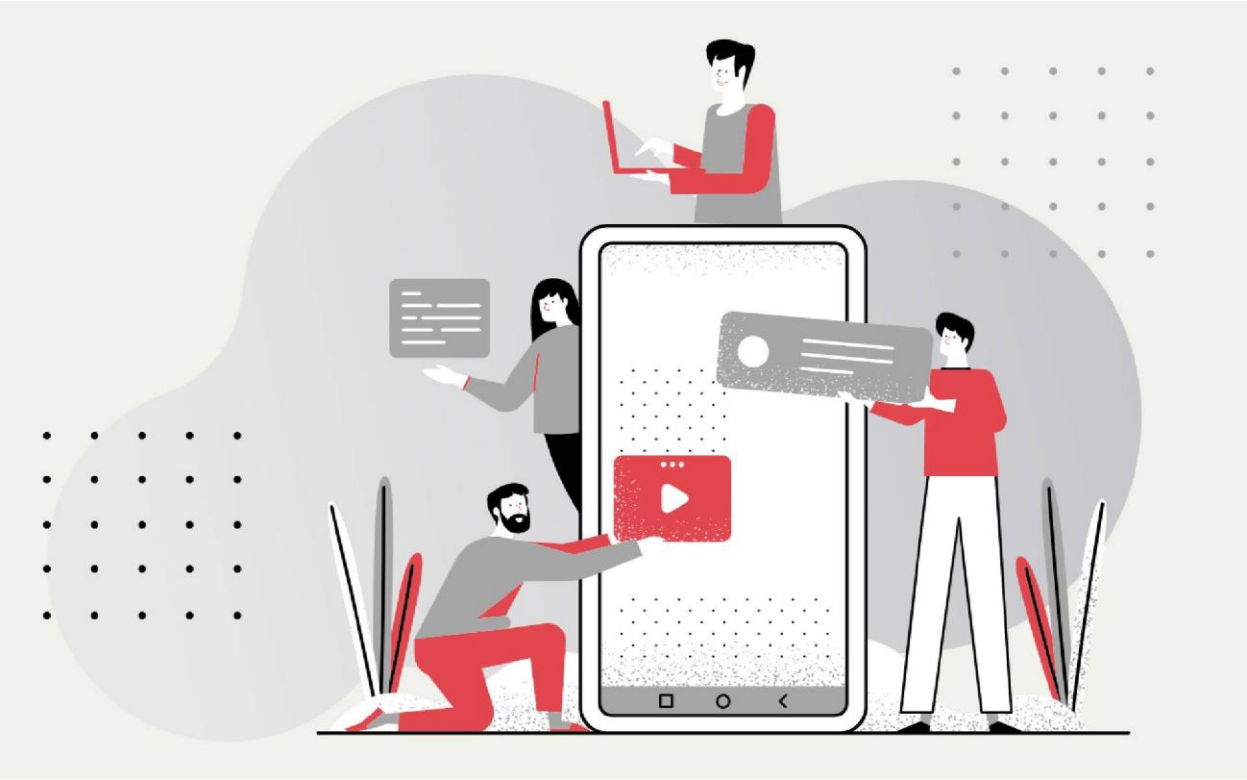
# APP LAB



| TEACHER: CERTIFIED TRAINER | DURATION: 16 CLASSES (x3 LEVELS ) (1 HOUR PER CLASS) | MODE: ONLINE AND OFFLINE |
|---|---|---|

App Lab allows users to build simple applications using a blend of block-based and text-based JavaScript coding. It focuses on event-driven programming, meaning students can design apps that respond to user inputs like button clicks or key presses. App Lab is often used in courses like *CS Discoveries* to teach fundamental programming concepts and web-based application development. It supports the creation of mobile apps and interactive tools with an easy-to-use interface.

CYBORG
ROBOTICS ACADEMY PVT. LTD.

# COURSE CURRICULUM

## Level 1: Introduction to Basics

- Color Manipulation: Learn to work with colors and apply them dynamically.
- Sound Integration: Explore how to incorporate sound and music into your projects.
- Visibility Management: Understand how to show and hide different elements in your app.
- User Interaction: Develop skills to respond to user clicks and inputs.
- Score Management: Create a basic score system to track points in your app.
- Shapes and Nesting: Experiment with drawing shapes and nesting elements within each other.
- Movement Control with Variables: Use variables to control movement and animations.
- Design Patterns: Create intricate designs using loops and variable manipulation.
- Lists and Geometry: Manage multiple elements using lists and geometric concepts.
- Skill Assessment: Reflect on and evaluate the skills learned throughout the level.

## Level 2: Intermediate Concepts

- Dynamic Sliders: Utilize sliders to enable real-time adjustments and color changes.
- Drawing with Sliders: Combine user input from sliders to create sketches.
- Randomization: Implement random number generation for interactive experiences.
- Debugging Techniques: Learn to identify and fix bugs in your code.
- Data Collection: Record and display user input data effectively.
- Game Development Basics: Understand fundamental principles of creating interactive games.
- Advanced Game Mechanics: Explore more complex game logic and user interactions.
- Interactive Reading Apps: Develop applications that allow users to engage with content.
- Advanced Graphics with Canvas: Use canvas for sophisticated graphic designs.
- Shape Creation: Draw simple shapes using canvas functionality.
- Geometric Concepts: Combine shapes to understand their relationships.

## Level 3: Advanced Development

- Pattern Creation: Use loops to create repeating patterns and designs.
- Complex Shape Drawing: Learn to draw intricate shapes like stars and polygons.
- Nested Loops: Utilize nested loops for advanced shape creation and control.
- Drawing Applications: Build applications that allow users to create drawings.
- Health Mechanics in Games: Implement health systems in interactive games.
- Game Logic Implementation: Develop the logic behind user choices in games.
- Logic and Strategy Games: Design games that challenge users' logic and strategy skills.
- Keyboard Interactions: Incorporate keyboard events for character movement.
- Mobile Interaction: Learn to handle mobile device inputs for app control.
- Action Game Development: Create fast-paced, action-oriented games.
- Interactive Gameplay: Design engaging games with complex interaction mechanics.
- User Engagement: Explore concepts to keep users actively engaged in gameplay.

**Level 4: Practical Application Development**

- Obstacle Challenges: Create games that involve navigating through obstacles.
- Event-Specific Projects: Develop applications tailored for specific events or celebrations.
- User Input Applications: Build functional applications that generate outputs based on user inputs.