Objective:

Develop a Personal Expense Tracker mobile application using Flutter that showcases expertise in Clean Architecture, state management, UI/UX design, local notifications, and unit testing.

Project: Personal Expense Tracker App

Requirements:

1. **App Overview**:

   - A mobile application for users to track and manage personal expenses.

2. **Features**:

   - **Add Expense**: Users can input the amount, date, and a brief description.
   - **View Expenses**: Display expenses in a list, sorted and filterable by date.
   - **Edit/Delete Expense**: Modify or remove expense entries.
   - **Expense Summary**: Show summaries categorized by type, on a weekly or monthly basis.
   - **Reminder Notifications**: Set up reminders for users to record their daily expenses.

3. **UI/UX**:

   - Develop a user-friendly, visually appealing interface.
   - Utilize responsive Flutter widgets.
   - Apply a consistent and appealing theme.

4. **State Management**:

   - Implement a robust state management solution.

5. **Data Persistence**:

   - Persist data locally using SQLite, Hive, or a similar database.

6. **Clean Architecture**:

   - Adhere to Clean Architecture principles.
   - Clearly define and separate presentation, business logic, and data layers.

7. **Unit Testing**:

   - Conduct thorough unit testing, particularly for business logic.
   - Aim for substantial code coverage.
   - Employ relevant testing frameworks and practices.

8. **Local Notifications**:

   - Implement local notifications for daily expense recording reminders.

- Ensure notifications are scheduled and managed effectively.

9. **Code Quality**:

- Ensure code is clean, maintainable, and follows best practices.

10. **Bonus (Optional)**:

- User authentication features.
- Multi-language support.
- Advanced UI animations or transitions.

Submission Guidelines:

- Code submission via GitHub repository.
- Include a README for setup and running instructions.
- A report explaining design decisions, architectural choices, and testing approach.

Evaluation Criteria:

- Functionality: Adherence to the specified features and requirements.
- Code Quality: Clean, well-organized, and documented code.
- UI/UX Design: Intuitive and attractive interface.
- Clean Architecture: Effective use of Clean Architecture.
- Unit Testing: Comprehensive tests and coverage.
- Notifications: Effective implementation and management of local notifications.
- Flutter Capabilities: Proficient use of Flutter features and widgets.