Task Management Application Using Flutter

1.Introduction

In today's fast-paced world, effective time and task management are crucial for personal productivity and organizational success. With the increasing complexity of daily responsibilities, individuals and teams require efficient tools to help them prioritize and track their tasks. Task management applications serve as vital solutions that allow users to organize their workloads, set deadlines, and monitor progress.

This report presents a task management application developed using Flutter, a popular open-source UI software development toolkit. Flutter allows developers to create natively compiled applications for mobile, web, and desktop from a single codebase, making it an ideal choice for cross-platform development.

The application is designed with a focus on usability, providing a seamless experience for users to create, manage, and complete their tasks. It features a clean and intuitive interface, allowing users to easily navigate between different functionalities such as adding new tasks, marking tasks as completed, and viewing a list of all tasks. By leveraging a local database (SQLite), the application ensures that user data is stored securely and is readily accessible even when the application is closed.

2.Objective

The primary objectives of the task management application are:

- **User-Friendly Interface**: Create an intuitive and easy-to-navigate interface for users.
- **Task Management**: Allow users to create, view, update, and delete tasks efficiently.
- **Local Database Storage**: Utilize a local database (SQLite) to store tasks persistently.

- **Task Status Tracking**: Enable users to mark tasks as completed or pending.
- **Cross-Platform Compatibility**: Ensure the application functions well on both Android and iOS platforms.

3. Modules Identified

- Main. dart
- Task dashboard. dart
- Calendar View
- Create Task Screen
- Setting Deadline for the break

4. Code

- Main. dart

```
import 'package:first_project/models/task.dart';
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import '../providers/task_provider.dart';
import '../widgets/task_card.dart';
import 'create_task.dart';
import 'calendar_view.dart';
import 'notifications.dart';
import 'user_profile.dart';
import 'settings.dart';

class TaskDashboardScreen extends StatelessWidget {
  const TaskDashboardScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Task Management App'),
        backgroundColor: Colors.lightBlue,
      ),
      drawer: Drawer(
        child: ListView(
          padding: EdgeInsets.zero,
          children: [
            const DrawerHeader(
              decoration: BoxDecoration(
                color: Colors.blue,
              ),
              child: Text(
                'Welcome',
                style: TextStyle(fontSize: 24, color: Colors.white),
              ),
            ),
            ListTile(
              leading: const Icon(Icons.calendar_today),
              title: const Text('Calendar'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
CalendarViewScreen()),
                );
              },
            ),
            ListTile(
              leading: const Icon(Icons.notifications),
              title: const Text('Notifications'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
NotificationsScreen()),
                );
              },
            ),
            ListTile(
              leading: const Icon(Icons.person),
              title: const Text('User Profile'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
UserProfileScreen()),
                );
              },
            ),
            ListTile(
              leading: const Icon(Icons.settings),
              title: const Text('Settings'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
SettingsScreen()),
                );
              },
            ),
          ],
        ),
      ),
      body: Container(
        decoration: const BoxDecoration(
          image: DecorationImage(
            image: AssetImage('assets/background.jpg'),
            fit: BoxFit.cover,
          ),
        ),
        child: Consumer<TaskProvider>(
          builder: (context, taskProvider, child) {
            return ListView.builder(
              itemCount: taskProvider.tasks.length,
              itemBuilder: (context, index) {
                return TaskCard(
                  task: taskProvider.tasks[index],
                  onDelete: () {
                    taskProvider.removeTask(taskProvider.tasks[index].id);
// Delete functionality
                  },
                  onToggleCompletion: (isChecked) {

taskProvider.toggleTaskCompletion(taskProvider.tasks[index].id); // Update
completion status
                  },
                );
              },
            );
          },
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {
          Navigator.push(
            context,
            MaterialPageRoute(
              builder: (context) => CreateTaskScreen(
                onSave: (title, description, deadline, isCompleted,
isPending) {
                  final task = Task(
                    id: DateTime.now().toString(),
                    title: title,
                    description: description,
                    deadline: deadline,
                    isCompleted: isCompleted,
                    isPending: isPending,
                  );
                  Provider.of<TaskProvider>(context, listen:
false).addTask(task);
                  Navigator.pop(context); // Close the form
                },
              ),
            ),
          );
        },
        backgroundColor: Colors.lightBlue,
        child: const Icon(Icons.add),
      ),
    );
  }
}
```

- **Task_ dashboard. dart**

```
import 'package:first_project/models/task.dart';
import 'package:flutter/material.dart';
import 'package:provider/provider.dart';
import '../providers/task_provider.dart';
import '../widgets/task_card.dart';
import 'create_task.dart';
import 'calendar_view.dart';
import 'notifications.dart';
import 'user_profile.dart';
import 'settings.dart';

class TaskDashboardScreen extends StatelessWidget {
  const TaskDashboardScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Task Management App'),
        backgroundColor: Colors.lightBlue,
      ),
      drawer: Drawer(
        child: ListView(
          padding: EdgeInsets.zero,
          children: [
            const DrawerHeader(
              decoration: BoxDecoration(
                color: Colors.blue,
              ),
              child: Text(
                'Welcome',
                style: TextStyle(fontSize: 24, color: Colors.white),
              ),
            ),
            ListTile(
              leading: const Icon(Icons.calendar_today),
              title: const Text('Calendar'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
CalendarViewScreen()),
                );
              },
            ),
            ListTile(
              leading: const Icon(Icons.notifications),
              title: const Text('Notifications'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
NotificationsScreen()),
                );
              },
            ),
            ListTile(
              leading: const Icon(Icons.person),
              title: const Text('User Profile'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
UserProfileScreen()),
                );
              },
            ),
            ListTile(
              leading: const Icon(Icons.settings),
              title: const Text('Settings'),
              onTap: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => const
SettingsScreen()),
                );
              },
            ),
          ],
        ),
      ),
      body: Container(
        decoration: const BoxDecoration(
          image: DecorationImage(
            image: AssetImage('assets/background.jpg'),
            fit: BoxFit.cover,
          ),
        ),
        child: Consumer<TaskProvider>(
          builder: (context, taskProvider, child) {
            return ListView.builder(
              itemCount: taskProvider.tasks.length,
              itemBuilder: (context, index) {
                return TaskCard(
                  task: taskProvider.tasks[index],
                  onDelete: () {
                    taskProvider.removeTask(taskProvider.tasks[index].id);
// Delete functionality
                  },
                  onToggleCompletion: (isChecked) {

taskProvider.toggleTaskCompletion(taskProvider.tasks[index].id); // Update
completion status
                  },
                );
              },
            );
          },
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {
          Navigator.push(
            context,
            MaterialPageRoute(
              builder: (context) => CreateTaskScreen(
                onSave: (title, description, deadline, isCompleted,
isPending) {
                  final task = Task(
                    id: DateTime.now().toString(),
                    title: title,
                    description: description,
                    deadline: deadline,
                    isCompleted: isCompleted,
                    isPending: isPending,
                  );
                  Provider.of<TaskProvider>(context, listen:
false).addTask(task);
                  Navigator.pop(context); // Close the form
                },
              ),
            ),
          );
        },
        backgroundColor: Colors.lightBlue,
        child: const Icon(Icons.add),
      ),
    );
  }
}
```

- **Calendar View**

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

class CalendarViewScreen extends StatefulWidget {
  const CalendarViewScreen({super.key});

  @override
  CalendarViewScreenState createState() => CalendarViewScreenState();
}

class CalendarViewScreenState extends State<CalendarViewScreen> {
  late final ValueNotifier<DateTime> _focusedDay;
  late DateTime _selectedDay;

  @override
  void initState() {
    super.initState();
    _focusedDay = ValueNotifier(DateTime.now());
    _selectedDay = DateTime.now();
  }

  @override
  void dispose() {
    _focusedDay.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Calendar View'),
      ),
      body: Padding(
        padding: const EdgeInsets.all(8.0),
        child: Column(
          children: [
            TableCalendar(
              firstDay: DateTime.utc(2010, 10, 16),
              lastDay: DateTime.utc(2030, 3, 14),
              focusedDay: _focusedDay.value,
              selectedDayPredicate: (day) {
                return isSameDay(_selectedDay, day);
              },
              onDaySelected: (selectedDay, focusedDay) {
                setState(() {
                  _selectedDay = selectedDay;
                  _focusedDay.value = focusedDay; // update `_focusedDay`
here as well
                });
              },
              calendarStyle: const CalendarStyle(
                todayDecoration: BoxDecoration(
                  color: Colors.blue,
                  shape: BoxShape.circle,
                ),
                selectedDecoration: BoxDecoration(
                  color: Colors.orange,
                  shape: BoxShape.circle,
                ),
              ),
              headerStyle: const HeaderStyle(
                formatButtonVisible: false,
                titleCentered: true,
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

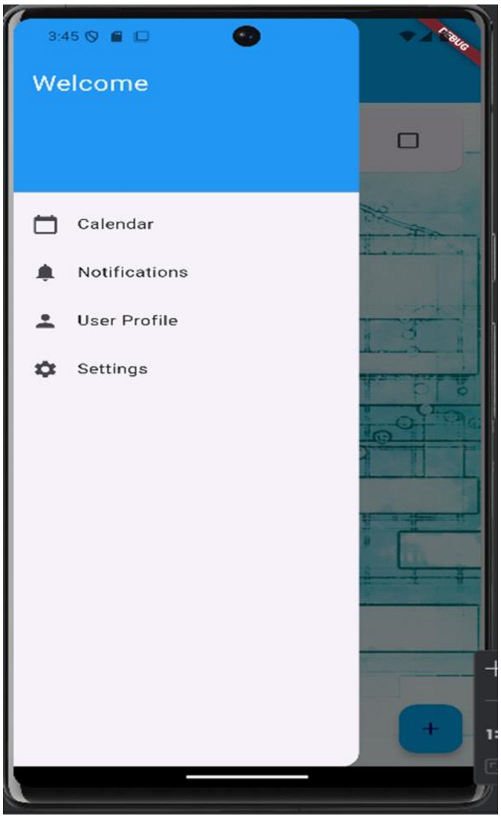
• Create Task Screen

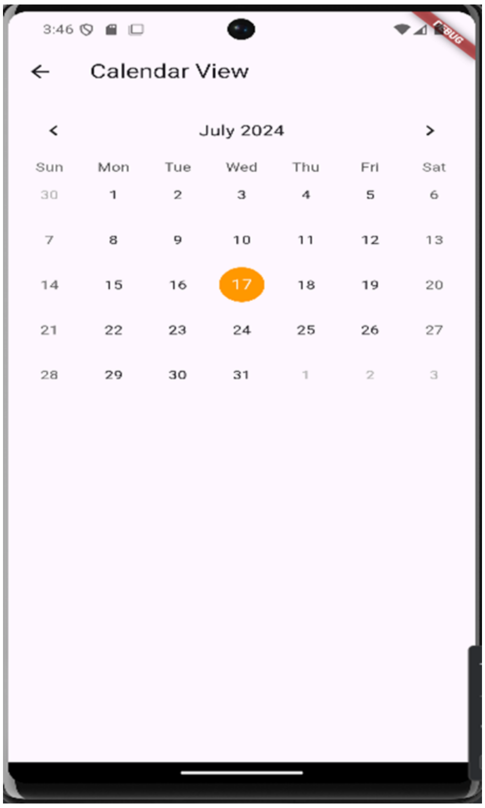
```
import 'package:flutter/material.dart';
import '../widgets/task_form.dart';

class CreateTaskScreen extends StatelessWidget {
  final Function(String, String, DateTime?, bool, bool) onSave;

  const CreateTaskScreen({super.key, required this.onSave});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Create Task'),
      ),
      body: TaskForm(
        onSave: (title, description, deadline, isCompleted, isPending) {
          onSave(title, description, deadline, isCompleted, isPending);
          // Display a SnackBar notification
          ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(
              content: Text('Task saved: $title\nDeadline:
${deadline?.toLocal().toString()}'),
              duration: const Duration(seconds: 3),
            ),
          );

          // Optionally show a dialog after saving
          showDialog(
            context: context,
            builder: (ctx) => AlertDialog(
              title: const Text('Task Saved'),
              content: Text('Task: $title\nDescription:
$description\nDeadline: ${deadline?.toLocal().toString()}\nCompleted:
$isCompleted\nPending: $isPending'),
              actions: [
                TextButton(
                  onPressed: () {
                    Navigator.of(ctx).pop(); // Close the dialog
                    Navigator.of(ctx).pop(); // Go back to the previous
screen
                  },
                  child: const Text('OK'),
                ),
              ],
            ),
          );
        },
      ),
    );
  }
}
```

• Setting Deadline for the task

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

class TaskDetailsScreen extends StatelessWidget {
  final String taskId;

  const TaskDetailsScreen({super.key, required this.taskId});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Task Details'),
      ),
      body: Center(
        child: Text('Details for Task ID: $taskId'),
      ),
    );
  }
}
```

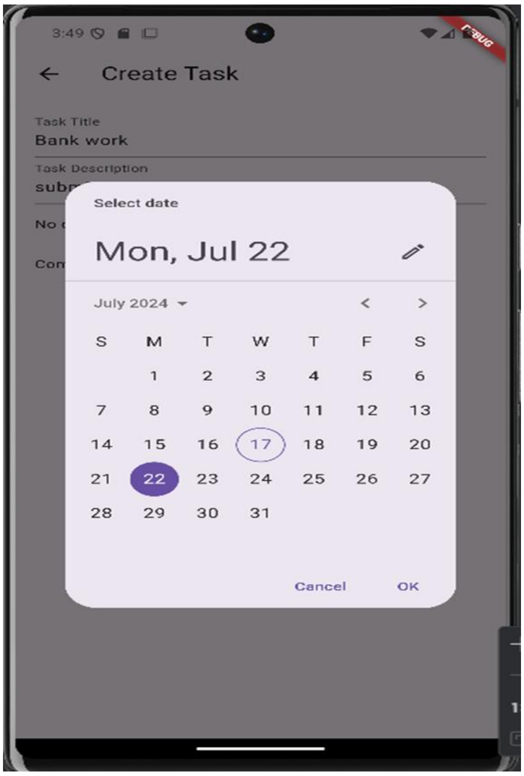
5.Output





July 2024

Sun	Mon	Tue	Wed	Thu	Fri	Sat
30	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3



Task Title
Bank work

Task Description
subr

No

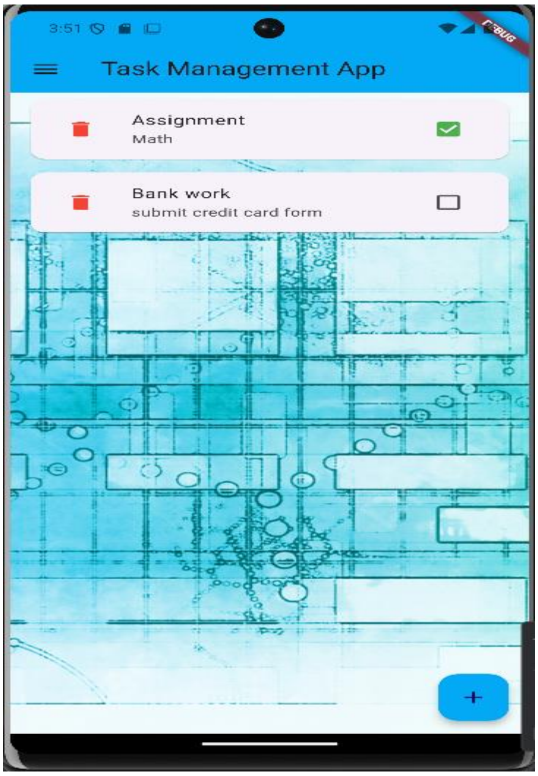
Con

Select date

Mon, Jul 22

July 2024

S	M	T	W	T	F	S
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			



Assignment
Math ☑

Bank work
submit credit card form ☐