

MODULE PLAN AND CONTENT

Module 5

Scratch & Flutter



1. MODULE PLAN

Module Plan	
Module Title	Scratch & Flutter
Duration	3 hours (total)
Learning Objectives	By the end of this module, participants will be able to - • Understand the basics of block-based coding using Scratch. • Recognize how visual programming concepts translate into mobile app development with Flutter. • Build a simple interactive project in Scratch and discuss how similar logic can be implemented in Flutter.
Lesson Plan	Digital Design & Development Basics Training Module for Deaf Youth Duration: ~90 minutes (2 topics × 40 minutes, + Break 10 minutes)
	1. Introduction to Scratch (40 minutes) • Objective:

Module Plan	
	Understand block-based programming and create simple animations. • Main Points to Cover: ◦ What is Scratch? ◦ Interface: Stage, Sprites, Blocks. ◦ Event-driven programming: "When flag clicked". ◦ Creating a simple animation or mini-game. • Activity: Build a simple project: moving a character across the screen using arrow keys. • Teaching Methodology: Demonstration + guided hands-on practice. 2. Transitioning from Scratch to Flutter (40 minutes) • Objective: Show how programming logic learned in Scratch applies to Flutter app development. • Main Points to Cover: ◦ Key similarities: Events, loops, conditionals. ◦ Introduction to Flutter: Widgets, UI structure, Dart language basics. ◦ Understanding the visual-to-code connection. • Activity: Compare a Scratch "button click" event with a Flutter <code>onPressed</code> button event. • Teaching Methodology: Visual comparison, side-by-side examples, discussion. -
Materials Needed	• Laptop • Internet

Module Plan	
	• Projector • Printed materials

2. MODULE CONTENT

Content	Description	Content
Lesson Content	Digital Literacy	Digital literacy means having the knowledge and skills to use digital tools like design software and programming languages to create apps, websites, and online content. It includes understanding how digital systems work and being able to build and design simple digital products. • Understanding digital systems – how computers, applications, and the internet work. • Creative use of tools – designing and building simple digital products. • Problem-solving through technology – applying digital solutions to real-life needs. Key Digital Terms: 1. Block-Based Coding ○ A visual way of programming by connecting “blocks” like puzzle pieces. ○ Example: Scratch – where blocks are used to control characters, sounds, and animations. 2. Events & Loops

Content	Description	Content
		○ Event: an action that triggers a program (e.g., “when green flag clicked” in Scratch or “onPressed” button in Flutter). ○ Loop: repeating a certain action multiple times (e.g., “forever” or “repeat until” in Scratch). 3. Widgets ○ The basic building blocks in Flutter that create an app (e.g., buttons, images, text fields). ○ Every part of a Flutter app is a widget that can be customized or combined. 4. Cross-Platform Development ○ Developing applications that work across different devices and operating systems (Android, iOS, Web). ○ Flutter is an example of a cross-platform development tool. -
Video summary	Youtube Links	• <i>Scratch Basics:</i> www.scratch.mit.edu • <i>Intro to Flutter:</i> www.flutter.dev • <i>Dart Language Overview:</i> www.dart.dev
Activity		1. Group Activity: Build a Mini-Game in Scratch Goal: Get to know the Scratch workspace • Objective: Encourage teamwork and creativity while applying block-based coding concepts. • Instructions:

Content	Description	Content
		1. Divide participants into small groups (3–4 members). 2. Each group chooses a simple game concept (e.g., character collecting objects, maze escape, or reaction time challenge). 3. Using Scratch, groups design characters (sprites), add backgrounds, and program interactions using events and loops. 4. Encourage adding at least one scoring system or win/lose condition. • Outcome: Each group creates a playable mini-game to share with the class.
		2. Role-Play: Translate Scratch Logic into Flutter Goal: Participants understand programming logic across two platforms • Objective: Help participants connect visual block-based coding to text-based coding in Flutter. • Instructions: 1. After completing the mini-game, assign one group member to act as the "Scratch Developer" and another as the "Flutter Developer." 2. The Scratch Developer explains the logic of their mini-game (e.g., "When the sprite touches the apple, score increases by 1"). 3. The Flutter Developer demonstrates how that logic could be expressed in Flutter (e.g., using an <code>onPressed</code> event, conditional statements, or updating a score variable in Dart). • Outcome: Participants practice explaining programming logic across two platforms, improving digital literacy and communication skills.

Content	Description	Content
Assessment	Quiz	1. What is Scratch mainly used for? A) Hardware control B) Visual coding for beginners ✓ C) Text-based programming only D) None
		2. Which language is used by Flutter? A) Java B) Dart ✓ C) Python D) C++
		3. In Scratch, which block starts most programs? A) When green flag clicked ✓ B) Repeat until C) Forever D) Wait
		4. What type of programming is Scratch best known for?

Content	Description	Content
		A) Block-based programming ✓ B) Command-line programming C) Machine learning D) Database management
		5. In Flutter, what is a “widget”? A) A hardware device B) A basic building block of the user interface ✓ C) A programming language D) A database connection
		6. What does “cross-platform development” mean? A) Building apps only for Android B) Building apps that run on multiple operating systems ✓ C) Writing code without a computer D) Using only block-based programming