



Certificazione CODING





UNIVERSITÀ DEGLI STUDI
DI GENOVA



INDICE

Le Certificazioni EPIC	3
Introduzione - Certificazione EPIC Coding	4
Syllabus della Certificazione EPIC Coding	5
Allegato 1	9
Allegato 2	12
Il processo di Certificazione	17
Lesson Plan e Rubrica di valutazione	18

Copyright © 2024 - Università degli Studi di Genova, Associazione EPICT Italia

Autori

Università di Genova: Giovanni Adorni, Giorgio Delzanno, Giovanna Guerrini, Davide Parmigiani, Valentina Pennazio

Associazione EPICT Italia: Federica Brigandì, Francesca Craviotto, Giuliana Lo Giudice, Giovanna Suetta, Angela Maria Sugliano. Per le rubriche di valutazione (Allegato 2) ha collaborato Giulia Carmeci.

Editor: Angela Maria Sugliano

Redazione: Serena Battigelli

Le Certificazioni EPICT

Le **Certificazioni EPICT - European Pedagogical ICT licence** - sono il risultato di un progetto Europeo¹ che ha sviluppato un sistema di syllabi di competenza e un sistema di valutazione e certificazione, relativi all'uso pedagogico delle tecnologie digitali.

I syllabi delle Certificazioni descrivono cosa i docenti devono essere in grado di fare quando usano in classe le diverse tecnologie digitali e i diversi approcci metodologici legati all'uso del digitale. **I docenti che applicano le competenze descritte nei syllabi delle Certificazioni padroneggiano:**

- **gli aspetti per la gestione ottimale delle attività didattiche:** la preparazione del setting d'aula, la predisposizione dei materiali didattici, le pratiche per la gestione degli studenti mentre svolgono le attività col digitale, la valutazione utilizzando strumenti digitali;
- **gli aspetti educativi sottesi all'uso delle diverse tecnologie digitali:** nella definizione gli obiettivi di una attività didattica sanno esplicitare le competenze digitali che eserciteranno gli studenti, le competenze trasversali che verranno allenate, gli stili di apprendimento che verranno favoriti o stimolati durante la realizzazione delle attività.

Per tutti i docenti e formatori che hanno sviluppato competenza nell'uso del digitale per la didattica, le Certificazioni EPICT sono lo strumento per attestarla in modo esplicito e formale con un documento pubblico emesso da una istituzione universitaria. L'attestazione di competenza è fornita in riferimento ai syllabi delle Certificazioni e in riferimento al framework DigCompEdu. **Per i docenti non ancora di ruolo** e che sono iscritti nelle *Graduatorie Provinciali per le Supplenze*, le Certificazioni EPICT in quanto "certificazioni informatiche" hanno un valore di 0,5 punti ciascuna e possono essere dichiarate fino a 4 (ordinanza ministeriale n. 60 del 10 luglio 2020).

I syllabi delle Certificazioni EPICT sono annualmente soggetti a revisione per verificare la necessità di un aggiornamento. La revisione e l'aggiornamento sono a cura di un team che comprende sia docenti universitari sia docenti formatori di insegnanti ed esperti nell'uso pedagogico delle tecnologie digitali membri della Teacher Educators Academy dell'Associazione EPICT Italia. L'impianto di certificazione è sempre stato mantenuto allineato con i framework internazionali sull'uso delle tecnologie digitali a Scuola: dal framework UNESCO ICT-CFT (ICT Competence Framework for Teachers), al framework e-CF (e-Competence Framework), il framework DigCompEdu

¹ European pedagogical ICT Licence - <https://cordis.europa.eu/project/id/22210>

Introduzione - Certificazione EPICT Coding

Perché a scuola si dovrebbe insegnare il coding? Sono molte le risposte a questa domanda che possono essere riassunte in quattro macro-obiettivi.

- 1) Per allenare negli studenti il pensiero logico (oltre a quello computazionale), stile cognitivo improntato sulla categorizzazione e scomposizione di problemi complessi in elementi più semplici molto utile per lo studio di ogni disciplina e per affrontare più in generale le sfide della vita;
- 2) Perché le conoscenze della disciplina “Informatica”, al pari delle conoscenze sulle altre discipline, devono ormai nella società dell’informazione e della comunicazione essere un patrimonio di tutti, utile anche pensando all’orientamento scolastico verso professioni che sicuramente trovano spazio nel mondo del lavoro. Per questo i docenti devono - seppur in modo non specialistico - saper condurre le attività di coding utilizzando il linguaggio dell’informatica e consapevoli dei costrutti computazionali che si propongono pur lavorando con semplici linguaggi di coding visuale.
- 3) Per utilizzare strategie didattiche improntate all’apprendimento attivo stimolando la creatività, il pensiero critico, *softskills* come la capacità di collaborazione in caso di lavoro di gruppo, di apprendimento auto-regolato (con lo stimolo a monitoraggio delle attività considerando gli errori quale momento di auto-valutazione e crescita),
- 4) Per formare le competenze digitali quando si chiede agli studenti di *cercare informazioni o risorse in rete* da utilizzare nei progetti, *comunicare e collaborare* in ambienti digitali quando si chiede agli studenti di collaborare a distanza e condividere i propri progetti, la competenza di *produzione corretta di contenuti digitali* quando oltre alla produzione dei propri progetti l’insegnante chiede di attribuire a questi opportune licenze, di esercizio di *comportamenti sicuri ed etici*, e di *problem-solving* se si considera lo stimolo continuo al trovare soluzioni sia in termini di creatività, sia in termini di uso delle applicazioni, sia in termini di scrittura di un codice corretto ed efficiente.

Il syllabus della Certificazione EPICT “Coding” descrive l’insieme delle competenze che un docente deve possedere per progettare e condurre significative lezioni utilizzando applicazioni per il coding visuale usate come tecnologie didattiche capaci di attrarre e motivare gli studenti portandoli ad acquisire conoscenze disciplinari attraverso attività stimolanti e creative.

Il presente sillabo illustra le competenze di un docente in grado di proporre agli studenti attività di coding visuale. Le applicazioni /strumenti di riferimento sono le seguenti:

- Carte direzionali tipo Cody Roby
- Scratch junior
- Scratch
- Pocket Code

Syllabus della Certificazione EP ICT Coding

1 - PROGETTAZIONE - COMPETENZE PER LA FASE DI PROGETTAZIONE DIDATTICA -Aree DigComp Edu - [Pratiche di insegnamento]

CODICE	I docenti sono in grado di:
1.1	Identificare le finalità per cui a scuola si propone agli studenti attività di coding: per realizzare storytelling, giochi, app per dispositivi,...
1.2	Identificare - oltre agli obiettivi disciplinari - gli obiettivi digitali, trasversali e gli stili di apprendimento che verranno allenati negli studenti a seguito di attività didattiche che vedono lo sviluppo progetti con il coding (v. allegato 2)
1.3	<p>Progettare attività didattiche scegliendo:</p> <ul style="list-style-type: none"> ● le applicazioni per fare coding e le funzioni delle applicazioni adeguate all'ordine di scuola e alle competenze degli studenti (si veda l'allegato 1) ● attività proposte da iniziative nazionali o internazionali (es. L'ora del codice) ● le strategie più adeguate al contesto specifico: attività di remix, riordino, descrizioni testuali, debugging, creazione originale di codice

2 - PREPARAZIONE - COMPETENZE PER LA FASE DI PREPARAZIONE DELLE ATTIVITA' DIDATTICHE- Aree DigComp Edu - [Risorse digitali] [Pratiche di insegnamento] [Valutazione] [Valorizzazione delle potenzialità degli studenti]

CODICE	I docenti sono in grado di:
2.1	<p>Definire e motivare la scelta del setting d'aula adottato (disposizione dei banchi, screen mirroring, uso di device personali o a piccoli gruppi, BYOD, ecc.,...) in base a:</p> <ul style="list-style-type: none"> ● ordine di scuola ● caratteristiche dell'utenza ● dispositivi a disposizione ● caratteristiche fisiche dell'aula ● metodologia didattica (lezione frontale, lavoro di gruppo)
2.2	<p>Predisporre per l'uso delle applicazioni prescelte:</p> <ul style="list-style-type: none"> ● installare e/o guidare gli studenti a installare le applicazioni sui dispositivi

2 - PREPARAZIONE - COMPETENZE PER LA FASE DI PREPARAZIONE DELLE ATTIVITA' DIDATTICHE- Aree DigComp Edu - [Risorse digitali] [Pratiche di insegnamento] [Valutazione] [Valorizzazione delle potenzialità degli studenti]	
	<ul style="list-style-type: none"> ● fare le adeguate richieste a un tecnico per il supporto o la risoluzione di eventuali problematiche ● verifica delle autorizzazioni necessarie per accedere alle applicazioni on line ● predisporre eventuali applicazioni per il mirroring dei dispositivi mobili sullo schermo interattivo della classe oppure predisporre istruzioni per la condivisione dello schermo a distanza ● considerare le buone pratiche per l'uso dei dispositivi personali (BYOD) a scuola (v. allegato 2) ● ...
2.3	<p>Predisporre materiali didattici per lo svolgimento delle attività:</p> <ul style="list-style-type: none"> ● realizzare/reperire e remixare un progetto/programma da condividere con gli studenti ● scrivere la consegna per la guida delle attività di apprendimento che contengano: <ul style="list-style-type: none"> ○ le istruzioni per lo svolgimento del compito in modo preciso e dettagliato con eventuali istruzioni personalizzate/individualizzate in coerenza con l'ordine di scuola e livello degli studenti (v. allegato 1) ○ fornire indicazioni sulla modalità di consegna della presentazione/infografica/mappa: sul portale e-learning, via email, stampata,... ○ un esempio di modello o di prodotto finale ○ ... ● sviluppare strumenti per la valutazione (diagnostica e/o formativa e/o sommativa) degli apprendimenti: rubriche di valutazione esplicitando quali elementi del prodotto finale saranno valutati (v. allegato 2) ed eventualmente verificare il raggiungimento degli obiettivi digitali e trasversali (v. allegato 2)
2.4	<p>Allestire ambienti online (portali e-learning, ambienti cloud) in cui condividere i materiali didattici e in cui gli studenti consegneranno i loro prodotti</p> <ul style="list-style-type: none"> ● aree per la consegna agli studenti dei materiali didattici utili per realizzare le attività proposte ● aree di comunicazione per il confronto e il supporto durante le eventuali attività a casa ● aree di consegna da parte degli studenti dei progetti realizzati

3 - CONDUZIONE - COMPETENZE PER LA FASE DI CONDUZIONE DELLE ATTIVITA' DIDATTICHE - Aree DigComp Edu - [Pratiche di insegnamento]
 [Valutazione] [Favorire le competenze digitali degli studenti]

CODICE	I docenti sono in grado di:
3.1	Spiegare le funzioni delle applicazioni prescelte illustrando l'ambiente di lavoro dell'applicazione, <i>oppure</i> illustrando le funzioni a partire da un progetto già sviluppato o da un progetto di prova (remix) , <i>oppure</i> proponendo attività basate sul metodo esplorativo da parte degli studenti, <i>oppure</i> ,....
3.2	Condurre la classe nelle fasi di attività per un progetto di coding: <ul style="list-style-type: none"> ● definizione dell'algoritmo (scomposizione, rappresentazione grafica) ● ideazione, ● progettazione, ● sviluppo, ● testing, ● debugging
3.3	Supportare gli studenti mentre svolgono le attività: <ul style="list-style-type: none"> ● suggerire le tecniche per mandare in esecuzione i programmi per la verifica e l'auto-correzione ● porre domande agli studenti per stimolare le idee, far confrontare gli studenti sulle diverse soluzioni adottabili, rintracciare gli errori e trovare soluzioni ● utilizzare gli indicatori della rubrica per guidare gli studenti nello sviluppo del proprio prodotto
3.4	Condurre gli studenti ad approfondire il contesto più ampio della disciplina informatica come ad esempio spiegare agli studenti come è strutturato un calcolatore, il suo linguaggio,

	4 - DOCUMENTARE - COMPETENZE PER LA FASE A CONCLUSIONE DELLE ATTIVITA' DIDATTICHE - Aree DigComp Edu - [Coinvolgimento e valorizzazione professionale]
CODICE	I docenti sono in grado di:
4.1	Creare una documentazione delle attività svolte: <ul style="list-style-type: none">• Report testuali, Contenuti Audio/Video, Slideshow, infografiche
4.2	Condividere la documentazione realizzata: <ul style="list-style-type: none">• con i colleghi di Istituto, in repository multimediali², in occasioni pubbliche (convegni, conferenze, pubblicazioni scientifiche,..)

² *Per i docenti della Liguria si segnala la Piattaforma "Osservatorio" sviluppata dal Progetto Scuola Digitale Liguria come luogo elettivo per la documentazione delle attività svolte. La piattaforma consente di 1) documentare le attività innovative col digitale svolte in classe, 2) consultare quanto svolto da altri docenti, 3) vedere pubblicato il proprio progetto sulla mappa georeferenziata delle scuole della Liguria, 4) creare un proprio portfolio digitale e condividerlo.

→ Osservatorio dei progetti innovativi Regione Liguria - www.scuoladigitaleliguria.it

Allegato 1

Di seguito una indicazione delle attività / concetti / tipologie di funzioni delle applicazioni e strumenti più usate per fare coding unplugged e a blocchi, ordinati per livelli di scuola o di competenza degli studenti. Le Linee guida CINI sono visionabili a questo indirizzo web: <https://www.consortio-cini.it/index.php/it/component/attachments/download/745>

	Il docente guida la classe nelle seguenti attività e nell'uso delle seguenti funzioni delle più usate applicazioni per il coding a Scuola.	Are di Competenze rispetto alle Linee guida CINI per l'Informatica
LIVELLO INIZIALE coding unplugged	Introduzione semplificata funzionamento dei computer	Area Consapevolezza Digitale
	Scomposizione di un problema in sotto problemi per la sua risoluzione	Area Algoritmi
	Comprensione del concetto di algoritmo utilizzando "algoritmi quotidiani": lavarsi i denti, ordinare le figurine, ecc.	Area Algoritmi
	Consapevolezza delle pratiche computazionali: programmare in modo incrementale, provare e risolvere, riusare e remix	Area Algoritmi Area Programmazione
LIVELLO BASE – Applicazione di riferimento Scratch Junior /Scratch Elementi da utilizzare per lo sviluppo dei programmi per realizzare semplici esempi di schemi narrativi (idee convergenti, viaggio dell'eroe, la	Scelta dello sfondo: scelta o creazione di immagini che rispettano privacy e copyright	Area Consapevolezza digitale
	Componenti e struttura di un'applicazione in ScratchJr: sprite (posizione e movimento nella scena 2D, aspetto, suono, scene)	Area Dati e Informazioni
	Sequenza di comandi: script scritti come strisce di blocchi di comandi	Area Programmazione
	Eventi (inizio, quando si preme,...ecc)	Area Programmazione
	Iterazione cicli finiti (ripetere per K volte) e infiniti (ripetere per sempre)	Area Programmazione
	Esecuzione concorrente di più script e più sprite (ad esempio, animazione di due o più sprite)	Area Programmazione
	Messaggi: payload dei messaggi come colori, invio di azioni e ricezione di eventi	Area Dati e Informazioni

	Il docente guida la classe nelle seguenti attività e nell'uso delle seguenti funzioni delle più usate applicazioni per il coding a Scuola.	Are di Competenze rispetto alle Linee guida CINI per l'Informatica
montagna, ...)	Alternativa/selezione tramite messaggi	Area Programmazione
	Creazione di sottoprogrammi utilizzando i messaggi	Area Algoritmi
	Semplici esempi di schemi narrativi (idee convergenti, viaggio dell'eroe, la montagna, ...) e come codificarli usando le scene e tutti gli altri elementi (sequenze, ecc.)	Area Creatività digitale
LIVELLO INTERMEDIO Applicazione di riferimento Scratch Elementi da utilizzare per lo sviluppo dei programmi per realizzare oltre a storytelling, programmi/app (es.: effettuare calcoli, videogiochi)	Conoscenza dell'ambiente di sviluppo: scratch cloud, progetti pubblici/privati, esecuzione, condivisione, remix	Area Programmazione Area Consapevolezza digitale Area Creatività digitale
	Uso di editor per creare icone di sfondo e sprite, suoni	Area Programmazione Area Creatività digitale
	Componenti e struttura di un'applicazione in Scratch: sprite e script - posizione e movimento nella scena 2D, aspetto, suono, scene	Area Programmazione
	Creazione di animazioni utilizzando sequenze di blocchi di movimento	Area Algoritmi Area Programmazione
	Creazioni di animazioni ripetendo sequenze di blocchi (ripetizioni limitate/illimitate)	Area Programmazione
	Creazione di script multipli per sprite diversi - programmazione concorrente	Area Programmazione
	Creazione di script che reagiscono agli eventi utilizzando blocchi situazione (ad esempio, quando si lancia, quando si preme, quando si tocca...)	Area Algoritmi Area Programmazione
	Ripetizione di una sequenza di istruzioni con semplici condizioni (cicli con condizioni): ad esempio, muovere uno sprite verso un determinato obiettivo e aggiungere effetti quando l'obiettivo viene raggiunto	Area Algoritmi Area Programmazione
	Selezione reciproca, esclusiva, es: gestione delle collisioni con diversi tipi di oggetti	Area Algoritmi Area Programmazione
Creazione di nuovi blocchi, es.: per creare un nuovo tipo di movimento o di animazione (es: il blocco "salto").	Area Algoritmi	

	Il docente guida la classe nelle seguenti attività e nell'uso delle seguenti funzioni delle più usate applicazioni per il coding a Scuola.	Aree di Competenze rispetto alle Linee guida CINI per l'Informatica
	Inviare e ricevere messaggi (es: attivare uno sprite in reazione a un determinato evento, conversazione, ecc.)	Area Programmazione Area Consapevolezza digitale,
	Sincronizzare sequenze di eventi attraverso temporizzazione	Area Programmazione Area Creatività digitale
	Utilizzare blocchi di input e output, ad esempio per creare giochi interattivi.	Area Programmazione Area Consapevolezza digitale Area Creatività digitale
	Esplorare le estensioni di Scratch (musica, webcam, ecc.) per creare giochi	Area Programmazione Area Consapevolezza digitale Area Creatività digitale
<p>LIVELLO INTERMEDIO/ AVANZATO</p> <p>Applicazioni di riferimento Scratch e per dispositivi mobili, Pocket Code</p> <p>Elementi da utilizzare per creare storytelling applicazioni di gioco con diversi livelli</p>	<p>Uso di gruppi di sprite, scene, editor di blocchi, script, variabili incorporate (es: posizione dell'oggetto, dimensione, velocità): Creazione di nuovi blocchi, es: per creare un nuovo tipo di movimento o di animazione (es: il blocco "salto")</p>	<p>Area Programmazione</p>
	<p>Creare variabili, ad esempio, per gestire e visualizzare il punteggio corrente del gioco.</p>	<p>Area Programmazione Area Dati e Informazioni</p>
	<p>Creare liste (numeri, stringhe di testo ecc.), es; per gestire oggetti raccolti nel gioco.</p>	<p>Area Programmazione Area Dati e Informazioni</p>
	<p>Utilizzare espressioni e logica: creazione di espressioni e relazioni logiche su variabili.</p>	<p>Area Programmazione Area Dati e Informazioni</p>
	<p>Creare programmi modulari utilizzando blocchi di clonazione</p>	<p>Area Programmazione</p>
	<p>Creare un'applicazione di gioco con diversi livelli di difficoltà</p>	<p>Area Creatività digitale</p>
	<p>Pocket Code</p>	
	<p>Usare gruppi di sprite e scene per modularizzare un gioco</p>	<p>Area Creatività digitale</p>
	<p>Aggiungere controlli basati su sensori del dispositivo, ad esempio movimenti guidati dall'inclinazione del dispositivo</p>	<p>Area Programmazione</p>

	Il docente guida la classe nelle seguenti attività e nell'uso delle seguenti funzioni delle più usate applicazioni per il coding a Scuola.	Are di Competenze rispetto alle Linee guida CINI per l'Informatica
	(giroscopio).	Area Dati e Informazioni Area Consapevolezza digitale,
	Esplorare le estensioni con sistemi esterni (ad esempio, interrogazioni web, comunicazione wifi e bluetooth, ecc.)	Area Consapevolezza Digitale Area Creatività digitale
	Creare animazioni e simulazioni utilizzando cicli nidificati (ad esempio giochi con balistica, sprite che ripetono all'infinito schemi di movimento dati, ad esempio da sinistra a destra)	Area Programmazione

Allegato 2

COMPETENZE DIGITALI

Lo sviluppo di progetti di coding nella didattica comporta sempre lo sviluppo di competenze di *creazione di contenuti digitali* (capacità d'uso di applicativi per il coding, capacità di gestione dei copyright e licenze delle risorse utilizzate,...) e di *problem solving* (capacità di risolvere eventuali problemi con gli applicativi utilizzati, per la creatività necessaria a progettare e realizzare un programma, ...).

Inoltre, a seconda dell'attività particolare svolta, le attività che portano allo sviluppo di programmi possono comportare l'esercizio da parte degli studenti delle seguenti ulteriori **competenze digitali**:

- alfabetizzazione su informazioni e dati (nel caso di utilizzo di applicazioni on line o di ricerca dei contenuti on line da inserire nella presentazione,...)
- comunicazione e collaborazione (nel caso di presentazioni collaborative e di condivisioni di queste in ambienti digitali,...)
- sicurezza/uso responsabile del digitale (nel caso di sviluppo delle presentazioni/infografiche/mappe in ambienti che necessitano di accesso con account personali, nel caso di risorse scaricate sui dispositivi personali per proteggerli durante lo scaricamento di immagini o altre risorse dal web,...)

COMPETENZE TRASVERSALI

In riferimento alle otto competenze chiave europee, le attività che portano allo sviluppo di programmi consentono di mettere in campo le competenze di area STEAM e cioè la *competenza digitale e la competenza matematica e in scienze, tecnologie e ingegneria*

Inoltre, a seconda dell'attività particolare svolta, le attività che portano allo sviluppo di programmi possono comportare l'esercizio da parte degli studenti delle seguenti ulteriori competenze trasversali :

- competenza alfabetica funzionale (se attività svolte in ambito umanistico)
- competenza multilinguistica (se attività svolte in ambito linguistico)
- competenza personale, sociale e capacità di imparare a imparare (se attività svolte in gruppo,...)
- competenza in materia di cittadinanza (se attività svolte su tematiche di educazione civica)
- competenza imprenditoriale/spirito di iniziativa (se attività svolte come compito autentico)
- competenza in materia di consapevolezza ed espressione culturali (se attività svolte in ambito artistico/o in progetti multiculturali)

STILI DI APPRENDIMENTO

In riferimento agli stili di apprendimento riferibili agli stili cognitivi interattivo, analitico, introspettivo (McKenzie- Intelligenze multiple e tecnologie per la didattica³), le attività che portano allo sviluppo di programmi consentono esercitare negli studenti le intelligenze che fanno riferimento all'ambito analitico.

Inoltre, a seconda dell'attività particolare svolta, le attività che portano allo sviluppo di programmi possono comportare l'esercizio da parte degli studenti dei seguenti ulteriori stili di apprendimento:

- Stile di apprendimento Interattivo (nel caso di attività realizzate in modo collaborativo...)
- Stile di apprendimento Introspettivo (nel caso di attività che prevedono un contenuto su aspetti valoriali, sociali, etici,...)

³ https://www.dirittoeconcorsi.it/pub/media/blfa_files/9788859007098.pdf - <https://www.assoepict.it/attivit%C3%A0/progetti/stili-di-apprendimento-e-didattica-digitale>

METODOLOGIE, STRATEGIE E TECNICHE

In riferimento al punto 1.3 del syllabus, si precisa che la metodologia didattica fornisce un quadro teorico che guida l'insegnamento e l'apprendimento. Le tecniche didattiche sono gli strumenti operativi utilizzati per favorire l'apprendimento, mentre le strategie didattiche sono i piani d'azione adottati dagli insegnanti per raggiungere gli obiettivi di apprendimento.

BYOD

Nel caso per lo svolgimento delle attività didattiche si preveda l'uso degli dispositivi personali degli studenti (riferimento syllabus punto 2.2), è necessario prevedere le regole che a livello di Istituto si seguono per garantire un uso corretto di questa forma di setting didattico. Si riprendono di seguito alcuni elementi tratti da: ESEMPIO DI REGOLAMENTO PER LA PRATICA DEL BYOD (da Progetto Scuola Digitale Liguria - <https://www.scuoladigitaleliguria.it/community/attivita/ricerca-azione/byod.html>).

I dispositivi personali degli studenti quando usati a scuola devono seguire regole che assicurano sicurezza per gli apparati della Scuola e degli stessi dispositivi personali:

- usati solo per soli scopi didattici e solo dopo previa autorizzazione esplicita dell'insegnante (no gioco libero, no registrazioni/scatto di foto non autorizzate)
- usati in modo personale e non prestati ai compagni
- manleva della scuola in caso di accidentali incidenti
- uso della rete internet della scuola a cui gli studenti accedono con account personale
- diritto di ispezione da parte dei docenti dei dispositivi personali utilizzati per scopi didattici

RUBRICHE DI VALUTAZIONE

Un esempio di rubrica di valutazione per valutare un progetto di coding

CRITERIO	INDICATORE
Sprite (personaggio)	Uso dello sprite predefinito Personalizzazione dello sprite Uso della libreria di sprite Uso della libreria dei costumi Creazione di nuovi sprite Modifica del nome predefinito dello sprite
Stage (sfondo)	Uso dello stage predefinito Personalizzazione dello stage Uso della libreria di stage Creazione di nuovi stage

CRITERIO	INDICATORE
	Modifica del nome predefinito dello stage
Documento che contiene lo Storyboard	<p>Cronologia chiara</p> <p>Presenza delle scene chiave che mostrano lo sviluppo completo della storia</p> <p>Presenza di indicazioni sui dialoghi, movimenti, aspetti, sfondi e sprite</p> <p>Formattazione del documento che contiene lo storyboard secondo la consegna</p>
Script (codice): Istruzioni condizionali (logical thinking)	Presenza di istruzioni condizionali minime, caratterizzanti, avanzate, secondo la consegna (Blocchi: <i>se, allora; se/altrimenti</i> ; operatori booleani che combinano più condizioni E, O e NON)
Script (codice): Rappresentazione dei dati (informazioni sugli sprite, memorizzazione di dati)	<p>Presenza di istruzioni per modificare gli attributi degli sprite (Blocchi: Movimento, Aspetto, Suono)</p> <p>Memorizzazione di valori in variabili (Blocchi: Variabili)</p> <p>Memorizzazione di più valori contemporaneamente (uso di Liste)</p>
Script (codice): Parallelismo	Esecuzione di più script contemporaneamente usando istruzioni minime, caratterizzanti, avanzate, secondo la consegna (Blocchi Evento: <i>“Quando si clicca su bandierina verde”</i> ; <i>“Quando si clicca sul tasto...”</i> ; <i>“Quando si clicca su un oggetto”</i> ; <i>“Quando ricevo messaggio...”</i> ; <i>“Quando lo sfondo passa a...”</i>)
Script (codice): Sincronizzazione	Presenza di sincronizzazione del comportamento degli sprite usando istruzioni minime, caratterizzanti, avanzate, secondo la consegna (uso blocco: <i>“Attendi”</i> ; uso di messaggi; uso blocchi: <i>“Attendi fino a quando”</i> , <i>“Quando lo sfondo passa a...”</i>)
Script (codice): Interattività utente	Presenza di interattività utente usando istruzioni minime, caratterizzanti, avanzate, secondo la consegna (uso blocco: <i>“Quando si clicca su bandierina verde”</i> ; uso blocchi: <i>“Quando si clicca sul tasto...”</i> , <i>“Quando si clicca questo sprite”</i> , scrivere risposte; uso webcam/microfoni)
Script (codice): Controllo di flusso (sequenze/cicli)	<p>Presenza di sequenza blocchi con inizio e fine</p> <p>Uso dei blocchi <i>“Ripeti”/ “Per sempre”</i></p> <p>Uso del blocco <i>“Ripeti fino a quando”</i></p>
Script (codice): Astrazione e scomposizione del problema	<p>Capacità di suddividere un problema in parti più piccole per facilitare la programmazione, le modifiche al codice e il debug</p> <p>Uso di più script/più sprite</p> <p>Creazione di nuovi blocchi che consistono in una sequenza di istruzioni</p> <p>Uso dei Cloni</p>
Efficienza del codice elaborato	- Codice pulito, breve

CRITERIO	INDICATORE
	<ul style="list-style-type: none"> - Assenza di ripetizioni - Frequenza di funzionamento - Assenza di codice inutilizzato (dead code)
Capacità di debugging	<ul style="list-style-type: none"> - Frequenza - Individuazione errori - Apporto di miglioramenti
Linguaggio di programmazione	<ul style="list-style-type: none"> - Conoscenza funzioni base, caratterizzanti, avanzate
Rispetto dei tempi indicati nella consegna	Definizione delle fasi di lavoro con relative tempistiche Puntualità di consegna Consegna in anticipo
Modalità di consegna	Rispetto delle indicazioni nella consegna (es. formato di pubblicazione del file, nome del file, area di consegna, ...)
Autonomia	Frequenza della richiesta di intervento dell'insegnante o di un pari Ricerca autonoma di risorse per rispondere ad un bisogno
Collaborazione / Lavoro di gruppo	Rispetto del proprio ruolo nel gruppo Capacità di relazione con gli altri in base al loro ruolo Comportamenti centrati sul compito Comportamenti volti al mantenimento del buon clima di gruppo (socio-emozionali): usare forme di cortesia, ascolto, aiuto,... Assenza di comportamenti individualisti (persegue propri scopi e non quelli definiti dal gruppo) Capacità di negoziare le proprie ed altrui idee

Il processo di Certificazione

Il candidato per completare il processo di Certificazione ha a disposizione 6 mesi dalla data di accesso al portale (<https://epict.unige.it/Avvio>) . Di seguito l'articolazione delle attività:

- **FASE 1**

- **CONSEGNA DELLE EVIDENZE RICHIESTE** - Le consegne delle evidenze devono avvenire entro l'ultimo giorno di ogni mese;
- **FEEDBACK DI VALUTAZIONE** - Il feedback da parte dei certificatori viene inviato entro il mese successivo alla consegna delle evidenze: contiene la valutazione e spunti che saranno oggetto del colloquio della fase 2. Il candidato può accedere alla fase 2 se riceve un punteggio minimo di 60 punti (massimo 80) conteggiato in base a una rubrica condivisa.

- **FASE 2**

- **ESAME DI CERTIFICAZIONE** - Le sessioni di certificazione sono previste ogni primo venerdì di ogni mese. Il colloquio d'esame è tenuto da una commissione formata da docenti universitari e dal certificatore che ha curato il processo di feedback. I colloqui consistono in un approfondimento delle evidenze presentate e si svolgono sulla base del feedback che il candidato ha ricevuto. La modalità del colloquio d'esame è sia in presenza sia a distanza. Il punteggio per il colloquio è di minimo 12 punti/ massimo 20 punti, assegnati secondo la rubrica di valutazione di cui sopra.

Per la Certificazione EPICT Coding sono richieste le seguenti evidenze:

1) Lesson Plan redatto secondo il format fornito che verrà consegnato nell'apposita sezione del portale certificazioni.

Il lesson plan descrive una unità didattica svolta da parte del candidato, Il format guida il candidato a mettere in evidenza tutte le competenze così come descritte nel Syllabus EPICT Coding. Il lesson plan deve essere accompagnato dalle seguenti evidenze: il testo della consegna fornita agli studenti, una rubrica o altro strumento di valutazione, altre evidenze a discrezione del candidato.

2) Un progetto con il coding.

Il progetto - realizzato con una delle applicazioni indicate nella sezione "Introduzione" - è relativo al lesson plan presentato e contiene tutti o parte degli elementi indicati nell'Allegato 1 del Syllabus.

3) Presentazione/Portfolio del candidato.

Il candidato compila un format dove dichiara evidenze della propria esperienza professionale di uso del coding per la didattica.

Lesson Plan e Rubrica di valutazione

La rubrica di valutazione del lesson plan (con i relativi allegati) considera ogni elemento del lesson plan e lo mette in relazione ai punti del syllabus. La valutazione avviene secondo i seguenti tre parametri:

A = presenza/coerenza del contenuto con l'elemento del syllabus

B = presenza di un contenuto che oltre ad essere coerente con la richiesta del syllabus contiene elementi descrittivi che permettono di comprendere in modo completo l'esperienza didattica descritta

C = presenza di un contenuto che oltre ad essere coerente e completo contiene elementi di specifica qualità (linguaggio specifico della didattica, descrizioni sintetiche ma molto chiare ed esaustiva, un'ottima impaginazione,)

Descrizione Lesson Plan	Codice Syllabus Coding
SINTESI DELL'ATTIVITÀ	1.1
HARDWARE/ SOFTWARE/ APPLICAZIONI	1.3
OBIETTIVI DI APPRENDIMENTO	1.2
PRODOTTI RICHIESTI AGLI STUDENTI	1.1
STRATEGIA/TECNICA DIDATTICA	1.3
METODO DI VALUTAZIONE DEGLI APPRENDIMENTI	2.3
MODALITÀ DI DOCUMENTAZIONE DELLE ATTIVITÀ SVOLTE	4.1 / 4.2
APPLICAZIONI E SETTING (FASI DI LAVORO)	2.1 / 2.2 / 2.4
MATERIALI DIDATTICI	2.3

DESCRIZIONE SVOLGIMENTO ATTIVITÀ	3.1 / 3.2 / 3.3
----------------------------------	-----------------

Al termine del processo, dopo il colloquio di certificazione, la Certificazione viene erogata con l'indicazione del livello. Il livello è definito come sintesi fra gli indicatori delle rubriche di valutazione per le fasi 1 e 2 delle valutazioni che individuano tre livelli di uso pedagogico delle tecnologie digitali così come indicati nel framework DigCompEdu Sperimentatore, Esperto, Leader.

<p>Sperimentatore (B1)</p> <p>Le evidenze e il colloquio danno conto di una abitudine sporadica all'uso del coding come metodo didattico per svolgere specifiche attività scolastiche senza mettere in atto particolari strategie tese all'insegnamento dell'informatica o all'esplicita valutazione di sviluppo negli studenti di pensiero analitico e computazionale.</p> <p>[da 72 – 80 punti secondo la rubrica di valutazione]</p>	<p>Esperto (B2): 80 -90</p> <p>B2 - [Esperto] Le evidenze e il colloquio danno conto dell'abitudine all'uso del coding come strumento didattico per attività disciplinari e interdisciplinari utilizzando in parallelo altri ambienti/strumenti digitali, con competenza nella formazione sui concetti chiave dell'informatica e con attenzione specifica allo sviluppo e valutazione delle competenze di pensiero logico e computazionale.</p> <p>[da 80 – 90 punti secondo la rubrica di valutazione]</p>	<p>Leader (C1): 90 -100</p> <p>Le evidenze e il colloquio danno conto dell'attitudine e di una consolidata competenza nell'uso del coding come strumento didattico per attività disciplinari e interdisciplinari con competenza nella formazione sui concetti chiave dell'informatica e con attenzione specifica allo sviluppo e valutazione delle competenze di pensiero logico e computazionale. Le evidenze dimostrano attività mirate alla diffusione presso i colleghi del coding come strumento didattico.</p> <p>[da 90 – 100 punti secondo la rubrica di valutazione]</p>
--	--	--