Perfect! I've provided you with comprehensive TypeScript configuration files for your grocery delivery app. Here's what each configuration file offers:

**📁 Configuration Files Overview**

**1. environment.ts**

* **Multi-environment support** (development, staging, production)
* **Firebase configuration** for push notifications and analytics
* **Payment gateway keys** for all supported providers
* **Business rules** (VAT rate, delivery charges, thresholds)
* **Feature toggles** for different app capabilities

**2. apiConfig.ts**

* **Complete API endpoints** for all app functionality
* **HTTP configuration** with timeouts, retry logic, headers
* **Status codes** and error handling constants
* **RESTful endpoint structure** following best practices

**3. paymentConfig.ts**

* **Multi-gateway support** (Stripe, Razorpay, PayPal)
* **Payment method configurations** with enabled/disabled states
* **UAE-specific settings** (AED currency, COD limits)
* **Security configurations** for PCI compliance

**4. pushConfig.ts**

* **Firebase Cloud Messaging** setup
* **Notification channels** for different message types
* **Action categories** for interactive notifications
* **Deep linking** configuration for navigation

**5. index.ts**

* **Barrel exports** for easy imports
* **App constants** and business rules
* **Feature flags** for gradual rollouts
* **UAE regional settings** (currency, timezone, VAT)

**🚀 Key Features**

**🌍 Environment Management**

typescript

*// Automatic environment detection*

const ENV = getEnvironment(); *// development | staging | production*

*// Environment-specific configurations*

const Config = {

API\_BASE\_URL: 'https://api.groceryapp.ae/api',

FIREBASE\_CONFIG: { */\* production config \*/* },

ENABLE\_DEBUG\_LOGS: false, *// Only in development*

};

**🔗 API Configuration**

typescript

*// Complete endpoint mapping*

export const API\_ENDPOINTS = {

LOGIN: '/auth/login',

PRODUCTS: '/products',

CREATE\_ORDER: '/orders',

DELIVERY\_SLOTS: '/delivery/slots',

*// ... 50+ endpoints*

};

*// Usage in services*

const response = await apiClient.get(API\_ENDPOINTS.PRODUCTS);

**💳 Payment Gateway Setup**

typescript

*// Multi-gateway configuration*

export const paymentConfig = {

gateways: {

stripe: { */\* Stripe config \*/* },

razorpay: { */\* Razorpay config \*/* },

paypal: { */\* PayPal config \*/* },

},

defaultGateway: 'stripe',

codEnabled: true, *// Cash on Delivery for UAE market*

};

**🔔 Push Notifications**

typescript

*// Notification channels*

export const pushConfig = {

channels: {

order\_updates: {

importance: 'high',

sound: true,

vibration: true,

},

promotions: {

importance: 'default',

sound: false,

},

},

};

**🇦🇪 UAE Market Specific**

**Currency & Localization**

* **AED currency** formatting and calculations
* **5% VAT rate** as per UAE regulations
* **Arabic timezone** (Asia/Dubai)
* **Weekend handling** (Friday-Saturday)

**Business Rules**

* **Free delivery threshold**: 100 AED
* **COD limits**: 20-500 AED
* **Express delivery**: 15 AED charge
* **Order cancellation**: 30-minute window

**Regional Settings**

typescript

export const REGIONAL\_SETTINGS = {

COUNTRY: 'AE',

CURRENCY: 'AED',

TIMEZONE: 'Asia/Dubai',

WEEKEND\_DAYS: [5, 6], *// Friday, Saturday*

VAT\_RATE: 0.05, *// 5%*

PHONE\_CODE: '+971',

};

**🔧 Usage Examples**

typescript

*// Import configurations*

import Config, {

API\_ENDPOINTS,

paymentConfig,

pushConfig,

APP\_CONSTANTS,

FEATURE\_FLAGS,

} from '@config';

*// Use in API services*

const apiUrl = `${Config.API\_BASE\_URL}${API\_ENDPOINTS.PRODUCTS}`;

*// Check feature flags*

if (FEATURE\_FLAGS.ENABLE\_BIOMETRIC\_AUTH) {

*// Initialize biometric authentication*

}

*// Use business constants*

const minOrderValue = APP\_CONSTANTS.BUSINESS\_RULES.MIN\_ORDER\_VALUE;

*// Payment gateway selection*

const preferredGateway = paymentConfig.defaultGateway;

**📦 Environment Variables**

Create these environment files:

**.env.development**

bash

API\_BASE\_URL=http://10.0.2.2:3000/api

STRIPE\_PUBLISHABLE\_KEY=pk\_test\_...

RAZORPAY\_KEY\_ID=rzp\_test\_...

FIREBASE\_API\_KEY=your\_dev\_firebase\_key

GOOGLE\_MAPS\_API\_KEY=your\_maps\_key

**.env.production**

bash

API\_BASE\_URL=https://api.groceryapp.ae/api

STRIPE\_PUBLISHABLE\_KEY=pk\_live\_...

RAZORPAY\_KEY\_ID=rzp\_live\_...

FIREBASE\_API\_KEY=your\_prod\_firebase\_key

GOOGLE\_MAPS\_API\_KEY=your\_maps\_key

**🛡️ Security Features**

**API Security**

* **Token-based authentication** headers
* **Request timeout** protection
* **Retry logic** for failed requests
* **Environment-specific** API keys

**Payment Security**

* **PCI-compliant** gateway configurations
* **Webhook verification** secrets
* **Environment isolation** for payment keys
* **Fraud prevention** settings

**Data Protection**

* **Encrypted storage** keys
* **Secure communication** headers
* **Privacy compliance** settings
* **GDPR-ready** configurations

**📁 File Structure Update**

Your app/config/ directory now contains:

app/config/

├── index.ts # ✅ NEW - Barrel exports & constants

├── environment.ts # ✅ NEW - Environment configurations

├── apiConfig.ts # ✅ NEW - API endpoints & settings

├── paymentConfig.ts # ✅ NEW - Payment gateway setup

└── pushConfig.ts # ✅ NEW - Push notification config

**🎯 Production Ready**

These configurations are ready for:

* **App Store/Play Store** submissions
* **Production deployment** with proper environment handling
* **Security audits** with encrypted keys and secure defaults
* **Scalability** with feature flags and environment switching
* **Maintenance** with clear separation of concerns
* **Team collaboration** with well-documented configurations

**🚀 Next Steps**

1. **Update your .env files** with actual API keys
2. **Configure Firebase project** for push notifications
3. **Set up payment gateway accounts** (Stripe, Razorpay, PayPal)
4. **Test environment switching** between dev/staging/production
5. **Implement feature flags** for gradual rollouts

Your grocery delivery app now has a complete, production-ready configuration system! 🛒⚙️

Would you like me to help you implement any specific configuration integration or set up additional services like Firebase or payment gateways?