

Videogiochi e abilità logico-matematiche: il progetto Game-Over

Videogames and logical mathematical ability: the Game-Over project

Lilliana D'Angelo^a

^a *Miur, Istituto Omnicomprensivo di Montenero di Bisaccia (CB), liliana.dng@gmail.com*

Abstract

Oggetto del presente articolo è l'ampliamento dell'esperienza didattica attraverso l'utilizzo dei videogiochi come mezzo per promuovere l'apprendimento attivo (learning by doing).

Con il progetto "Game-Over" ci si è posti l'obiettivo di fornire, attraverso l'ideazione e la creazione di un videogioco, esperienze formative significative che simulano scenari di apprendimento interattivi, allo scopo di migliorare le competenze di problem-solving degli studenti.

Parole chiave: videogame; apprendimento attivo; problem-solving.

Abstract

Focus of this paper is the possibility to broaden the learning experience by using videogames to promote a "learning by doing" methodology.

Purpose of the "Game-Over" project was in fact giving, through designing and implementation of a videogame, valuable educational experiences simulating interactive learning scenarios, aiming to improve students' problem solving skills.

Keywords: videogame; learning by doing; problem-solving.

1. Introduzione

Negli ultimi anni è molto evidente il fenomeno della diffusione e dell'uso costante dei videogiochi da parte dei ragazzi, non solo sul pc ma anche su dispositivi quali smartphone e tablet.

Da parte della scuola è opportuno non ignorare, bensì considerare queste attività nel loro aspetto ludico quali fonti di valore aggiunto nei nuovi contesti della formazione, poiché forniscono un'ulteriore motivazione all'apprendimento.

Le Information and Communications Technology (ICT) applicate alla formazione (Overmars, 2004; 2006), infatti, non costituiscono solo un mezzo efficace di raggiungimento del sapere, ma cambiano il modo stesso di apprendere grazie alle potenzialità di:

- rendere autonoma la costruzione del proprio sapere;
- potenziare le abilità di analisi e problem-solving;
- aumentare lo sviluppo di abilità sociali e comunicative;
- personalizzare l'apprendimento.

Il progetto “Game-Over” si colloca in un contesto di ampliamento dell'esperienza didattica condotto da diversi anni nelle classi della scuola secondaria di primo grado dell'Istituto Omnicomprensivo di Montenero di Bisaccia.

Un primo importante approccio è stata la sperimentazione in una classe prima dell'uso giochi dell'IPRASE (2001), che richiedevano, sia al docente che agli alunni, di fornire una valutazione critica sull'efficacia dei giochi descrivendone i punti di forza e di debolezza.

L'esperienza è stata poi arricchita grazie al progetto cl@ssi2.0 (Avvisati, Hennessy, Kozma & Vincent-Lacrin, 2013), che ha consentito un utilizzo più frequente delle ICT nella didattica.

Il laboratorio “Game-over”, finalizzato al potenziamento delle abilità logico-matematiche e all'acquisizione di competenze digitali, è stato attivato in orario extracurricolare con un gruppo di alunni della classe terza. L'approccio iniziale è stato centrato sul concetto di algoritmo, di linguaggio di programmazione (i ragazzi hanno appreso le istruzioni del linguaggio Qbasic e hanno creato semplici programmi di matematica), successivamente hanno utilizzato il software Game Maker.

2. Game Maker

Game Maker è un ambiente di sviluppo integrato per la produzione di videogame di utilizzo immediato. Il software nella sua versione completa è a pagamento, ma esiste una versione “lite” disponibile gratuitamente. Pur disponendo di una modalità di programmazione classica basata su un linguaggio di programmazione proprietario, offre una seconda modalità di sviluppo basata su una programmazione “visuale”: in questa modalità, è possibile realizzare una scena disponendo gli oggetti con il mouse, inoltre è possibile associare a diversi eventi (quali pressioni di un tasto del mouse o della tastiera, eventi temporali, o collisioni con altri oggetti) una sequenza di azioni.

La Figura 1 mostra la finestra iniziale di un nuovo gioco in Game Maker, in particolare la finestra aperta *Room Properties* consente di creare una stanza, ovvero un'ambientazione,

disponendo oggetti all'interno della griglia e personalizzandola impostando uno sfondo per l'ambientazione.

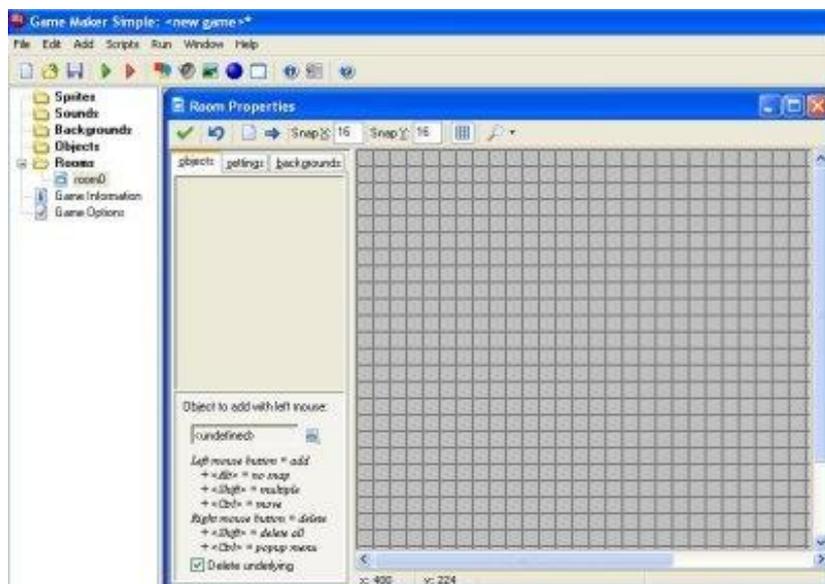


Figura 1. Finestra di definizione della scena in Game Maker.

Gli *oggetti* sono composti da due parti: lo *sprite* ovvero la componente grafica, che può essere caricata come immagine, e l'*object* che consente di programmare il comportamento in risposta ad un eventuale input del giocatore (click, tastiera, etc.) o all'interazione con altri oggetti.



Figura 2. Caricamento dello sprite di un oggetto.

La Figura 2 mostra la creazione di uno sprite, in particolare l'immagine di un drago, che potrà essere associata ad un oggetto da animare nel videogioco. È possibile caricare un set di immagini visibili una dopo l'altra in modo da creare un'animazione.

La Figura 3 mostra la schermata di creazione di un nuovo oggetto. Sul lato sinistro, è possibile personalizzare l'oggetto con un nome, rendere graficamente visibile uno sprite, e impostare la visibilità e la solidità dell'oggetto. Al centro, vi è la lista di eventi (al momento vuota) a cui reagisce l'oggetto, e la lista di azioni che l'oggetto eseguirebbe in risposta all'evento selezionato. A destra infine, è possibile scegliere un'azione (sono visualizzate le azioni di movimento, quali il movimento di una casella della griglia della room, l'inizio o la modifica del movimento in una direzione, il rimbalzo e lo spostamento in una determinata casella).

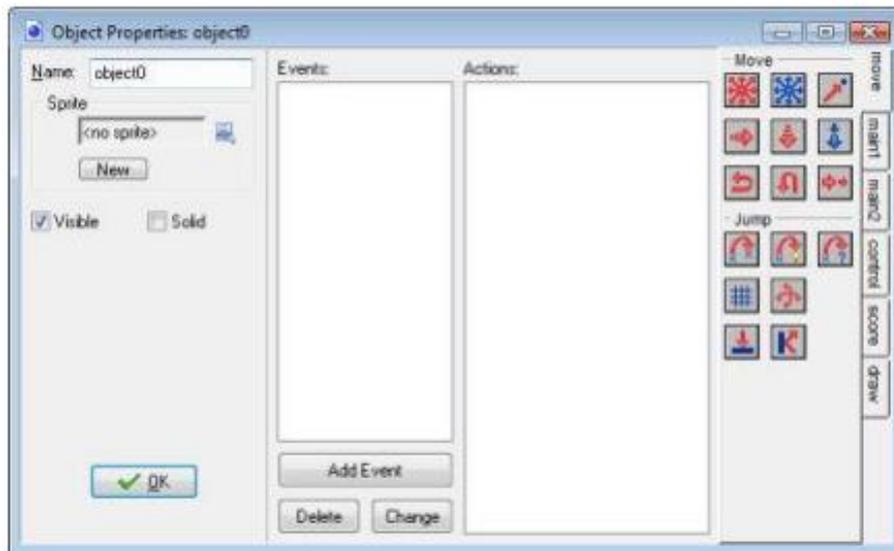


Figura 1. Creazione di un oggetto.

In Figura 4 è rappresentato un esempio di sequenza di azioni in risposta all'evento di pressione del tasto "Sinistra" della tastiera, che verifica che l'oggetto a cui è associata non sia in collisione con altri: in questo caso, viene avviato il movimento in una direzione, in caso contrario, l'oggetto viene spostato in una posizione diversa e viene emesso un suono.

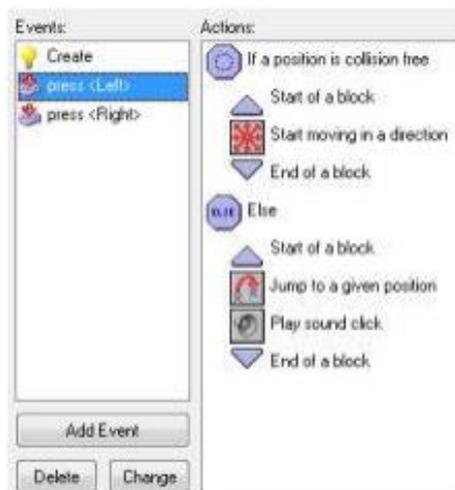


Figura 2. Esempio di algoritmo in Game Maker.

È evidente come una tale modalità costituisca, per uno studente delle scuole secondarie di primo grado, una forma di espressione di un algoritmo molto intuitiva ed accattivante, e costituisca un buon inizio per avvicinarsi all'informatica ed ai linguaggi di programmazione.

3. Il videogioco Biomi-Maze

Considerata la complessità della creazione di un videogioco, in relazione al tempo disponibile e alle competenze da sviluppare, si è scelto di realizzare un *maze game*, ovvero un labirinto dal quale fare uscire il protagonista dopo aver superato vari ostacoli.

Per mantenere l'aspetto didattico, il labirinto ha un tema, scelto dai ragazzi, i biomi: ad ogni livello, il protagonista si trova in uno specifico bioma (deserto, etc.), deve evitare "mostri" (animali caratteristici del bioma) che, se toccati, lo riportano al punto di partenza, e deve rispondere a domande riguardo il bioma in esame.

Il programma didattico adottato è stato strutturato in cinque step:

1. progettazione astratta del videogioco, mediante la scelta del tema, e l'ideazione dell'ambientazione;
2. definizione e implementazione degli oggetti del gioco, realizzando sprite e oggetti di personaggi, animali e ostacoli;
3. definizione degli eventi e delle regole del gioco, implementando la logica per il movimento del personaggio e degli animali, e le azioni da eseguire in risposta ad eventi (quali il contatto tra un personaggio ed un ostacolo o un personaggio ed un animale);
4. definizione dei quesiti attraverso la conoscenza dei contenuti disciplinari;
5. definizione della sequenza condizionata e non degli eventi del gioco.

Le lezioni, nel rispetto del learning by doing, sono state esercitazioni interattive con esempi opportunamente predisposti, illustrati e realizzati immediatamente dagli alunni. I ragazzi hanno progettato i livelli, ricercato le immagini per gli sfondi considerando il bioma e la rispettiva flora e fauna, alternando labirinti con quesiti a labirinti con "mostri".

3.1. Implementazione del videogioco

La Figura 5 mostra la *room* del primo livello (ambientato nel deserto), popolata con dei muri che definiscono il labirinto. I muri sono degli oggetti "inerti" il cui unico comportamento è quello di impedire al personaggio il movimento nella stessa direzione. L'obiettivo del gioco è, per ogni livello, superare il labirinto, rispondendo correttamente alle domande ed evitando di essere attaccati dagli animali in movimento, e raggiungere il cartello "exit".

Sono inoltre visibili gli oggetti che rappresentano il protagonista e gli ostacoli, in questo caso delle pergamene.

La Figura 6 mostra l'evento di collisione del protagonista con una pergamena, in risposta al quale viene posta una domanda al giocatore: se la risposta è corretta, la pergamena sparisce e il protagonista può continuare, altrimenti il protagonista viene riportato alla posizione di partenza.

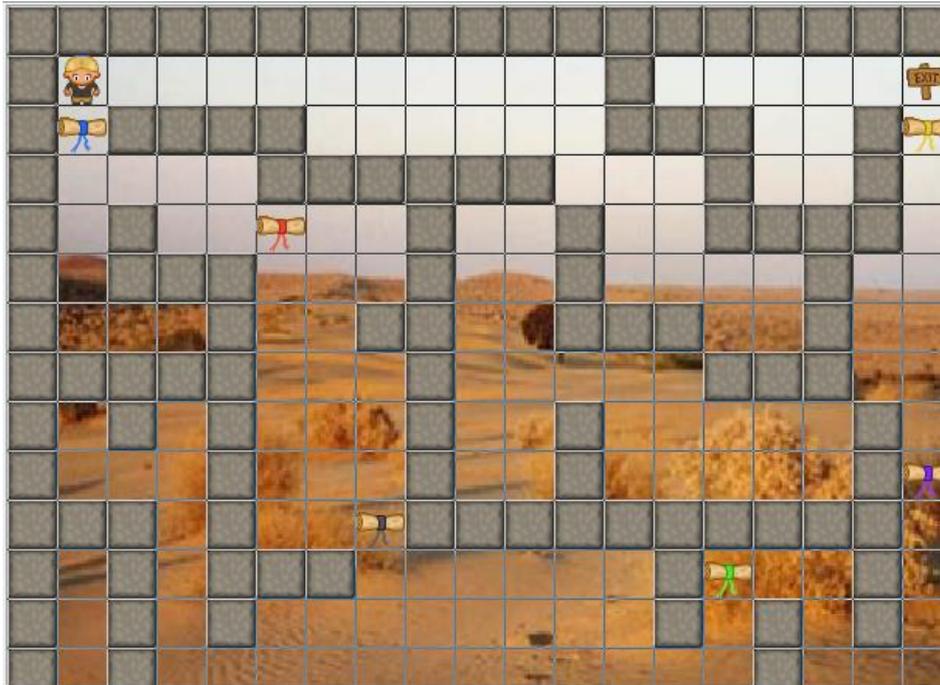


Figura 5. Room del primo livello.

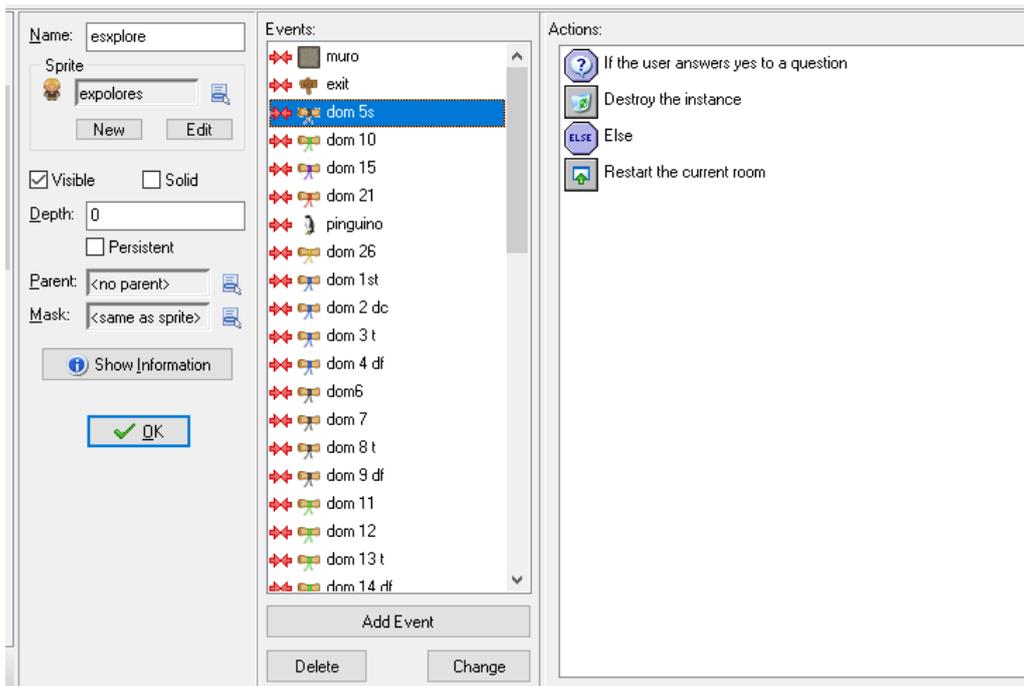


Figura 6. Evento di collisione tra il protagonista e una pergamena.

La Figura 7 mostra la room di un livello più avanzato, in cui, invece delle pergamene e delle domande, sono presenti degli antagonisti (delle volpi) che si muovono lungo il

labirinto e che il protagonista deve evitare. Vi sono anche degli oggetti mobili (massi) che il protagonista può spostare per bloccare le volpi.

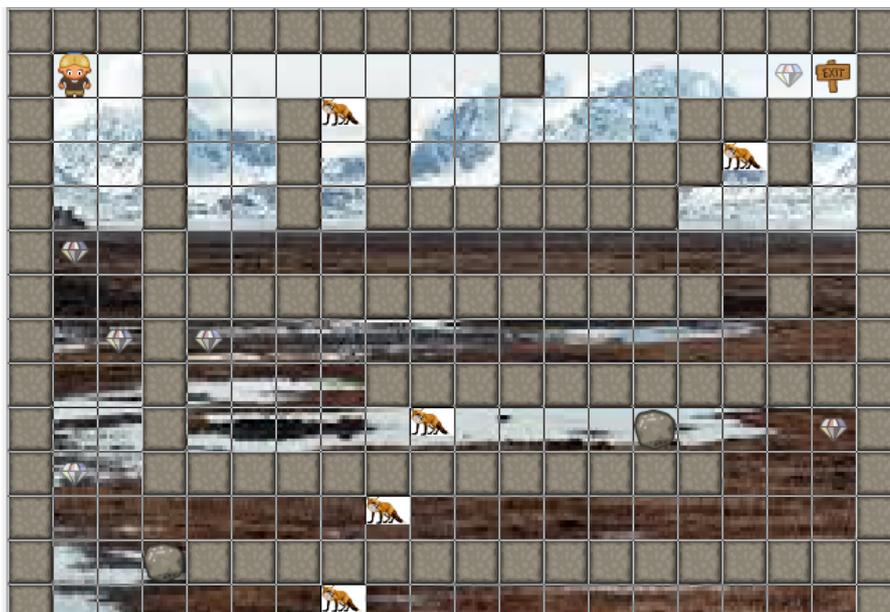


Figura 7. Ottavo livello.

I massi e i muri vengono infatti utilizzati per definire il comportamento degli animali: in Figura 8 è mostrato il comportamento di una volpe che si muove orizzontalmente lungo il labirinto, invertendo la direzione al contatto con i muri.

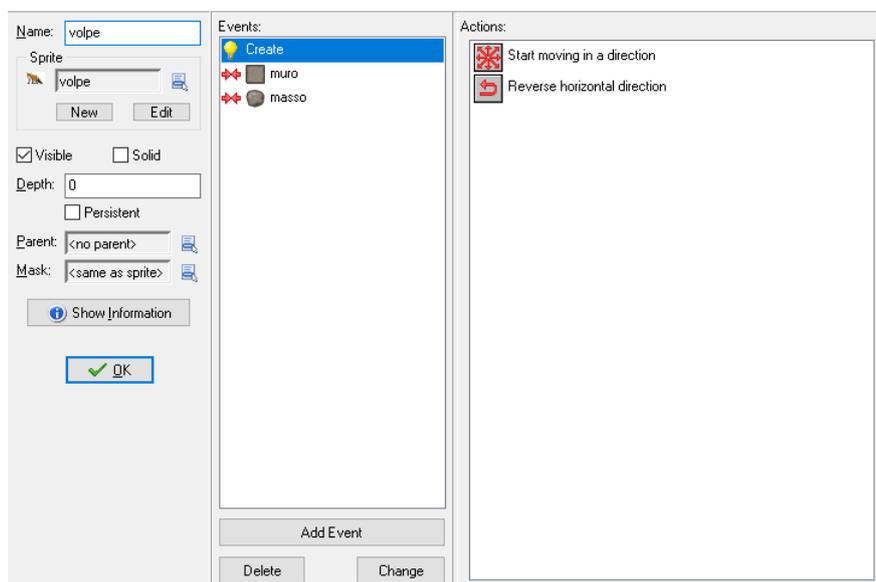


Figura 8. Evento di collisione tra un animale e un masso che delimita il labirinto.

La Figura 9 invece mostra l'evento di collisione tra il protagonista e il "mostro" del livello ambientato nel bioma polare: un pinguino. Il protagonista deve evitare gli animali, nel caso venga toccato, viene riportato alla posizione di partenza.

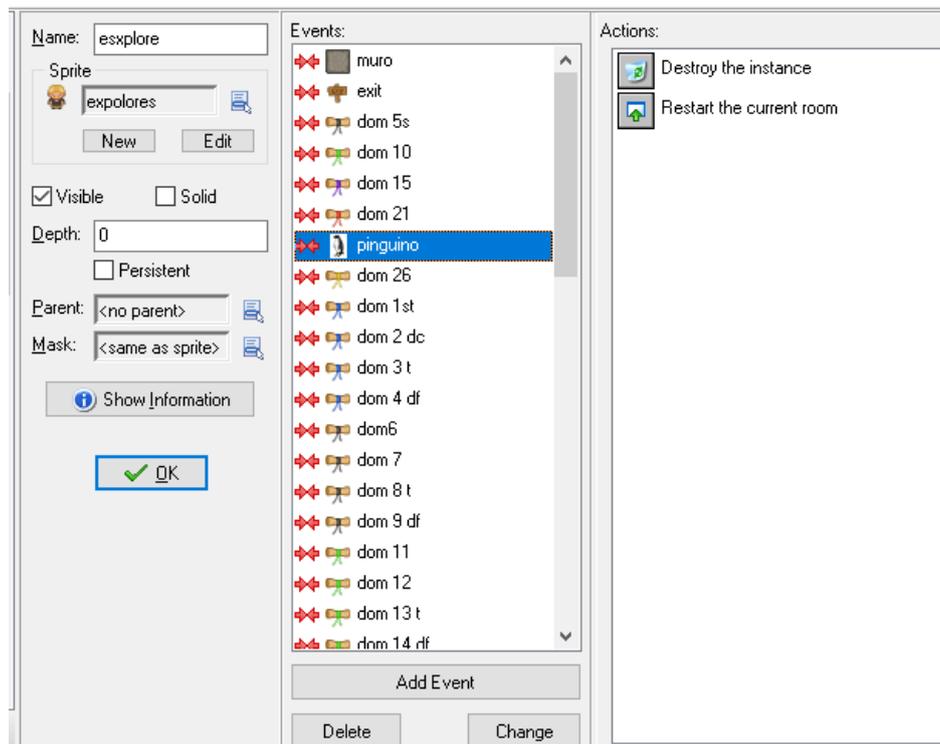


Figura 9. Evento di collisione tra il protagonista ed un oggetto attivo.

4. Conclusioni

Il lavoro proposto è stato ben lontano da quello dei programmatori professionisti, ma per un po' di tempo, sia pure in forma limitata, i ragazzi hanno assunto il piacevole ruolo di autori e creatori, e non solo utilizzatori, di videogiochi. Evitando di banalizzare le semplici attività realizzate con Game Maker, i ragazzi hanno avuto modo di comprendere che prima di arrivare al codice di un programma bisogna comprendere la situazione problematica che si ha di fronte, analizzarla e formalizzarla.

In particolare, la fase di avvio del progetto ha presentato diverse criticità. Posti davanti al problema di immaginare un nuovo gioco, è stato necessario guidare i ragazzi nello sviluppo di strategie decisionali e di problem-solving per evitare situazioni di blocco creativo, e soprattutto stimolare la loro creatività laddove le idee mancassero di originalità.

L'attività ha avuto nel complesso un esito molto positivo, evidenziando grande curiosità, interesse e un'ottima partecipazione da parte dei ragazzi.

La fase di ricerca e definizione dei quesiti tematici ha richiesto uno studio degli argomenti e uno sforzo a non banalizzarli eccessivamente, producendo un effettivo potenziamento delle conoscenze disciplinari parallelamente al potenziamento delle abilità digitali.

Infine, l'attività collettiva di ideazione e progettazione, il lavoro in team e lo scambio di idee in un contesto diverso dall'aula ha prodotto un sostanziale miglioramento delle relazioni di gruppo fra gli alunni.

Dato il successo del progetto, l'attività è stata riproposta attraverso la realizzazione di un nuovo videogioco, sull'argomento del corpo umano, introducendo elementi di novità rispetto al progetto precedente.

Inoltre, si stanno valutando diverse alternative, sia per la programmazione di videogiochi (utilizzando ad esempio il software Scratch) sia per la programmazione mobile (utilizzando l'ambiente MIT App Inventor), che per l'elettronica, con la possibilità di utilizzare hardware a basso costo e facile da programmare (ad esempio Raspberry PI, Arduino e CodeBug) allo scopo di creare semplici circuiti e robot.

Bibliografia

- Arduino. <https://www.arduino.cc/> (ver. 30.03.2017).
- Avvisati, F., Hennessy, S., Kozma, R.B., & Vincent-Lacrin, S. (2013). *Review of the Italian Strategy for Digital Schools*. Paris: OECD.
- Codebug. <http://www.codebug.org.uk/> (ver. 30.03.2017).
- Game Maker. <http://www.yoyogames.com/gamemaker> (ver. 30.03.2017).
- IPRASE (2001), *Giochi didattici*, http://try.iprase.tn.it/prodotti/software_didattico/giochi/ (ver. 30.03.2017).
- MIT App Inventor. <http://appinventor.mit.edu/explore/index-2.html> (ver.30.03.2017).
- Overmars, M. (2004). Teaching computer science through game design. *Computer*, 37(4), 81–83.
- Overmars, M. (2006). *Progettare videogiochi con Game Maker, versione 6.1*. www.isissitinfo.altervista.org/materiale/gamemaker/GameMaker.doc (ver. 30.03.2017).
- Raspberry PI. <https://www.raspberrypi.org/> (ver. 30.03.2017).
- Scratch. <https://scratch.mit.edu/> (ver. 30.03.2017).