TailorPro Backend - Handover Document (UPDATED)

Date: 2026-01-19

Status: ☑ Backend Complete + Order Detail Screen Requirements

---

What We Built

3 Core Apps:

1. **orders -** Customer, Order, OrderItem, Item (with inventory)
2. **invoicing -** Invoice, InvoiceItem (GST compliant)
3. **financials -** ReceiptVoucher, Payment, RefundVoucher

Key Models & Fields

Customer Model

- Basic: name, phone, whatsapp, email, gender, customer_type (B2C/B2B)
- Business: business_name, gstin, pan
- Address: address, city, state, pincode
- **Measurements:** All gender-specific fields (women/men) + custom fields 1-10
- Soft delete: is_active, deleted_at

Order Model

- Order tracking: order_number (auto), order_date, expected_delivery_date
- Status: order_status, delivery_status

- Work: assigned_to (employee assignment)
- **QR Code:** Auto-generated for tracking
- **Reference Photos:** Unlimited via OrderReferencePhoto model (max 6 in UI)
- Lock: is_locked (after invoice created)
- **Notes:** order_notes (read-only after confirmation), change_requests (always editable with timestamp)

Item Model (Unified Catalog)

- Type: item_type (SERVICE/PRODUCT)
- **Inventory:** track_stock (yes/no), current_stock, min_stock_level, allow_negative_stock
- Stock: opening_stock, current_stock (auto-calculated)
- Pricing: purchase_price, selling_price (both optional)
- GST: hsn_sac_code, tax_percent
- Barcode: barcode field
- Safety: has_been_used (locks critical fields), is_active, deleted_at

Invoice Model

- Numbers: invoice_number (INV-YYYYMM-00001)
- Customer: Links to Customer
- Order: Optional link to Order (null for walk-ins)
- **Addresses:** billing_address, shipping_address (snapshot at invoice time)
- **GST:** tax_type (INTRASTATE/INTERSTATE/ZERO based on tenant.gst_enabled + states)
- Amounts: subtotal, cgst, sgst, igst, grand_total
- **Advance:** total_advance_adjusted (from ReceiptVouchers)
- Payment: total_paid, remaining_balance, payment_status

ReceiptVoucher Model (Advance)

- Numbers: voucher_number (RV-YYYYMM-00001)
- **GST on Advance:** advance_amount + cgst + sgst/igst = total_amount
- Link: customer, order (optional)
- **Adjustment:** adjusted_amount (property: is_adjusted)
- Cash tracking: deposited_to_bank, deposit_date

Payment Model

- Links to Invoice only
- No GST (GST already in invoice)
- Tracks: amount, payment_mode, transaction_reference

RefundVoucher Model

- Links to original ReceiptVoucher
- **GST Reversal:** Copies GST from original receipt and reverses

---

Critical Business Rules

GST Logic (FIXED)

☑ ZERO GST only when:

- `tenant.gst_enabled = False` OR
- `item.tax_percent = 0`

✖ NOT based on customer GSTIN

GST on Advances - NEW CALCULATION METHOD

Weighted Average Tax Rate (Rounded to Standard Rates):

```python
# For orders with mixed tax rates
# Calculate proportional tax based on item values

weighted_avg = Σ(item_amount/total × item_tax_percent)
rounded_rate = nearest_standard_rate([0, 5, 12, 18, 28])

# Reverse calculate from inclusive amount
base_amount = total_received / (1 + rounded_rate/100)
gst_amount = total_received - base_amount
cgst = sgst = gst_amount / 2  # or IGST if interstate
```

Example:

```
Order Items:
- Blouse (5%):    ₹10,000
- Embroidery (18%): ₹2,000
Total: ₹12,000

Weighted avg = (10000/12000 × 5%) + (2000/12000 × 18%)
             = 4.17% + 3% = 7.17%
Rounded to nearest = 5%

Customer pays ₹5000 advance:
Base = 5000/1.05 = ₹4761.90
GST @ 5% = ₹238.10
```

For Non-GST Users:

```python
if not tenant.gst_enabled:
    base_amount = total_received
    gst = 0
```

Inventory Logic

- If `item.track_stock = True` → deduct stock on order/invoice
- If `allow_negative_stock = True` → allow orders even when stock low
- StockTransaction: Auto-created audit trail (IN/OUT/ADJUSTMENT)

Safety Features

- **Soft Delete:** Never hard-delete Customer, Order, Item (use is_active=False)
- **Lock Fields:** After item used in order: lock item_type, track_stock, unit
- **Opening Stock:** Editable only once on creation
- **Immutability:** Issued receipts/invoices cannot be edited (set is_issued)

Auto-Generated Fields

- order_number: ORD-YYYYMM-00001
- invoice_number: INV-YYYYMM-00001
- voucher_number: RV-YYYYMM-00001
- payment_number: PAY-YYYYMM-00001
- refund_number: RF-YYYYMM-00001
- QR codes: Auto-generated for orders and inventory items

Order Detail Screen Requirements

Tab Structure (4 Tabs)

1. Overview Tab (Default)

Cards:

- **Customer & Status**
  - Customer name, phone
  - Order Status badge with ✏️ (tap to change)
  - Delivery Status badge with ✏️ (tap to change)
- **Order Information**
  - Order Date
  - Expected Delivery Date
  - Assigned To (employee) with ✏️
- **Financial Summary**
  - Order Total
  - Advance Received
  - Balance Due
- **Reference Photos**
  - 3 column grid layout
  - Max 6 photos
  - Delete (✕) button on each
  - [+ Add] button
- **QR Code**

    - Small thumbnail
    - Tap to enlarge/view full screen

Bottom Buttons:

- [Edit Order Details] - Opens dialog to edit order date, expected delivery, assigned to, order notes
- [Create Invoice]

Lock Indicator:

- Show 🔒 icon in header when invoice created
- Show banner: "This order is locked. Invoice INV-XXXXXX has been created. Items cannot be modified."
- Disable edit buttons when locked

---

2. Items Tab

Table Format:

```
Item Name | Qty | Rate | Tax% | Amount | ⋮
```

⋮ Menu Actions (per row):

- Edit Item (opens dialog with item details)
- Delete Item (shows confirmation)

Bottom Button:

- [+ Add Item] - Opens same dialog as order creation

When Locked:

- Hide/disable all edit/delete actions
- Show read-only table

Edit Allowed:

- Until Order Status = "Completed"
- Once invoice created = LOCKED (no edits)

---

3. Payments Tab

Sections:

1. **Financial Summary**
    - Order Total
    - Total Paid (advances + invoice payments)
    - Balance Due
2. **All Payments** (Combined list - chronological order)
    - Type badge (Advance / Invoice Payment)
    - Voucher/Payment number
    - Date
    - Payment mode
    - Amount
    - Status (for advances: Adjusted/Not Adjusted)

    Shows both:

    - ReceiptVouchers (advances)

    - Payments (invoice payments)

3. **Invoice Details** (if invoice exists)
    - Invoice number, date
    - Invoice total
    - [View Invoice] link

Bottom Button:

- [+ Record Payment]
    - If no invoice: Creates ReceiptVoucher (advance)
    - If invoice exists: Can create either advance or invoice payment

---

4. Notes Tab

Two Sections:

1. **Order Notes**
    - Original requirements captured at order creation
    - **Read-only after order confirmed**
    - Editable only when Order Status = "Draft"
2. **Change Requests**
    - Always editable (even after confirmation/lock)
    - Shows: "Last updated: DD-MM-YYYY HH:MM AM/PM by [Username]"
    - Updates timestamp on every save
    - For recording modifications after order confirmed

Bottom Button:

- [Save Changes]

Record Payment Dialog

Fields:

- Amount Received (inclusive of GST)
- Payment Mode dropdown
  - Cash
  - Card
  - UPI
  - Bank Transfer
  - Cheque
- Transaction Reference (optional)

Tax Calculation (Auto-calculated):

```
Total Received:     ₹X,XXX.XX
Base Amount:        ₹X,XXX.XX
GST @ XX%:          ₹XXX.XX
  ├─ CGST (X%):     ₹XXX.XX
  └─ SGST (X%):     ₹XXX.XX
```

For Non-GST Users:

```
Amount:             ₹X,XXX.XX
GST:                ₹0.00
Total:              ₹X,XXX.XX
```

Buttons:

- [Cancel]
- [Record Payment]

Create Invoice Flow

Step 1: Preview Dialog Shows:

- Customer details
- Order number
- All items with amounts
- Subtotal
- CGST, SGST/IGST breakdown
- Grand Total
- Advances to be adjusted
- Balance Due
- Warning: "⚠ Order will be locked after invoice is created. Items cannot be modified."

Buttons:

- [Cancel]
- [Create Invoice]

Conditions to Enable:

- ☑ Order has items
- ☑ Order Status = Confirmed/In Progress/Completed

- ✖️ Disabled if: No items, Status = Draft/Cancelled, Invoice already exists

Step 2: After Creation

- Navigate to Invoice Detail Screen
- Show back link to Order Detail screen

Status Updates

Three Status Types:

- Order Status (Draft/Confirmed/In Progress/Completed/Cancelled/On Hold)
- Delivery Status (Not Started/In Progress/Ready/Delivered)
- Assigned To (Employee)

UI Pattern:

- Badge display with ✏️ icon
- Tap badge → Opens bottom sheet with dropdown options
- Select → Auto-saves and updates

Located In: Overview Tab, in Customer & Status card

Order Locking Behavior

Trigger: When invoice is created for order

What Gets Locked:

- ✖️ Cannot add/edit/delete items

- ✖ Cannot edit order dates
- ✖ Cannot change assigned employee
- ✖ Cannot edit order notes

What Remains Enabled:

- ☑ Can add/edit change requests
- ☑ Can record advance payments
- ☑ Can record invoice payments
- ☑ Can view all details

Visual Indicators:

- 🔒 icon in header
- Locked banner message
- Disabled/hidden edit buttons

Print Options

Access: Top-right ⋮ menu

Two Print Formats:

1. **Print Order Detail** (Office Copy)
   - Full order details
   - Customer information
   - All items with prices
   - Financial summary (order total, advances, balance)
   - Payment history

    - QR Code
    - Notes (order notes + change requests)
    - **Audience:** Office/Owner use
2. **Print Workshop Copy** (Production Copy)
    - Customer name, phone
    - Order number, dates
    - Items list (NO prices)
    - Measurements/specifications
    - Special instructions
    - QR Code (for tracking)
    - **Hidden:** All financial information (prices, payments, totals)
    - **Audience:** Workshop tailors/workers

Top Menu (⋮) Actions

- Print Order Detail
- Print Workshop Copy
- Duplicate Order
- Cancel Order
- Delete Order (if no invoice)

Context-Aware Bottom Buttons

```
Overview Tab:   [Edit Order Details] [Create Invoice]
```

```
Items Tab:      [+ Add Item]
Payments Tab:   [+ Record Payment]
Notes Tab:      [Save Changes]
```

Buttons change based on active tab for better UX.

Edit Capabilities Summary

Field/Action	Draft	Confirmed	In Progress	Completed	Locked (Invoice Created)
Order Date	☑	☑	☑	☑	✖
Expected Delivery	☑	☑	☑	☑	✖
Assigned To	☑	☑	☑	☑	✖
Order Notes	☑	✖	✖	✖	✖
Change Requests	☑	☑	☑	☑	☑
Add/Edit/Delete Items	☑	☑	☑	☑	✖
Record Payments	☑	☑	☑	☑	☑
Update Status	☑	☑	☑	☑	☑

Required API Endpoints

Order Management

```
GET    /api/orders/{id}/
PATCH  /api/orders/{id}/
DELETE /api/orders/{id}/
POST   /api/orders/{id}/duplicate/
```

Order Items

```
GET    /api/orders/{id}/items/
POST   /api/orders/{id}/items/
PATCH  /api/orders/{id}/items/{item_id}/
DELETE /api/orders/{id}/items/{item_id}/
```

Reference Photos

```
GET    /api/orders/{id}/reference-photos/
POST   /api/orders/{id}/reference-photos/
DELETE /api/orders/{id}/reference-photos/{photo_id}/
```

Payments

```
GET  /api/orders/{id}/receipt-vouchers/
POST /api/orders/{id}/receipt-vouchers/
POST /api/orders/{id}/calculate-advance-tax/
    Input: {amount: decimal}
    Output: {base_amount, tax_percent, cgst, sgst, igst, total_amount}
```

Invoice

```
GET  /api/orders/{id}/invoice/
POST /api/orders/{id}/create-invoice/
    Returns: Invoice preview data
POST /api/orders/{id}/confirm-invoice/
    Creates actual invoice, locks order
```

Status Updates

```
PATCH /api/orders/{id}/update-status/
    Input: {order_status: string} or {delivery_status: string}
PATCH /api/orders/{id}/assign-employee/
    Input: {assigned_to: employee_id}
```

Print

```
GET /api/orders/{id}/print-detail/
   Returns: PDF or formatted data for office print
GET /api/orders/{id}/print-workshop/
   Returns: PDF or formatted data for workshop (no prices)
```

Database Status

☑ All models created
☑ Migrations applied
☑ Admin interfaces ready
☑ GST compliance fixed

✅ Inventory integrated into Item model
✅ Weighted average GST calculation for advances

Files to Use

1. `orders/models.py` - orders_models_final.py
2. `orders/admin.py` - orders_admin_final.py
3. `invoicing/models.py` - invoicing_models_fixed.py
4. `financials/models.py` - financials_models_fixed.py
5. `financials/admin.py` - financials_admin_fixed.py

Next Steps

Backend Development:

1. ✅ Core models complete
2. 🔄 Add new API endpoints:
    - Calculate advance tax with weighted average
    - Invoice preview
    - Print order detail / workshop copy
    - Status update endpoints
3. 🔄 Add order locking logic to order item endpoints
4. 🔄 Implement print PDF generation

Flutter Development:

1. Build Order Detail Screen with tab navigation
2. Implement Overview Tab (customer, status, financial, photos, QR)
3. Implement Items Tab (table with ⋮ menu actions)
4. Implement Payments Tab (all payments combined list)
5. Implement Notes Tab (order notes + change requests)
6. Build Record Payment dialog with GST calculation
7. Build Create Invoice preview dialog
8. Implement status update bottom sheets
9. Build print functionality (office + workshop formats)
10. Add order locking UI behavior

Key Flutter Packages Needed

- `flutter_bloc` or `provider` - State management
- `dio` - API calls
- `image_picker` - Reference photos
- `qr_flutter` - QR code display
- `pdf` - Print PDF generation
- `printing` - Print functionality
- `cached_network_image` - Photo display

Backend Team: Phase 1 Complete ✅ | Phase 2 In Progress 🔄
Flutter Team: Ready to start Order Detail Screen 🚀

Document Version: 2.0

Last Updated: 19-Jan-2026

Updated By: Lokesh + Claude