Mini project - TODO list

This project does not require fetching data from the Internet or store something in a database. To keep it simple, we will use local objects.

# Object Model

The application uses a Todo class that has two attributes but no method (remember that we do not ask you to use a backend or a database):

|  |
| --- |
| **Todo** |
| **title**: String |
| **isDone**: bool |

# Description of the Application

## Requirements

The purpose of the application is to display a list of tasks to do and to manage these tasks:

* Display the list of tasks and if they are done or not yet.
* User can add a new task to do.
* User can mark a task as done.

## User Interface

### List of tasks

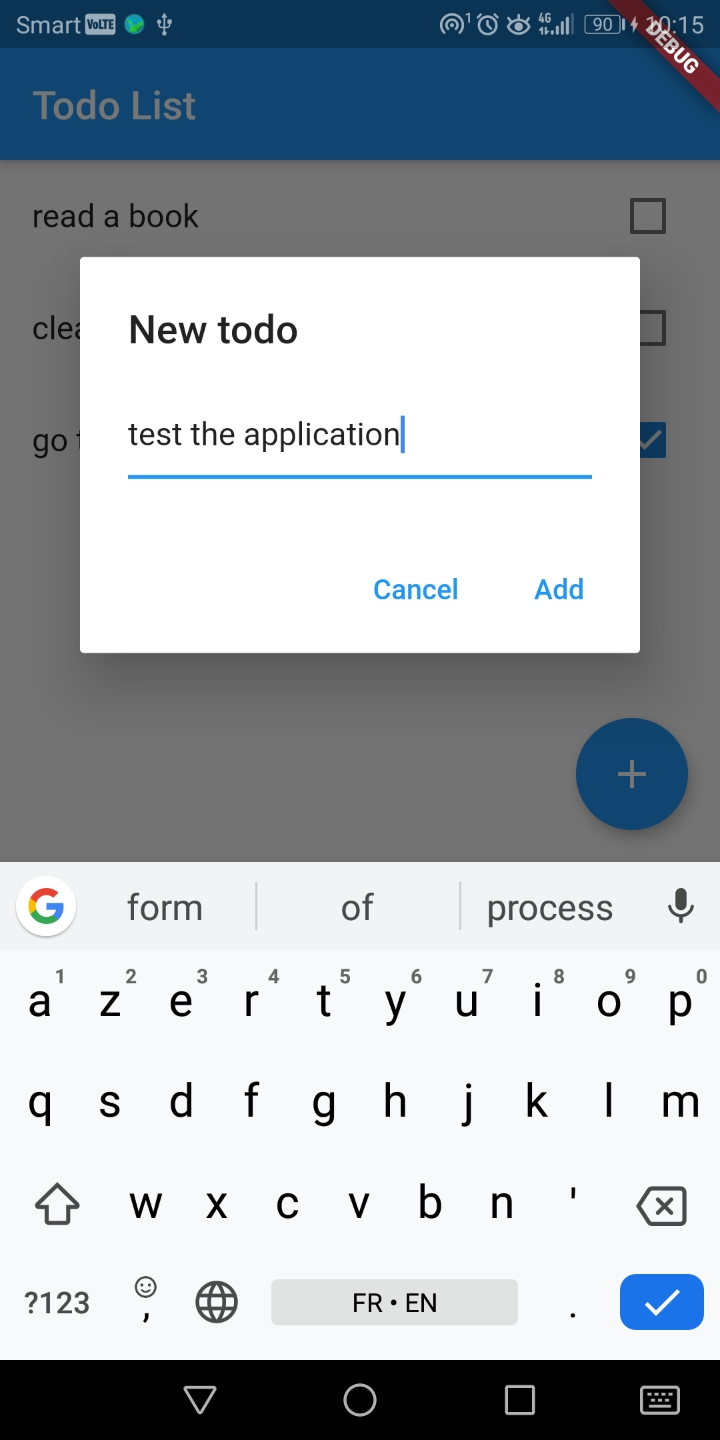
This is the main screen of the application. It displays the list of tasks. On this screen, you can:

* Mark any task as done by touching the square box.
* Add a new task by clicking on the floating button.

In order to develop this screen, you need to read the following documentation:

* Build a list of checkbox items: <https://api.flutter.dev/flutter/material/CheckboxListTile-class.html>
* Use a floating action button: <https://api.flutter.dev/flutter/material/FloatingActionButton-class.html>

### Add a new task



When the user clicks on the floating button to add a new task a dialog appears and the user has to enter the title of the task.

In order to develop this screen, you need to read the following documentation:

* How to use AlertDialog widget: <https://api.flutter.dev/flutter/material/AlertDialog-class.html>
* Here the dialog will allow you to enter data, so the Widget will not be a simple Text widget but a TextField widget: <https://api.flutter.dev/flutter/material/TextField-class.html>
* And if you want to get the data from the TextField widget, you need to set the controller property of the widget. You can read the documentation about TextEditingController. But in order to help you, what you need to do is when the user clicks on the Add button, is to get the value (of the TextField widget) like this (you must set up the controller properly):

|  |
| --- |
| final todo = new Todo(title: controller.value.text);  controller.clear(); |