

1. PIANO E CONTENUTO DEL MODULO



Versione 1.0

1. PIANO DEL MODULO

Piano del Modulo	
Titolo del modulo	Graffio e Sbattimento
Durata	3 ore (in totale)
Obiettivi di apprendimento	Alla fine di questo modulo, i partecipanti potranno - • Comprendi le basi della programmazione a blocchi usando Scratch. • Riconosci come i concetti di programmazione visiva si traducono nello sviluppo di app mobili con Flutter. • Crea un progetto interattivo semplice in Scratch e discuti come una logica simile possa essere implementata in Flutter.
Piano di lezione	Fondamenti del Design e dello Sviluppo Digitale Modulo di Formazione per Giovani Sordi Durata: ~90 minuti (2 argomenti × 40 minuti, + pausa 10 minuti)
	1. Introduzione a Scratch (40 minuti) • Obiettivo: Comprendere la programmazione basata su blocchi e creare animazioni semplici. • Punti principali da trattare: ○ Cos'è Scratch? ○ Interfaccia: Livello, Sprite, Blocchi.

Piano del Modulo	
	○ Programmazione basata su eventi: "Quando la bandiera è stata cliccata". ○ Creare un'animazione semplice o un minigioco. ● Attività: Crea un progetto semplice: muovere un personaggio sullo schermo usando i tasti freccia. ● Metodologia di insegnamento: Dimostrazione + pratica guidata. 2. Passaggio da Scratch a Flutter (40 minuti) ● Obiettivo: Mostra come la logica di programmazione appresa in Scratch si applica allo sviluppo di app Flutter. ● Punti principali da trattare: ○ Somiglianze chiave: eventi, cicli, condizionali. ○ Introduzione a Flutter: Widget, struttura dell'interfaccia utente, basi del linguaggio Dart. ○ Comprendere la connessione visuale-codice. ● Attività: Confronta un evento di "clic del pulsante" Scratch con un evento Flutter <code>onPressed</code> button. ● Metodologia di insegnamento: Confronto visivo, esempi affiancati, discussione. -
Materiali Necessari	● Laptop ● Internet ● Proiettore ● Materiali stampati

2. CONTENUTO DEL MODULO

Contenuto	Descrizione	Contenuto
Contenuto della lezione	Alfabetizzazione digitale	L'alfabetizzazione digitale significa avere le conoscenze e le competenze per utilizzare strumenti digitali come software di progettazione e linguaggi di programmazione per creare app, siti web e contenuti online. Include la comprensione di come funzionano i sistemi digitali e la capacità di costruire e progettare prodotti digitali semplici. • Comprendere i sistemi digitali – come funzionano computer, applicazioni e internet. • Uso creativo degli strumenti – progettare e costruire prodotti digitali semplici. • Problem solving attraverso la tecnologia – applicare soluzioni digitali alle esigenze della vita reale. Termini digitali chiave: 1. Codifica basata su blocchi ○ Un modo visivo di programmare collegando "blocchi" come pezzi di puzzle. ○ Esempio: Scratch – dove i blocchi vengono usati per controllare personaggi, suoni e animazioni. 2. Eventi e Loop ○ Evento: un'azione che attiva un programma (ad esempio, "quando la bandiera verde è stata cliccata" in Scratch o il pulsante "onPressed" in Flutter).

Contenuto	Descrizione	Contenuto
		○ Loop: ripetere una certa azione più volte (ad esempio, "per sempre" o "ripeti finché" in Scratch). 3. Widget ○ I mattoni base di Flutter che creano un'app (ad esempio, pulsanti, immagini, campi di testo). ○ Ogni parte di un'app Flutter è un widget che può essere personalizzato o combinato. 4. Sviluppo multiplatforma ○ Sviluppo di applicazioni che funzionano su diversi dispositivi e sistemi operativi (Android, iOS, Web). ○ Flutter è un esempio di strumento di sviluppo multiplatforma.
Riassunto video	Link su Youtube	• <i>Nozioni di base per scratch:</i> www.scratch.mit.edu • <i>Introduzione a Flutter:</i> www.flutter.dev • <i>Panoramica del linguaggio dei dardi:</i> www.dart.dev
Attività		1. Attività di gruppo: Crea un minigioco da zero Obiettivo: Conoscere lo spazio di lavoro Scratch • Obiettivo: Incoraggiare il lavoro di squadra e la creatività applicando i concetti di programmazione basati su blocchi. • Istruzioni: 1. Dividere i partecipanti in piccoli gruppi (3–4 membri). 2. Ogni gruppo sceglie un concetto di gioco semplice (ad esempio, collezionare oggetti del personaggio, fuga nel labirinto o sfida di tempo di reazione).

Contenuto	Descrizione	Contenuto
		- Utilizzando Scratch, i gruppi progettano personaggi (sprite), aggiungono sfondi e programmano interazioni tramite eventi e loop. - Incoraggia l'aggiunta di almeno un sistema di punteggio o una condizione di vittoria/sconfitta. Risultato: Ogni gruppo crea un minigioco giocabile da condividere con la classe.
		2. Role-Play: Traduce la logica di scratch in Flutter Obiettivo: I partecipanti comprendono la logica di programmazione su due piattaforme Obiettivo: Aiutare i partecipanti a collegare la codifica a blocchi visivi a quella testuale in Flutter. Istruzioni: - Dopo aver completato il minigioco, assegna a un membro del gruppo il ruolo di "Scratch Developer" e un altro come "Flutter Developer". - Lo sviluppatore Scratch spiega la logica del loro minigioco (ad esempio, "Quando lo sprite tocca la mela, il punteggio aumenta di 1"). - Lo sviluppatore Flutter dimostra come quella logica potrebbe essere espressa in Flutter (ad esempio, usando un evento <code>onPressed</code>, istruzioni condizionali o aggiornando una variabile di punteggio in Dart).

Contenuto	Descrizione	Contenuto
		Risultato: I partecipanti si esercitano a spiegare la logica di programmazione su due piattaforme, migliorando l'alfabetizzazione digitale e le capacità comunicative.
Valutazione		1. A cosa serve principalmente Scratch? A) Controllo hardware B) Codifica visiva per principianti ☒ C) Solo programmazione testuale D) Nessuna
	Quiz	2. Quale lingua usa Flutter? A) Java B) Dart ☒ C) Python D) C++
		3. In Scratch, quale blocco avvia la maggior parte dei programmi? A) Quando la bandiera verde clicca ☒ B) Ripeti finché C) Per sempre D) Aspetta
		4. Per quale tipo di programmazione è più conosciuto Scratch? A) Programmazione ☒ basata su blocchi B) Programmazione a riga

Contenuto	Descrizione	Contenuto
		di comando C) Machine learning D) Gestione di database
		5. In Flutter, cos'è un "widget"? A) Un dispositivo hardware B) Un mattone base dell'interfaccia ☒ utente C) Un linguaggio di programmazione D) Una connessione a un database
		6. Cosa significa "sviluppo multiplatforma"? A) Costruire app solo per Android B) Costruire app che girano su più sistemi ☒ operativi C) Scrivere codice senza computer D) Usare solo programmazione basata su blocchi