🛒 Flutter Shopping App Project

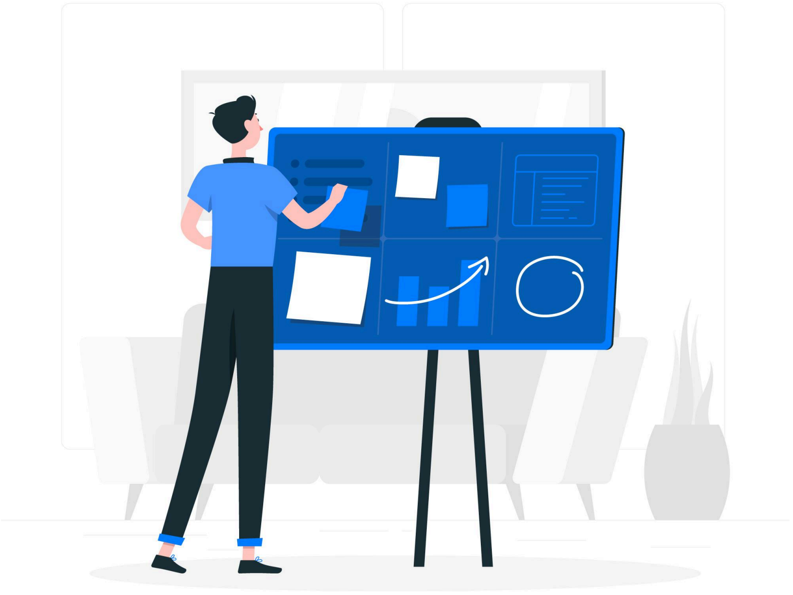
1. General Description of the Task:

This comprehensive Flutter project guides participants through the development of a fully functional **shopping application**, starting from core Dart programming principles and advancing to professional mobile app development with Flutter. The application simulates a real-world shopping interface and includes user authentication, product displays, cart interactions, and responsive UI.

Throughout the project, participants will incrementally apply key programming concepts including **variables, lists, maps, control flow, functions, and object-oriented programming** in Dart. As the app evolves, they will implement **Flutter UI widgets**, create **stateless and stateful interfaces**, manage **user input and validation**, and enable **page navigation** with transitions and animations.

The final stages of the project focus on **localizing the app into Arabic**, ensuring accessibility and internationalization readiness. Participants will also follow best practices by publishing their work to **GitHub**, using proper file structures, modular widgets, and detailed documentation.

This project provides a step-by-step path from beginner to intermediate Flutter developer while simulating the workflow of a real mobile app development lifecycle.



2. Objectives:

- Build a **complete shopping app** using **Dart and Flutter**.
- Implement core **Dart programming concepts**: variables, constants, control flow, functions, and OOP.
- Develop **Flutter UI** using Stateless and Stateful widgets.
- Add **input validation**, **navigation**, **animations**, and **localization (Arabic)**.
- Publish code to **GitHub**, adhering to best practices in structure, documentation, and styling.

3. Tools to be Used:

- **Dart**: For logic, functions, OOP.
- **Flutter SDK**: For app UI and features.
- **GitHub**: Public repo with README, screenshots, and code.
- **VS Code / Android Studio**: IDEs for development.
- **Flutter Intl**: For localization.
- **Command Line Tools**: For Flutter version checks and project generation.

4-Requirements:

1. Aesthetic Welcome Screen

- **Feature**: Static Intro Widget
  - **Displays:**
    - AppBar widget as a title app
    - Two images in a row (1 local image + 1 online image)
    - Use the following fonts:
      - Type: Suwannaphum-Regular
      - Do Custom font size, bold, colored
    - All elements centered with adequate spacing
    - Add two buttons:
      - Sign-up
      - Sign-in
- According the choice it leads to the second feature.

2. Sign Up/In to Access the App

- **Feature 2-A**: Sign-Up Form
  - Users are greeted with a sign-up form containing:
    - Full Name (First letter must be uppercase)
    - Email (Must include `@`)
    - Password (At least 6 characters)
    - Confirm Password (Must match password)
  - Valid submission opens a dialog: *"Account created successfully"*
  - A "Close" button then navigates the user to the main shopping screen.
- **Feature 2-B**: Sign-in Form:
  - Users are greeted with a sign-in form containing:
    - Email (Must include `@`)
    - Password (At least 6 characters)
  - Valid submission opens a dialog: *"Account sign-in successfully"*
  - A "Close" button then navigates the user to the main shopping screen (Feature 3)

3. Smooth Transition into the App

- **Feature**: Animated Navigation
    - After signing up, the Sign-Up page **fades out**
    - The Shopping Home page **fades in**, creating a seamless transition.

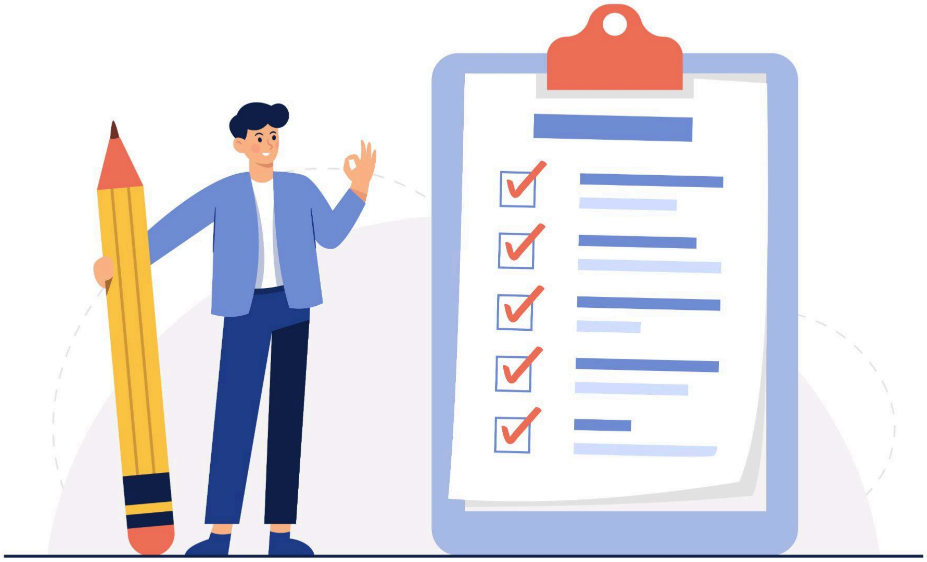
4. Explore the Shopping Home Screen

- **Feature**: Interactive Shopping Page
    - AppBar titled **“Our Products”**
    - A horizontal **PageView** shows featured product images.
    - Below, a responsive **GridView** shows product cards (2 per row):
        - Each card includes:
            - Product image
            - Product title
            - **Add to Cart icon**
              → shows **SnackBar**: *"Item added to the cart"*
    - Further down: **"Hot Offers"** section
        - Built using **ListView.builder** with 5 vertically scrollable offers.
        - Each item includes an image and a description using `Expanded`.

5. View the App in Arabic (Bonus)

- **Feature**: Arabic Language Support
  - The app's text content appears in **Arabic**:
    - “منتجاتنا” instead of “Our Products”
    - “العروض الساخنة” for “Hot Offers”
    - Sign-Up field labels and messages in Arabic
  - No hardcoded strings in UI code.
  - Uses `.arb` files and localization setup via `intl`.

UI Sample: https://aesthetic-flow-shop-intro.lovable.app/



🗂️ Repository & Submission Guidelines

- ✅ Upload project to a **public GitHub repo**
  - Username should match your real name
  - Include a `README.md` explaining:
    - Project overview
    - Features
    - Setup instructions
- ✅ Screenshot all outputs and add them to the repo
- ✅ Code Quality:
  - Each widget/class in a separate file
  - No unused or duplicated code
  - Clear variable/function/class names
  - Formatted and clean codebase