiegare cos'è il pensiero computazionale. Vorrei provare a spiegare il pensiero di Seymour Papert, ovvero l'idea di un apprendimento vicino alla realtà, che propone un pensiero concreto, che sprona chi apprende a pensare in modo creativo, sistematico e collaborativo.

Ogni partecipante alla fine del modulo conoscerà l'interfaccia di Scratch e creerà un'animazione usando gli effetti grafici, i blocchi causalità e i blocchi variabili. Per valutare il progetto realizzato dai partecipanti la tirocinante ha stabilito e condiviso i seguenti criteri:

Nel primo tutorial vengono spiegati i comandi base, l'impostazione degli script e dello stage. Per prima cosa la tirocinante conduce i partecipanti su un motore di ricerca e chiede di digitare l'indirizzo *scratch.mit.edu*. Una volta caricata la home page del software viene impostare la lingua italiana. Inizia quindi la presentazione dell'interfaccia:

...indico attraverso il cursore a quale zona dello schermo corrisponde lo stage, ovvero il luogo in cui si svolgerà l'animazione. Procedo con la scelta dello stage: disegno uno

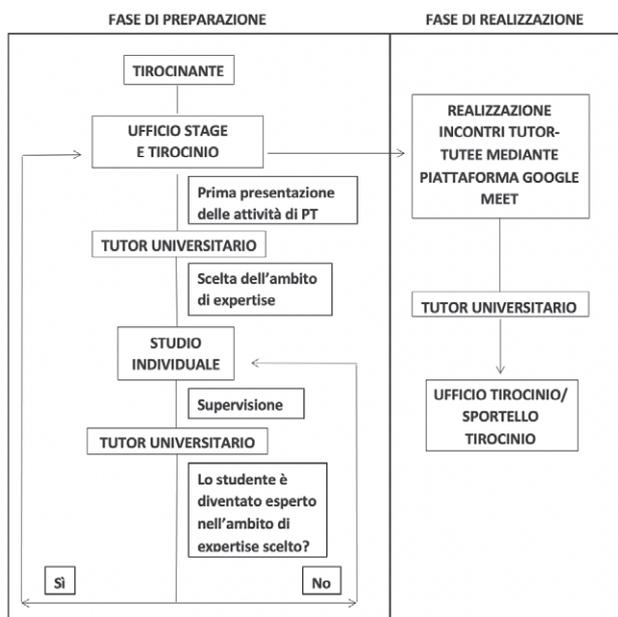


Figura 2. Diagramma di flusso della procedura alternativa per lo svolgimento delle attività di tirocinio.

Tabella 1. Check-list di valutazione degli artefatti prodotti dai partecipanti.

| CRITERI DI VALUTAZIONE | SI | NO |
|---|----|----|
| Deve contenere almeno una variabile | | |
| Deve durare almeno 3 minuti | | |
| Deve avere un senso logico | | |
| Deve esserci un clone | | |
| Lo Sprite deve muoversi da destra verso sinistra e dal basso verso l'altro. | | |

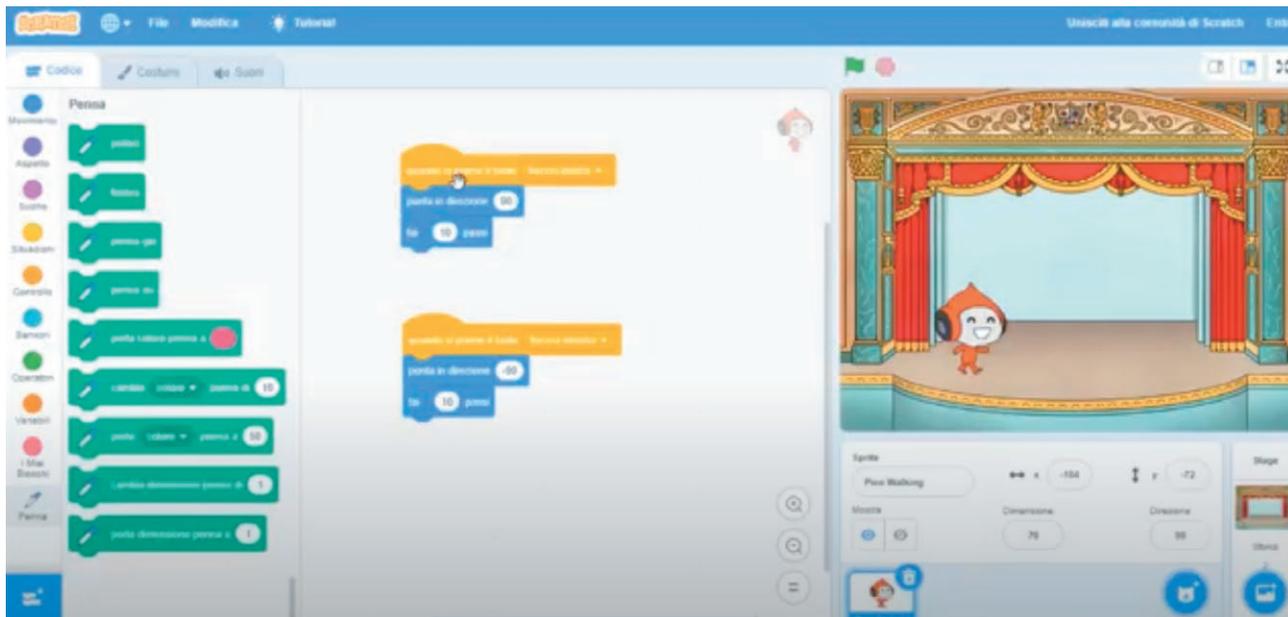


Figura 3. Un esempio di progetto realizzato con Scratch.

sfondo verde che rappresenta un prato, successivamente scelgo lo sprite (il personaggio da animare). Lo seleziono dalla libreria che mi fornisce il software e scelgo la rana. Segue un'introduzione della mia idea e di come voglio realizzarla: spiego che il mio sprite dovrà mangiare i topi di dimensioni più piccole di lui, e man mano che la sua dimensione aumenterà potrà mangiare i topi più grossi. Il mio sprite dovrà di conseguenza evitare i topi più grandi di lui, altrimenti il gioco finirà.

Nel secondo video la tirocinante introduce gli sprite e i blocchi movimento:

...spiego come impostare la dimensione dello Sprite. Inizio ad utilizzare situazioni e trascino nell'area di programmazione il blocco "quando si clicca su bandierina verde". Questo blocco è fondamentale per far partire il gioco, funge da classico start, se non lo impostiamo i comandi non funzionano. Dopodiché creo una variabile movimento: il mio sprite deve potersi muovere nello stage da sinistra verso destra. Infine, imposto la variabile dimensione, per fare in modo che ogni qualvolta che la rana mangia un topo più piccolo il suo corpo aumenta di dimensione.

Nel terzo tutorial l'obiettivo è far muovere lo Sprite con le frecce della tastiera.

...prima imposto i blocchi "freccia destra" e "freccia sinistra", poi duplico il blocco e imposto "freccia giù" e "freccia su". L'impostazione permette di fare un accenno agli assi cartesiani. Spiego come utilizzare i blocchi di comando e come impostare le coordinate sugli assi x e y.

Nel quarto tutorial la tirocinante crea topi piccoli, che la rana dovrà mangiare per crescere, e topi grandi, che la rana dovrà evitare altrimenti il gioco terminerà. I topi devono comparire in ordine casuale, essere di dimensioni diverse e dovranno spostarsi da destra verso sinistra.

...creo lo sprite numero 2 (il topo) e inizio di nuovo con il blocco "quando si clicca su bandierina verde". Per prima cosa devo modificare l'aspetto dello Sprite in questione e poi imposto il comando per generare i suoi cloni. Tra la comparsa di un clone e l'altro definisco un tempo variabile da 1 a 3 secondi. Adesso creo il clone e ne modifico la dimensione. Ciò che voglio è che i topi abbiano una dimensione minore o maggiore rispetto alla rana. Ovviamente anche i topi devono avere un movimento che va da sinistra verso destra. Molto semplice: rivediamo i comandi che abbiamo usato precedentemente, impostando le coordinate. Ma come faccio a muovere i topi? Con un ciclo infinito e impostando un movimento che gli faccia fare un passo alla volta.

Secondo obiettivo di questo tutorial è far scomparire il topo quando tocca il bordo a destra. Questa situazione è un pochino più complicata ma vediamo insieme: se la posizione di x è maggiore di 240...

Infine, l'ultimo tutorial imposta le regole del gioco.

Cosa ci manca da fare? Definire le regole del gioco. Dalla sezione controllo imposto la regola se la dimensione del topo è più grande della rana allora game over altrimenti

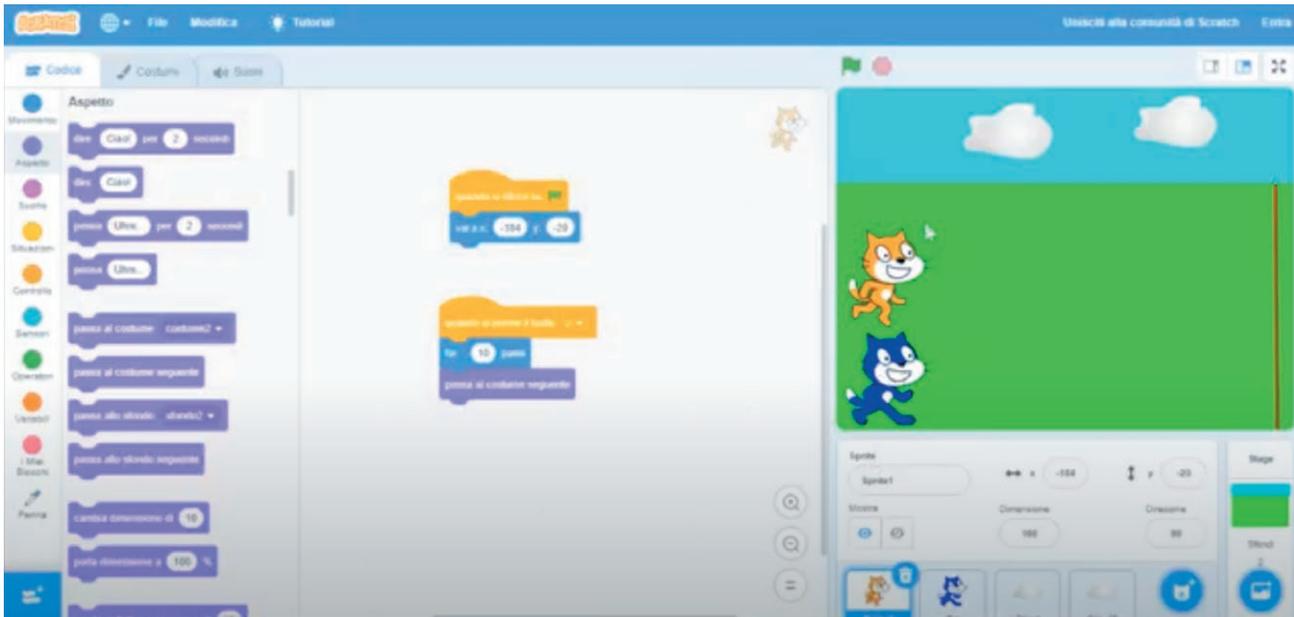


Figura 4. Un esempio di progetto realizzato con Scratch.

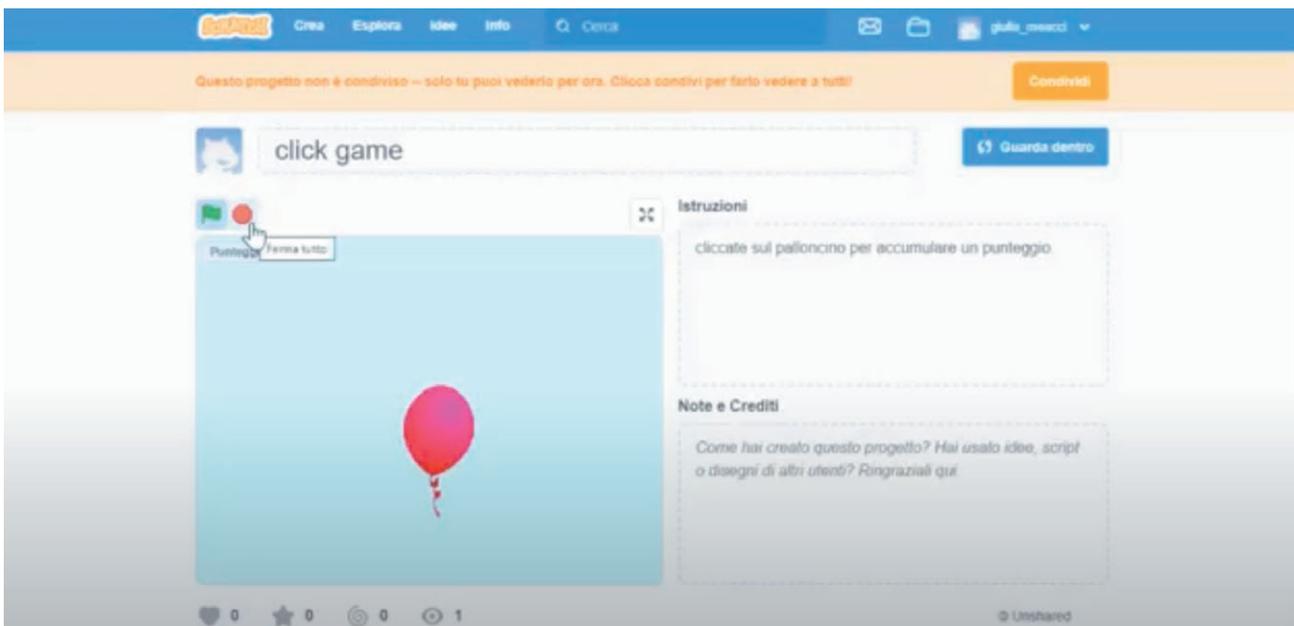


Figura 5. Un esempio di progetto realizzato con Scratch.

ti aumento il valore di size (la variabile che ho impostato per la dimensione della mia rana) per far crescere la rana. Devo ricordarmi che ogni volta il clone del topo si elimina. Per concludere disegno lo stage con la scritta game over, ne modifico il carattere e il font. Quindi anche per esso clicco su "se clicchi su bandierina verde". Quando vedo la scritta game over il gioco termina.

Il laboratorio descritto precedentemente, si basa su un lavoro di progettazione che ha riguardato la definizione degli obiettivi d'apprendimento, dei risultati di apprendimento e dei criteri di valutazione del prodotto realizzato dai partecipanti al laboratorio. Il laboratorio è stato progettato indicando le tempistiche, i contenuti e il materiale da utilizzare come supporto alla spiegazione e

alla realizzazione del progetto su Scratch. Le attività di valutazione non sono mirate a riconoscere l'acquisizione di conoscenze teoriche ma di abilità per la produzione di un artefatto digitale, come ad esempio un'animazione che fa riferimento a dei criteri specifici che definiscono la qualità di ciò che si è fatto. Il tirocinio, quindi, è vissuto come un processo che coinvolge il tirocinante nell'immaginare e progettare un'attività formativa tra pari, creare e utilizzare artefatti, i progetti su Scratch, che saranno poi condivisi, riflettere sulla progettazione svolta, sugli errori commessi e sulla capacità di supportare l'apprendimento di altre persone. La tirocinante nelle conclusioni del suo elaborato ritiene di aver maturato una conoscenza teorica e pratica relativa al *coding* ma soprattutto la consapevolezza di quanto e come esso possa essere implementata nella progettazione di azioni educative innovative.

5. RIFLESSIONI CONCLUSIVE

Il contributo, partendo da una riflessione sull'importanza della creatività in ambito professionale ha descritto uno studio di caso sulla realizzazione di attività di peer tutoring tra studenti pensate come percorso alternativo al tirocinio durante il periodo del lockdown. Queste attività si sono concretizzate in laboratori di familiarizzazione al coding e al pensiero computazionale. Gruppi di studenti più e meno esperti hanno avuto l'opportunità di partecipare a palestre in cui muovere i primi passi nella pratica di programmazione. Attività come questa potranno essere sostenibili ed efficaci come forme di tirocinio curriculare anche nell'era post-Covid? Quali effetti hanno avuto questi laboratori sullo sviluppo professionale dei futuri educatori? Per rispondere a queste domande possiamo mettere in risalto alcuni limiti riscontrati. Il tirocinante e i partecipanti al laboratorio hanno lavorato riproducendo storie animate e videogiochi come proposti dal tutor, sicuramente maggiore spazio dovrà essere lasciato a idee e progetti pensati autonomamente dai tutee. Ciò li aiuterebbe a improvvisare, ad adattare le proprie scelte e a ripetere le loro azioni quando non portano ai risultati desiderati. Aumentare il numero di tutee coinvolti nei laboratori amplierebbe le possibilità di collaborazione. Lavorando insieme, spiegando in cosa riescono e in cosa no, condividerebbero le loro opinioni e le loro strategie su come si fa qualcosa. Un ulteriore elemento di sviluppo potrebbe essere quello di comprendere come è mutata la loro consapevolezza circa i linguaggi di programmazione: Hanno conosciuto una nuova attività prima pensata come specialistica e complessa? Questa consapevolezza acquisita avrà un

impatto su altre attività? Numerose sono le domande che potrebbero essere poste per comprendere l'impatto di questi laboratori: i partecipanti hanno imparato dai loro sbagli? Hanno utilizzato i blocchi di programmazione in modi diversi da quelli pensati dal tutor? Nelle attività realizzate i tutee hanno condiviso informazioni o gareggiato per dare una risposta o per terminare un progetto? Hanno riflettuto sulle loro idee progettuali, sulle possibili strategie per affinare e migliorare gli artefatti realizzati o sui collegamenti con concetti scientifici correlati ai fenomeni del mondo reale? L'impegno futuro per migliorare la qualità di questi laboratori non potrà prescindere dalla valutazione della loro efficacia nel fornire futuri educatori un esempio di come approcciarsi a contesti complessi, come quelli lavorativi, in cui viene richiesto loro di prevedere, progettare, collaborare e riflettere.

BIBLIOGRAFIA

- Bruner, J. S. (1966). *Toward a theory of instruction*. Harvard University Press.
- Butera, F. (2020). Le condizioni organizzative e professionali dello smart working dopo l'emergenza: progettare il lavoro ubiquo fatto di ruoli aperti e di professioni a larga banda. *Studi organizzativi*, 1, 141-165.
- Fabbri, L. (2003). La costruzione dell'identità professionale. il contributo delle storie formative. In A.A.V.V. (a cura di), *Il progettista di formazione individuale* (pp. 18-36). Ucodep.
- Fabbri, L. (2012). Le tecnologie incarnate nelle comunità professionali. Traiettorie di apprendimento trasformativo. In A.A.V.V. (a cura di), *L'educazione tra reale e virtuale* (pp. 101-109). La Scuola.
- Fabbri, L. (2014). Università e nuove professionalità. *MeTis-Mondi educativi. Temi indagini Suggestioni*, 4(1).
- Fabbri, L. (2017). Ricercatori ed insegnanti dentro setting di apprendimento trasformativo. In P. C. Rivoltella (a cura di), *L'agire didattico. Manuale per l'insegnante* (pp. 351-364). La Scuola.
- Fabbri, L. (2019). Le metodologie attive di ricerca: che cosa hanno a che fare azione, partecipazione, apprendimento e ricerca. *Educational reflective practices*, 1, 7-18. <https://doi.org/10.3280/ERP2019-001002>
- Fabbri, L., & Giampaolo, M. (2019). Apprendimento trasformativo e tecnologie "in uso" come paradigmi di ricerca per l'innovazione delle organizzazioni complesse. In M. R. P. Federighi (a cura di), *Digital Scholarship tra ricerca e didattica. Studi, ricerche, esperienze* (pp. 53-65). Franco Angeli.

- Fabbri, L., Giampaolo, M., & Di Benedetto, E. (2019). Quando la tecnologia sostiene le pratiche. La prospettiva del facilitatore tecnologico nella comunità "in buona salute". In A. G. F. Bruni (a cura di), *Media Education in Italia. Oggetti e ambiti della formazione* (pp. 297-306). Franco Angeli.
- Fabbri, L., Melacarne, C., Giampaolo, M., & Romano, A. (2017). Methodology and approaches to support employability and organizational learning at University of Siena: the case of the Teaching and Learning Center. In *Proceedings of the 2017 international Pre-Conference* (pp. 69-80). Atlanta, GA: American Association for Adult and Continuing Education.
- Fabbri, L., & Melacarne, C. (2016). Didattica dell'innovazione e innovazione didattica. L'apprendimento come condizione per il cambiamento. In M. Fedeli (a cura di), *Coinvolgere per apprendere. Metodi e tecniche partecipative per la formazione* (pp. 319-339). Pensa Multimedia.
- Fabbri, L., & Rossi, B. (2008). *Cultura del lavoro e formazione universitaria*. Franco Angeli.
- Lave, J. (1996). Teaching, as learning, in practice. *Mind, culture, and activity*, 3(3), 149-164. https://doi.org/10.1207/s15327884mca0303_2
- Lave, J., & Wenger, E. (2002). Practice, person, social world. In H. Daniels (a cura di), *An introduction to Vygotsky* (pp. 155-162). Routledge.
- Lacasa, P. (2013). *Learning in real and virtual worlds. Commercial video games as educational tools*. Palgrave MacMillan.
- Lodi, M. (2018). Pensiero Computazionale: dalle "scuole di samba della computazione" ai CoderDojo. *Atti del convegno Didamatica 2018*, Aprile 2018, Cesena, Italy. <https://hal.inria.fr/hal-01913063/document>
- Prensky, M. (2001). *Computer games and learning*. McGraw-Hill.
- Schmidt, J. P., Resnick, M., & Ito, J. (2016). Creative learning and the future of work. In D. Noedfords, V. Perf, & M. Seneges (Eds.), *Disrupting unemployment* (pp. 147-155). Ewing Marion Kaufman Foundation.
- Robinson, K. (2010). *Changing education paradigms*. <https://bit.ly/3nChCiL>
- Robinson, K., & Aronica, L. (2016). *Creative schools: The grassroots revolution that's transforming education*. Penguin books.
- Resnick, L. B. (1991). Shared cognition: Thinking as social practice. In L. B. Resnick, J. M. Levine, & S. D. Teasley (Eds.), *Perspectives on socially shared cognition* (pp. 1-20). American Psychological Association.
- Resnick, M. (2007). All I Really Need to Know (About Creative Thinking) I Learned (by Studying How Children Learn) in Kindergarten. In *Proceedings of the 6th ACM SIGCHI Conference on Creativity and Cognition* (pp. 1-6). ACM.
- Resnick M. (2014) Give P's a chance: Projects, peers, passion, play. In: Futschek G. & Kynigos C. (Eds.), *Constructionism and creativity: Proceedings of the Third International Constructionism Conference* (pp. 13-20). Austrian Computer Society.
- Resnick, M. (2018). *Come i bambini. Immagina, crea, gioca e condividi. Coltivare la creatività con il Lifelong Kindergarten del MIT*. Edizioni Centro Studi Erickson.
- Papert, S. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.
- Papert, S. (1991). Situating Constructionism, In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 1-11). Ablex Publishing.
- Piaget J. (1976). Piaget's Theory. In B. Inhelder, H.H. Chipman, & C. Zwingmann C. (Eds.), *Piaget and His School* (pp. 11-23). Springer. https://doi.org/10.1007/978-3-642-46323-5_2
- Piaget, J. (2016). *L'epistemologia genetica*. Edizioni Studium.
- Vygotskij L. S. (1997). *The collected works of LS Vygotskij: Problems of the theory and history of psychology*. Springer.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35. <https://doi.org/10.1145/1118178.1118215>
- Wing, J. (2017). Computational thinking's Influence on Research and Education for all. *International Journal of Educational Technology*, 25(2), 7-14. <https://doi.org/10.17471/2499-4324/922>

APPENDICE I

Scheda di progettazione compilata dalla tirocinante

| Titolo della sessione di formazione: Informatica creativa per gli Educatori | | |
|---|---|---|
| Obiettivi d'apprendimento Voglio far conoscere i fondamenti teorici del pensiero computazionale; Voglio far conoscere il programma Scratch Voglio far conoscere quali siano gli utilizzi più efficaci di questi programmi in ambito educativo. | | |
| Risultati d'apprendimento Al termine della sessione di formazione sarai in grado di: conoscere i principi pratici del coding utilizzando Scratch; creare un'animazione; usare degli effetti grafici; usare i blocchi causalità; usare i blocchi variabili. | | |
| Modalità di valutazione Check-list di valutazione degli artefatti prodotti dai partecipanti | | |
| Quando | Cosa | Materiale da utilizzare |
| 10 min. | Approcci teorici sul pensiero computazionale e sul coding | Slide: coding e pensiero computazionale |
| 5 min. | A cosa servono programmi come "Scratch"? | Slide: scratch |
| 10 min. | I° tutorial | |
| 10 min. | II° tutorial | |
| 10 min. | III° tutorial | |
| 10 min. | IV° tutorial | |
| 10 min. | V° tutorial | |
| 40 min. | Attività pratica | |
| 15 min. | Presentazione del mio progetto su scratch. | Il mio progetto salvato in Scratch |