



# 155+ Innovative Flutter Project Ideas

[Leave a Comment](#) / [Computer Science](#)



Check out fun Flutter project ideas to help you learn mobile app development. Whether you're a beginner or already know some Flutter, these projects will help you create cool apps for iOS and Android!

Do you want to make mobile apps? Flutter is a great tool for building apps for both iOS and Android.

Where can you start? In this guide, we'll share some fun Flutter project ideas. These ideas will help you learn and get creative, whether you're a beginner or know some Flutter already.

Each project idea will tell you what you can do and what features to add. These projects will help you understand Flutter better and let you try things like using APIs and setting up logins.

So, if you're ready to jump into Flutter and make something cool, let's look at these project ideas!

### Table of Contents ☰ ↕

1. What is Flutter?
2. Benefits of Using Flutter
3. Setting Up the Flutter Environment
4. Flutter Project Ideas
5. Tips for Successful Flutter Projects
6. Flutter project Ideas With Source Code
7. Flutter Project Ideas for Final Year Students
8. Flutter Project Ideas Github
9. Flutter Project Ideas for Beginners
10. Conclusion

## What is Flutter?

Flutter is a free tool from Google that helps you build fast and beautiful apps for iOS, Android, web, and desktop. You write your app once, and it works on many devices.

Flutter uses an easy programming language called Dart. It has ready-made widgets to help you create designs quickly.

One great feature is hot reload, which lets you see changes right away without restarting the app.

With Flutter, you can make awesome apps that look good everywhere. It has a helpful community, so whether you're a beginner or know some coding, Flutter is great for making apps.

## Benefits of Using Flutter

Here are the benefits of using Flutter:

Feature	Description
<b>Cross-Platform</b>	Write your app once and use it on iOS, Android, web, and desktop.
<b>Fast Changes</b>	See updates right away with hot reload; no restart needed.
<b>Great Design</b>	Many ready-made widgets to make your app look good.
<b>Quick Performance</b>	Apps run fast because they use native code.
<b>Easy to Learn</b>	Dart, the programming language, is simple to understand.
<b>Helpful Community</b>	Lots of tutorials and support from other developers.
<b>Customizable</b>	Change your app's look easily with widgets.
<b>Easy Prototyping</b>	Quickly build and test ideas.
<b>Built-in Testing</b>	Easy tools to test your app.
<b>Free to Use</b>	Flutter is free and open-source for everyone.

## Setting Up the Flutter Environment

Here are the best ways for setting up the flutter environment:

Step	Description
<b>Download Flutter</b>	Go to the <a href="#">Flutter website</a> and download Flutter for your computer (Windows, macOS, or Linux).

Step	Description
<b>Unzip the Files</b>	Extract the downloaded file to a folder (like C:\flutter on Windows).
<b>Update Path</b>	Add the Flutter bin folder to your PATH. This allows you to run Flutter commands from any command line.
<b>Dart Included</b>	Dart is included with Flutter, so no need to install it separately.
<b>Check Your Setup</b>	Open a terminal (Command Prompt or Terminal) and type <code>flutter doctor</code> . This checks if everything is set up correctly.
<b>Install an IDE</b>	Download an IDE like Android Studio or Visual Studio Code to write your code.
<b>Set Up an Emulator</b>	If using Android Studio, create an Android Virtual Device (AVD) to test your apps.
<b>Add Flutter Plugins</b>	In your IDE, install the Flutter and Dart plugins for extra features.
<b>Create a New Project</b>	Use the command <code>flutter create project_name</code> in your terminal to make a new Flutter project.
<b>Run Your App</b>	Go to your project folder and type <code>flutter run</code> to see your app work!

Now you're ready to build apps with Flutter!

## Flutter Project Ideas

Here are some flutter project ideas:

### Beginner Projects

#### To-Do List App

- **Features:** Add, edit, delete tasks, mark as complete.
- **Tech:** Local storage using Shared Preferences.
- **UI:** Simple and clean layout with lists.

## Weather App

- **Features:** Show current weather, 7-day forecast, location search.
- **Tech:** Use a weather API (e.g., OpenWeatherMap).
- **UI:** Attractive icons and background images based on weather.

## Expense Tracker

- **Features:** Log expenses, categorize spending, monthly summaries.
- **Tech:** Local database using SQLite.
- **UI:** Charts for visual representation of spending.

## Note-Taking App

- **Features:** Create, edit, delete notes; organize by categories.
- **Tech:** Use local storage for saving notes.
- **UI:** Simple text editor with formatting options.

## Flashlight App

- **Features:** Toggle flashlight on/off.
- **Tech:** Use device flashlight API.
- **UI:** Simple button interface with feedback.

## Counter App

- **Features:** Increment, decrement, reset counter.
- **Tech:** State management using Provider.
- **UI:** Basic layout with large buttons.

## Unit Converter

- **Features:** Convert between various units (length, weight, etc.).
- **Tech:** Dynamic input fields for units.
- **UI:** Dropdown menus for selecting units.

## BMI Calculator

- **Features:** Input height and weight; calculate and display BMI.
- **Tech:** Use simple mathematical calculations.

- **UI:** Informative results with health categories.

## Random Color Generator

- **Features:** Generate random colors with display.
- **Tech:** Use color manipulation functions.
- **UI:** Show color codes and examples.

## Simple Calculator

- **Features:** Basic arithmetic operations.
- **Tech:** Use state management for operations.
- **UI:** Grid layout for buttons and display.

## Intermediate Projects

### Recipe App

- **Features:** Search recipes, save favorites, step-by-step instructions.
- **Tech:** Use a recipe API (e.g., Spoonacular).
- **UI:** Card layout for recipe previews.

### Chat Application

- **Features:** Real-time messaging, user profiles, group chats.
- **Tech:** Firebase for backend services.
- **UI:** Chat bubbles and user avatars.

### Fitness Tracker

- **Features:** Log workouts, track progress, set goals.
- **Tech:** Local database for storing workout data.
- **UI:** Progress charts and history lists.

### Quiz App

- **Features:** Multiple-choice questions, scoring system, timer.
- **Tech:** JSON for quiz data storage.
- **UI:** Engaging layout with question and answer buttons.

## Travel Guide App

- **Features:** Explore destinations, travel tips, save favorite places.
- **Tech:** Use Google Maps API for location services.
- **UI:** Map view and list view options.

## Movie Database App

- **Features:** Browse movies, search by genre, save favorites.
- **Tech:** Use The Movie Database API.
- **UI:** Grid view for movie posters.

## Recipe Recommendation App

- **Features:** Suggest recipes based on user input.
- **Tech:** Algorithm for matching ingredients.
- **UI:** Clean display of suggested recipes.

## Online Portfolio

- **Features:** Showcase projects, skills, and contact form.
- **Tech:** Use web services for hosting.
- **UI:** Professional layout with sections for each skill.

## Flashcard Study App

- **Features:** Create, edit, and review flashcards.
- **Tech:** Local storage for saving flashcards.
- **UI:** Simple interface for flipping cards.

## Barcode Scanner App

- **Features:** Scan barcodes and display information.
- **Tech:** Use camera and barcode scanning libraries.
- **UI:** Minimalist design for easy scanning.

See also [70 Astonishing INSPIRE Award Science Projects Ideas](#)

# Advanced Projects

## E-commerce App

- **Features:** User accounts, shopping cart, secure payments.
- **Tech:** Use Firebase for authentication and database.
- **UI:** Product grids and detailed product views.

## Social Media App

- **Features:** Post sharing, commenting, user profiles.
- **Tech:** Firebase for real-time data.
- **UI:** Engaging feed layout and profile customization.

## News Reader

- **Features:** Aggregator for news articles, save favorites.
- **Tech:** Use News API for fetching articles.
- **UI:** Card layouts for article previews.

## Music Player

- **Features:** Play music, create playlists, offline support.
- **Tech:** Local storage for songs.
- **UI:** Play/pause buttons and song information display.

## Real Estate Listing App

- **Features:** Browse properties, filter searches, save favorites.
- **Tech:** Use a real estate API for listings.
- **UI:** Map view with property markers.

## Event Management App

- **Features:** Organize events, RSVP, notifications.
- **Tech:** Local database for storing events.
- **UI:** Calendar view for events.

## Job Board



- **Features:** Job listings, application tracking, company profiles.
- **Tech:** Use job posting API.
- **UI:** Clean layout for job listings.

## Fitness App

- **Features:** Personalized workout plans, progress tracking.
- **Tech:** Use fitness tracking APIs.
- **UI:** Interactive workout sessions.

## Online Learning Platform

- **Features:** Courses with video lectures, quizzes, progress tracking.
- **Tech:** Use video hosting services for content.
- **UI:** Course catalog and user dashboards.

## Expense Sharing App

- **Features:** Log shared expenses, track balances among friends.
- **Tech:** Use local storage for expense records.
- **UI:** Clear lists for shared expenses.

## Educational Projects

### Language Learning App

- **Features:** Vocabulary quizzes, pronunciation guides.
- **Tech:** Use APIs for language data.
- **UI:** Interactive lessons with audio clips.

### Flashcard App

- **Features:** Create custom flashcards, spaced repetition.
- **Tech:** Local database for flashcards.
- **UI:** Simple card flip animations.

### Virtual Classroom

- **Features:** Live classes, video conferencing, quizzes.
- **Tech:** Use WebRTC for real-time communication.

- **UI:** Dashboard for class management.

## Math Quiz App

- **Features:** Random math problems, scoring, and timers.
- **Tech:** Generate questions dynamically.
- **UI:** Engaging quiz interface.

## Science Experiment Guide

- **Features:** Step-by-step experiments, safety tips.
- **Tech:** Use local storage for saving favorites.
- **UI:** Visual guides with images.

## History Timeline App

- **Features:** Interactive timeline of events.
- **Tech:** Use graphs or timeline libraries.
- **UI:** Scrollable timeline with event details.

## Spelling Bee App

- **Features:** Random word generation, scoring system.
- **Tech:** Local storage for tracking scores.
- **UI:** Engaging word display.

## Learning Game

- **Features:** Educational games for various subjects.
- **Tech:** Use game development principles.
- **UI:** Fun graphics and user interactions.

## Math Solver

- **Features:** Input problems and show step-by-step solutions.
- **Tech:** Use mathematical libraries.
- **UI:** Simple input fields and output display.

## Study Planner

- **Features:** Create study schedules, set reminders.
- **Tech:** Local storage for schedules.
- **UI:** Calendar view for organizing study sessions.

## Utility Projects

### Password Manager

- **Features:** Securely store and encrypt passwords.
- **Tech:** Use encryption libraries for security.
- **UI:** Simple input forms for adding passwords.

### Habit Tracker

- **Features:** Track daily habits, visual progress.
- **Tech:** Local storage for tracking data.
- **UI:** Calendar view for habit completion.

### Reminder App

- **Features:** Set reminders for tasks and events.
- **Tech:** Use local notifications for alerts.
- **UI:** Simple list for upcoming reminders.

### Unit Converter

- **Features:** Convert various units dynamically.
- **Tech:** Use dropdown menus for selection.
- **UI:** Clean input fields for entering values.

### Document Scanner

- **Features:** Capture images, convert to PDF.
- **Tech:** Use camera and PDF libraries.
- **UI:** Simple interface for scanning.

### File Manager

- **Features:** Browse files, create folders, delete files.
- **Tech:** Use file system APIs.

- **UI:** List view for files and folders.

## URL Shortener

- **Features:** Shorten long URLs and track clicks.
- **Tech:** Use URL shortening API.
- **UI:** Simple input field for URLs.

## Voice Recorder

- **Features:** Record audio notes and save files.
- **Tech:** Use audio recording libraries.
- **UI:** Simple buttons for recording and playback.

## Calendar App

- **Features:** View, add, and manage events.
- **Tech:** Local storage for event data.
- **UI:** Month and week view options.

## Image Gallery

- **Features:** Browse and manage images on the device.
- **Tech:** Use image picker libraries.
- **UI:** Grid layout for displaying images.

## Fun and Creative Projects

### Photo Filter App

- **Features:** Apply filters and effects to images.
- **Tech:** Use image processing libraries.
- **UI:** Interactive image editing interface.

### Meme Generator

- **Features:** Create memes from images and text.
- **Tech:** Use local storage for saving memes.
- **UI:** Fun layout for meme creation.

## Random Joke Generator

- **Features:** Fetch and display random jokes.
- **Tech:** Use a jokes API.
- **UI:** Simple button to get a new joke.

## Virtual Pet

- **Features:** Feed, play, and take care of a virtual pet.
- **Tech:** Local storage for pet stats.
- **UI:** Cute graphics and animations.

## Augmented Reality App

- **Features:** View 3D models in real-world environments.
- **Tech:** Use AR libraries like ARCore.
- **UI:** Interactive AR experience.

## Custom Sticker Maker

- **Features:** Create and save custom stickers.
- **Tech:** Use image manipulation libraries.
- **UI:** Simple editing tools for stickers.

## Mood Tracker

- **Features:** Log daily moods and track changes.
- **Tech:** Use local storage for tracking data.
- **UI:** Color-coded mood selections.

## Voice Assistant

- **Features:** Voice commands for various tasks.
- **Tech:** Use speech recognition libraries.
- **UI:** Minimalist interface for interaction.

## Storytelling App

- **Features:** Create and share short stories.
- **Tech:** Local storage for stories.

- **UI:** Engaging text editor for writing.

## Interactive Quiz Game

- **Features:** Trivia questions with scoring.
- **Tech:** Use JSON for questions.
- **UI:** Fun interface for gameplay.

## Health and Fitness Projects

### Personal Health Tracker

- **Features:** Log symptoms, medications, and health metrics.
- **Tech:** Use local storage for health records.
- **UI:** Dashboard for health overview.

### Meditation Timer

- **Features:** Set timers for meditation sessions.
- **Tech:** Local notifications for reminders.
- **UI:** Simple timer interface.

### Diet Planner

- **Features:** Plan meals and track calorie intake.
- **Tech:** Use local storage for meal data.
- **UI:** Meal calendar and logging options.

### Fitness Challenge App

- **Features:** Set fitness challenges and track progress.
- **Tech:** Use local storage for challenge data.
- **UI:** Progress tracking and leaderboards.

### Step Counter

- **Features:** Count steps and track daily activity.
- **Tech:** Use pedometer APIs.
- **UI:** Daily step display and goals.

## Health Tips App

- **Features:** Daily health tips and reminders.
- **Tech:** Use local notifications for alerts.
- **UI:** Simple layout for tips.

## Yoga Pose Guide

- **Features:** Display yoga poses with instructions.
- **Tech:** Use image and video resources.
- **UI:** Attractive visuals for poses.

## Calorie Counter

- **Features:** Log food intake and track calories.
- **Tech:** Use food database APIs.
- **UI:** Searchable food database.

## Workout Routine Generator

- **Features:** Create custom workout routines.
- **Tech:** Use local storage for saving routines.
- **UI:** Simple interface for routine creation.

See also [149+ Best Capstone Project Ideas for STEM Students](#)

## Mood and Energy Tracker

- **Features:** Log moods and energy levels daily.
- **Tech:** Use local storage for tracking data.
- **UI:** Graphical representation of moods.

## Social Impact Projects

### Community Service Finder

- **Features:** Find local volunteer opportunities.
- **Tech:** Use local databases for listings.
- **UI:** Map view of opportunities.

## Donation Tracker

- **Features:** Log donations and track impacts.
- **Tech:** Local storage for donation records.
- **UI:** Visual representation of donation impact.

## Mental Health Resources

- **Features:** Provide information on mental health services.
- **Tech:** Use a resource API.
- **UI:** Simple layout for resource listings.

## Environmental Awareness App

- **Features:** Share tips on sustainability and recycling.
- **Tech:** Local storage for tracking eco-friendly habits.
- **UI:** Engaging graphics for tips.

## Pet Adoption App

- **Features:** Browse pets available for adoption.
- **Tech:** Use APIs from shelters for listings.
- **UI:** Card layout for pet profiles.

## Crisis Management App

- **Features:** Provide emergency contacts and resources.
- **Tech:** Use local storage for quick access.
- **UI:** Clear interface for emergencies.

## Food Donation App

- **Features:** Connect donors with food banks.
- **Tech:** Use APIs for logistics tracking.
- **UI:** Simple layout for connecting donors.

## Charity Event Organizer

- **Features:** Create and manage charity events.
- **Tech:** Local storage for event data.



- **UI:** Calendar view for events.

## Community Feedback App

- **Features:** Collect and analyze community feedback.
- **Tech:** Use local databases for feedback storage.
- **UI:** Survey forms for feedback.

## Awareness Campaign App

- **Features:** Share information on social issues.
- **Tech:** Use social media integration for sharing.
- **UI:** Attractive layout for campaigns.

## Business and Finance Projects

### Stock Market Tracker

- **Features:** Monitor stock prices and trends.
- **Tech:** Use stock market APIs for live data.
- **UI:** Graphical representation of stock performance.

### Invoice Generator

- **Features:** Create and manage invoices.
- **Tech:** Local storage for saving invoices.
- **UI:** Professional invoice layout.

### Budgeting App

- **Features:** Track income and expenses, set budgets.
- **Tech:** Use local storage for financial data.
- **UI:** Charts and summaries for budgeting.

### Expense Reporting App

- **Features:** Log and categorize expenses for reporting.
- **Tech:** Use local storage for expense records.
- **UI:** Simple forms for logging expenses.

## Personal Finance Tracker

- **Features:** Track financial goals and savings.
- **Tech:** Use local storage for savings records.
- **UI:** Visual representation of financial goals.

## Shopping List Manager

- **Features:** Create and manage grocery lists.
- **Tech:** Use local storage for lists.
- **UI:** Simple layout for list management.

## Currency Converter

- **Features:** Convert between currencies with live rates.
- **Tech:** Use currency conversion APIs.
- **UI:** Dropdown menus for currency selection.

## Sales Dashboard

- **Features:** Monitor sales data and trends.
- **Tech:** Use local storage for sales data.
- **UI:** Visual charts and graphs.

## Freelance Project Tracker

- **Features:** Manage freelance projects and invoicing.
- **Tech:** Use local storage for project data.
- **UI:** Simple layout for project management.

## Time Tracking App

- **Features:** Log time spent on tasks/projects.
- **Tech:** Use local storage for time records.
- **UI:** Dashboard for time summary.

## Learning and Education Projects

### Language Learning App

- **Features:** Vocabulary quizzes and flashcards.
- **Tech:** Use local storage for progress tracking.
- **UI:** Interactive quiz interface.

## Study Planner

- **Features:** Organize study schedules and tasks.
- **Tech:** Local storage for schedules.
- **UI:** Calendar view for tasks.

## E-book Reader

- **Features:** Read and bookmark e-books.
- **Tech:** Use file management for e-book storage.
- **UI:** Simple reading interface with bookmarks.

## Online Quiz App

- **Features:** Create and take quizzes.
- **Tech:** Use local storage for quiz data.
- **UI:** Fun interface for quiz taking.

## Flashcard App

- **Features:** Create and study flashcards.
- **Tech:** Use local storage for card data.
- **UI:** Simple interface for flipping cards.

## Virtual Classroom

- **Features:** Host and join online classes.
- **Tech:** Use video conferencing APIs.
- **UI:** User-friendly interface for classrooms.

## Research Organizer

- **Features:** Collect and organize research notes.
- **Tech:** Use local storage for notes.
- **UI:** Simple layout for note-taking.

## Grade Tracker

- **Features:** Log and track grades for courses.
- **Tech:** Use local storage for grade data.
- **UI:** Graphical representation of grades.

## Homework Reminder

- **Features:** Set reminders for homework assignments.
- **Tech:** Local notifications for alerts.
- **UI:** Simple reminder interface.

## Science Experiment Logger

- **Features:** Document and track science experiments.
- **Tech:** Use local storage for experiment data.
- **UI:** Organized layout for logging experiments.

# Tips for Successful Flutter Projects

Here are the tips for successful flutter projects:

## Best Practices in Flutter Development

### Write Clean Code

Keep your code simple and easy to read.

### Use State Management Wisely

Choose the best state management method for your project (like Provider, Riverpod, or Bloc).

### Optimize Performance

Monitor your app's speed and improve it when needed.

## Common Pitfalls and How to Avoid Them

### Overcomplicating the UI

Make your user interface simple and easy to use.

## Neglecting Testing

Test your app regularly to find and fix bugs.

## Ignoring Updates

Keep your Flutter SDK and dependencies updated to use the latest features.

# Resources for Continuous Learning

## Flutter Documentation

The official documentation is a great resource for learning.

## Online Courses

Websites like Udemy, Coursera, and Codecademy have Flutter courses.

## Community Forums

Join the Flutter community on forums like Stack Overflow and Reddit for help and ideas.

# Flutter project Ideas With Source Code

Here are some of the best flutter project ideas with source code:

## Todo List App

**Description:** A simple app to manage daily tasks.

### Source Code

```
import 'package:flutter/material.dart';

void main() => runApp(TodoApp());

class TodoApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
```

```

        home: TodoHome(),
    );
}
}

class TodoHome extends StatefulWidget {
  @override
  _TodoHomeState createState() => _TodoHomeState();
}

class _TodoHomeState extends State<TodoHome> {
  final List<String> _todos = [];
  final TextEditingController _controller = TextEditingController();

  void _addTodo() {
    setState(() {
      _todos.add(_controller.text);
      _controller.clear();
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Todo List')),
      body: Column(
        children: [
          Expanded(
            child: ListView.builder(
              itemCount: _todos.length,
              itemBuilder: (context, index) => ListTile(title: Text(_to
            ),
          ),
          TextField(
            controller: _controller,
            decoration: InputDecoration(labelText: 'Add a todo'),
          ),
          ElevatedButton(onPressed: _addTodo, child: Text('Add')),
        ],
      ),
    );
  }
}

```

# Weather App

**Description:** Displays current weather data for a given location.

## Source Code

```
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
import 'dart:convert';

void main() => runApp(WeatherApp());

class WeatherApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(home: WeatherHome());
  }
}

class WeatherHome extends StatefulWidget {
  @override
  _WeatherHomeState createState() => _WeatherHomeState();
}

class _WeatherHomeState extends State<WeatherHome> {
  String _weather = '';

  Future<void> _fetchWeather() async {
    final response = await http.get(Uri.parse('https://api.openweathermap.org/data/2.5/weather?q=London&appid=669c089730874179876a23436c81888'));
    final data = json.decode(response.body);
    setState(() {
      _weather = data['weather'][0]['description'];
    });
  }

  @override
  void initState() {
    super.initState();
    _fetchWeather();
  }
}
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: Text('Weather App')),
    body: Center(child: Text(_weather)),
  );
}
}

```

## Quiz App

**Description:** A simple quiz app with multiple-choice questions.

See also [331+ Top-Listed & Best Design Engineering Project Ideas](#)

### Source Code

```

import 'package:flutter/material.dart';

void main() => runApp(QuizApp());

class QuizApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(home: QuizHome());
  }
}

class QuizHome extends StatefulWidget {
  @override
  _QuizHomeState createState() => _QuizHomeState();
}

class _QuizHomeState extends State<QuizHome> {
  final List<Map<String, dynamic>> _questions = [
    {'question': 'What is 2 + 2?', 'answers': ['3', '4', '5'], 'correct': '4'},
  ];
  int _currentQuestion = 0;

  void _nextQuestion() {
    setState(() {

```



```

        _currentQuestion = (_currentQuestion + 1) % _questions.length;
    });
}

@override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(title: Text('Quiz App')),
        body: Column(
            children: [
                Text(_questions[_currentQuestion]['question']),
                ...(_questions[_currentQuestion]['answers'] as List<String>).
                    return ElevatedButton(
                        onPressed: _nextQuestion,
                        child: Text(answer),
                    );
            ]).toList(),
        ],
    ),
);
}
}

```

## BMI Calculator

**Description:** Calculates Body Mass Index based on user input.

### Source Code

```

import 'package:flutter/material.dart';

void main() => runApp(BmiCalculator());

class BmiCalculator extends StatelessWidget {
    @override
    Widget build(BuildContext context) {
        return MaterialApp(home: BmiHome());
    }
}

class BmiHome extends StatefulWidget {
    @override

```

```

    _BmiHomeState createState() => _BmiHomeState();
  }

class _BmiHomeState extends State<BmiHome> {
  final TextEditingController _weightController = TextEditingController();
  final TextEditingController _heightController = TextEditingController();
  String _result = '';

  void _calculateBmi() {
    final double weight = double.parse(_weightController.text);
    final double height = double.parse(_heightController.text) / 100;
    final double bmi = weight / (height * height);
    setState(() {
      _result = 'Your BMI is ${bmi.toStringAsFixed(1)}';
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('BMI Calculator')),
      body: Column(
        children: [
          TextField(controller: _weightController, decoration: InputDecoration()),
          TextField(controller: _heightController, decoration: InputDecoration()),
          ElevatedButton(onPressed: _calculateBmi, child: Text('Calculate')),
          Text(_result),
        ],
      ),
    );
  }
}

```

## Note-taking App

**Description:** A simple app for users to create and save notes.

Source Code

```

import 'package:flutter/material.dart';

void main() => runApp(NoteApp());

```

```

class NoteApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(home: NoteHome());
  }
}

class NoteHome extends StatefulWidget {
  @override
  _NoteHomeState createState() => _NoteHomeState();
}

class _NoteHomeState extends State<NoteHome> {
  final List<String> _notes = [];
  final TextEditingController _controller = TextEditingController();

  void _addNote() {
    setState(() {
      _notes.add(_controller.text);
      _controller.clear();
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Notes')),
      body: Column(
        children: [
          Expanded(
            child: ListView.builder(
              itemCount: _notes.length,
              itemBuilder: (context, index) => ListTile(title: Text(_no
            ),
          ),
          TextField(controller: _controller, decoration: InputDecoratio
          ElevatedButton(onPressed: _addNote, child: Text('Add')),
        ],
      ),
    );
  }
}

```

# Flutter Project Ideas for Final Year Students

Here are some easy Flutter project ideas for final year students:

## E-commerce App

- **Purpose:** Users can browse and buy products.
- **Features:** Login, product categories, and payment options.

## Social Media App

- **Purpose:** Users create profiles and share updates.
- **Features:** Likes, comments, and notifications.

## Health Tracker App

- **Purpose:** Users track fitness and health goals.
- **Features:** Step counting and calorie tracking.

## Online Learning App

- **Purpose:** Users enroll in courses and watch videos.
- **Features:** Profiles and progress tracking.

## Recipe App

- **Purpose:** Users find and save recipes.
- **Features:** Ingredient search and meal types.

## Expense Tracker App

- **Purpose:** Users track spending and budgeting.
- **Features:** Expense categories and charts.

## Travel Planner App

- **Purpose:** Helps users plan trips with hotels and activities.
- **Features:** Itineraries and maps.

## Event Management App

- **Purpose:** Users create and RSVP to events.
- **Features:** Guest lists and notifications.

## Job Portal App

- **Purpose:** Connects job seekers with employers.
- **Features:** Job listings and applications.

## Mental Health App

- **Purpose:** Provides mental health support and resources.
- **Features:** Self-help tools and community forums.

## Music Streaming App

- **Purpose:** Users listen to music and create playlists.
- **Features:** Search and offline access.

## Flashcard Study App

- **Purpose:** Helps students study with flashcards.
- **Features:** Quiz modes and sharing.

## Fitness Challenge App

- **Purpose:** Users join fitness challenges and track progress.
- **Features:** Sharing and rewards.

## Book Review App

- **Purpose:** Users browse and review books.
- **Features:** Ratings and reading lists.

## Smart Home Controller

- **Purpose:** Controls smart home devices.
- **Features:** Remote access and management.

## Flutter Project Ideas Github

Here are some simple Flutter project ideas along with GitHub links where you can find source code and examples:

### E-commerce App

- **Purpose:** Users can browse and buy products.
- **GitHub Link:** [E-commerce App](#)

### Social Media App

- **Purpose:** Users create profiles and share updates.
- **GitHub Link:** [Social Media App](#)

### Health Tracker App

- **Purpose:** Users track fitness and health goals.
- **GitHub Link:** [Health Tracker App](#)

### Online Learning App

- **Purpose:** Users enroll in courses and watch videos.
- **GitHub Link:** [Online Learning App](#)

### Recipe App

- **Purpose:** Users find and save recipes.
- **GitHub Link:** [Recipe App](#)

### Expense Tracker App

- **Purpose:** Users track spending and budgeting.

- **GitHub Link:** [Expense Tracker App](#)

## Travel Planner App

- **Purpose:** Helps users plan trips with hotels and activities.
- **GitHub Link:** [Travel Planner App](#)

## Event Management App

- **Purpose:** Users create and RSVP to events.
- **GitHub Link:** [Event Management App](#)

## Job Portal App

- **Purpose:** Connects job seekers with employers.
- **GitHub Link:** [Job Portal App](#)

## Mental Health App

- **Purpose:** Provides mental health support and resources.
- **GitHub Link:** [Mental Health App](#)

## Music Streaming App

- **Purpose:** Users listen to music and create playlists.
- **GitHub Link:** [Music Streaming App](#)

## Flashcard Study App

- **Purpose:** Helps students study with flashcards.
- **GitHub Link:** [Flashcard Study App](#)

## Fitness Challenge App

- **Purpose:** Users join fitness challenges and track progress.
- **GitHub Link:** [Fitness Challenge App](#)

## Book Review App

- **Purpose:** Users browse and review books.
- **GitHub Link:** [Book Review App](#)

## Smart Home Controller

- **GitHub Link:** [Smart Home Controller](#)
- **Purpose:** Controls smart home devices.

## Flutter Project Ideas for Beginners

Here are some simple Flutter project ideas for beginners:

### To-Do List App

- **Purpose:** Users can add, remove, and check off tasks.
- **Features:** Simple UI with task input, a list view, and task completion status.

### Weather App

- **Purpose:** Shows current weather conditions.
- **Features:** Fetches data from a weather API, displays temperature, humidity, and weather icons.

### Calculator App

- **Purpose:** Performs basic arithmetic operations.
- **Features:** Simple interface with buttons for numbers and operations.

### Notes App

- **Purpose:** Users can create and save notes.
- **Features:** Add, edit, and delete notes with a simple text input field.

### Quiz App

- **Purpose:** Users can take quizzes on various topics.



- **Features:** Multiple-choice questions, score tracking, and feedback.

## Flashlight App

- **Purpose:** Turns on the phone's flashlight.
- **Features:** Simple toggle button to switch the light on and off.

## Currency Converter

- **Purpose:** Converts one currency to another.
- **Features:** Input fields for amount, selection of currencies, and conversion logic.

## BMI Calculator

- **Purpose:** Calculates Body Mass Index.
- **Features:** Input for height and weight, shows the BMI result and category.

## Recipe App

- **Purpose:** Users can find and save recipes.
- **Features:** List of recipes with ingredients and steps.

## Random Quote Generator

- **Purpose:** Displays a random quote each time the user taps a button.
- **Features:** Pulls quotes from a predefined list or an API.

## Image Gallery App

- **Purpose:** Displays a collection of images.
- **Features:** Grid view of images with tap-to-view functionality.

## Timer App

- **Purpose:** Users can set timers.
- **Features:** Simple UI to input time and start/stop functionality.

## Expense Tracker

- **Purpose:** Helps users track daily expenses.
- **Features:** Input for amounts and categories, and displays a summary.

## Simple Chat App

- **Purpose:** Basic chat interface for users.
- **Features:** Send and receive messages with a simple UI.

## Habit Tracker

- **Features:** Input for habits, check off completion, and view progress.
- **Purpose:** Helps users track habits.

## Conclusion

In conclusion, Flutter is a simple tool for making apps that look good and work on many devices. You can write your code once and use it everywhere, which is great!

The project ideas shared are perfect for beginners. Each one helps you learn important skills, like using widgets and getting data from the internet. For example, making a To-Do List app teaches you how to take user input, while a Weather app shows you how to get online data.

As you build these projects, you'll understand Flutter better. The more you practice, the easier it gets!

So, pick a project you like, start coding, and have fun! If you need help, ask the friendly Flutter community. Enjoy creating your apps!

[← Previous Post](#)

[Next Post →](#)

## Related Posts



## 99+ Best & Trending Azure Project Ideas for CS Students

[Leave a Comment](#) / [Computer Science](#) / [By Adam Tesla](#)



## 169+ reMarkable Solidworks Project Ideas for Engineering Students

[Leave a Comment](#) / [Computer Science](#) / [By Adam Tesla](#)

## Leave a Comment

Your email address will not be published. Required fields are marked \*

Type here..

Name\*

Email\*

Save my name, email, and website in this browser for the next time I comment.

[Post Comment »](#)

## Recent Posts

181+ Fun and Creative Hydrology Projects for Students

191+ Creative Simple Slope Project Ideas for Students

161+ Easy and Engaging Science Projects for Class 7

151+ Fun & Engaging Science Project Ideas for Kids

189+ Engaging Turkey Project Ideas for Student Learning

## Categories

[Computer Science](#)

[General](#)

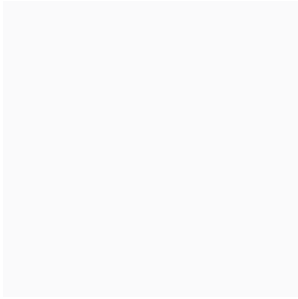
[Humanities](#)

[Mini](#)

## Subscribe to Our Newsletter

Subscribe us for latest project ideas on all subjects into your email.

[Subscribe](#)



## Top Pages

- [Privacy Policy](#)
- [Disclaimer](#)
- [Terms And Conditions](#)

## Top Categories

- [Computer Science](#)
- [General](#)
- [Humanities](#)
- [Mini](#)

## Follow us on

