Technical Documentation: Fertilizer Shop Management App

⬚ Technical Documentation: Fertilizer Shop Management App

⬚ Project Overview

Title: Fertilizer Shop Owner Native Mobile Application
Platform: Flutter (Android)
Backend: Firebase (Firestore, Auth, Storage)
Purpose: Digitize fertilizer product inventory, customer credit tracking, billing, AI bill scanning, and supply management for fertilizer shops.

⬚ Project Structure

```
lib/
├── main.dart
├── models/
│   ├── fertilizer.dart
│   ├── customer.dart
│   ├── bill.dart
│   └── supply.dart
├── screens/
│   ├── login_screen.dart
│   ├── home_screen.dart
│   ├── fertilizer_screen.dart
│   ├── customer_screen.dart
│   ├── billing_screen.dart
│   ├── credit_history_screen.dart
│   ├── ocr_scanner_screen.dart
│   ├── backup_screen.dart
│   └── supply_screen.dart
├── services/
│   └── notification_service.dart
```

Features and Modules

1. Login
- Firebase Email/Password authentication
- Only authenticated users can access the app

2. Fertilizer Management
- Add/update/delete fertilizers
- Fields: name, price, company name, size, chemical, purpose, MFD/EXP dates, image
- Search bar to quickly find fertilizers
- Firestore as database

3. Customer & Credit Tracking
- Add customer manually with:
  - Name, surname
  - Village
  - Phone
  - Signature (captured using signature pad)
- Track purchases made on credit
- Finance section to record customer repayments
- Searchable by name/village

4. Billing System
- Create credit and cash bills
- Choose fertilizers from database
- Allow manual price override per bill
- Saves billing record to customer credit history

5. OCR Scanner for Bill Extraction
- Upload scanned bill image
- Extract product details using ML Kit Text Recognition
- Save scanned bill data as a digital record

6. Company Supply Tracker
- Record incoming supplies from companies
- Track quantity, total amount, paid amount
- Filter by company/date
- PDF export of supplies
- Local notification for unpaid balances

7. Google Drive Backup
- Backup Firestore data and images to Drive

- Auto/manual backup modes
- Option to restore from backup

8. ⬜ Local Notifications
- Notify owner of unpaid balances
- Notification channel setup via flutter_local_notifications

⬜ Firebase Configuration

1. Create project in Firebase Console
2. Enable:
   - Firestore Database
   - Firebase Authentication
   - Firebase Storage
3. Add Android app to project
4. Download google-services.json
5. Place inside android/app/
6. Edit Gradle files:

// android/build.gradle
classpath 'com.google.gms:google-services:4.3.15'

// android/app/build.gradle
apply plugin: 'com.google.gms.google-services'

⬜ Dependencies (pubspec.yaml)

```
dependencies:
 flutter:
  sdk: flutter
 firebase_core: ^2.13.0
 firebase_auth: ^4.7.3
 cloud_firestore: ^4.9.0
 firebase_storage: ^11.2.1
 image_picker: ^1.0.4
 signature: ^5.3.0
 path_provider: ^2.1.2
 flutter_local_notifications: ^17.0.0
 pdf: ^3.10.4
```

```
  printing: ^5.12.0
  google_mlkit_text_recognition: ^0.10.0
  googleapis: ^11.5.0
  googleapis_auth: ^1.4.0
  file_picker: ^6.2.0
```

⬜ Deployment Steps

1. Firebase Setup
- Follow instructions in Firebase Console to add Android app
- Copy google-services.json into android/app/
- Setup Firestore collections:
  - fertilizers
  - customers
  - bills
  - supplies

2. Run App Locally
flutter run

3. Build APK
flutter build apk --release
Output at:
build/app/outputs/flutter-apk/app-release.apk

4. Publish to Play Store (Optional)
- Create Google Play Developer account
- Sign APK using keystore
- Upload release APK
- Complete app listing

⬜ Security & Rules

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read, write: if request.auth != null;
    }
```

```
 }
}
```

 Testing Checklist

Module	Test	✅
Login	Email/password auth	✅
Fertilizers	Add/search/update/delete	✅
Customers	Add/view/search/signature	✅
Billing	Credit/cash, override prices	✅
OCR	Extract data from photo	✅
Backup	Upload/download files	✅
Notifications	Alert on unpaid dues	✅
PDF Export	Generate reports	✅

 Notes
- This app uses Firebase free Spark plan (sufficient for small businesses)
- Upgrade to Blaze plan if data or storage usage exceeds free limits
- App is modular and scalable for multi-user setup later

 Support & Maintenance

- Scheduled backups (manual/cron possible)
- Add PIN lock feature for sensitive sections (optional)
- Create admin dashboard (web) if expansion needed

End of documentation.