

Technical Requirement Document (TRD)

Project Name: MyShipFlex.com

Prepared For: Zonos Integration Team

Prepared By: Tapan Joshi (OM Software)

Date: 17-Dec-2024

1. Project Overview

The project involves integrating multiple carrier services into the shipping website to provide:

1. Real-time multiple carrier rates based on package details and destination.
2. **Creating Shipment, generating label** and commercial invoices, via multiple carriers.
3. Accurate HTS (Harmonized Tariff Schedule) code fetching based on item description and country of origin/destination.
4. Providing status tracking of shipments, feature to cancel un-picked shipments before a designated time period.
5. Integration of invoices send by Carriers (DHL, FedEx, UPS) to MyShipFlex for payments and payment reconciliation for underpaid shipments.

This integration will enhance user experience, streamline shipping processes, and ensure compliance with international shipping requirements.

2. Functional Requirements

2.1 Multiple Carrier Rate Fetching

Description

Enable users to fetch multiple carrier rates in real-time based on the following input parameters:

- Package dimensions: Length, Width, Height (in cm/inches)
- Weight (in kg/lbs)
- Origin address (ship-from location)
- Destination address (ship-to location: domestic and international)
- Shipping service level (e.g., standard, express, overnight)

Note: API should be able to handle multiple line items (final rate fetched based on grand total weight of the shipment). **API for address validation also required to avoid errors.**

Technical Requirements

- Integration with APIs of multiple carriers (particularly FedEx, UPS & DHL).
- Dynamic rate calculations based on carrier APIs for various services.

- Ability for MyShipFlex Admin to change markups and discounts to customers based on volume of shipments.
- Display rates in an organized format, sorted by cost, delivery time, or carrier.
- Error handling: Display meaningful error messages for invalid inputs (shipping address, weight, illegal item, etc.) or API failures.

API Calls

- Carrier Rate API: Fetch shipping costs for packages.
- Input: Package details, origin, destination.
- Output: Carrier name, service type, cost breakup (shipping charges, fuel surcharge, additional charges, peak surcharges, etc.), estimated delivery time.

2.2 Shipment Creation, Label Generation and Commercial Invoice

Description

Enable the creation of shipments with valid and verified addresses, generate shipping labels and commercial invoices after paying for their preferred carrier's shipping service.

Technical Requirements

- Creating shipments with valid addresses (street address, zip codes, etc.) ensuring compliance with carrier's shipping norms and rules.
- Support label generation for multiple carriers via API integration.
- Generate a downloadable and printable PDF shipping label.
- Support both domestic and international shipping labels.
- Store generated labels and invoices for admin access.
- Generate commercial invoices for international shipments with details:
 - Sender, recipient address and 'sold to' address details
 - Item description, value, currency, weight, units,
 - HTS code (as fetched in Section 2.3)
 - Origin and destination countries
 - Shipping terms (e.g., DAP, CIF, etc.), Place of Incoterm, etc.
 - Other relevant information from our system (VAT, EORI, Tax related information, etc.)

API Calls

- Carrier Label API: Generate shipping label after receiving shipment details and payment by customers.
- Input: Package details, carrier selection, user contact information.
- Output: Shipping label in PDF format, tracking number.

2.3 Accurate HTS Code Fetching

Description

Automatically fetch HTS (Harmonized Tariff Schedule) codes based on item description and

destination country.

Technical Requirements

- Integration with an HTS code classification API/database.
- Input: Item description, country of origin, and destination country.
- Output: Accurate HTS code for the provided details.
- Allow fallback or manual input for HTS codes if API returns no match.
- Ensure compliance with global customs regulations for international shipments.

API Calls

- HTS Code API: Fetch HTS codes based on product description and countries.
- Input: Item description, destination country.
- Output: HTS code, product classification details, and tariff details (if available).

2.4 Carrier Invoice Integration

Description

Fetch carrier-generated invoices post-delivery for admin payments and reconcile underpayments due to discrepancies.

Technical Requirements

- Integrate with carrier APIs to retrieve final delivery invoices.
- Input parameters: Shipment ID, Tracking Number.
- Output: Invoice ID, Final weight, dimensions, charges, Additional fees (e.g., weight discrepancies, surcharges)
- Compare initial shipment details with final invoice data to identify discrepancies.
- Highlight underpaid amounts for follow-up and allow payment reconciliation.
- Display invoices and discrepancy details on the admin dashboard.
- Ensure secure API calls and error logging for failed retrievals.

API Calls

- **Invoice Retrieval API:** Invoice ID, Final Charges, Additional Fees
- **Carrier Payment API:** For making payments to the carrier

3. Integration Requirements

3.1 APIs to Integrate

- **Carrier APIs** (FedEx, UPS, DHL): For rate fetching, shipment creation, label generation, and status tracking.
- **HTS Code API:** For accurate classification of items for international shipping.
- **Commercial Invoice API/Generation Tools:** If not provided by carrier APIs, build a template-based generator.

- **Carrier Invoice APIs:** For making payments to carriers and payment reconciliation process.

3.2 Data Flow

1. User Input:

- Package details (dimensions, weight)
- Ship-from and ship-to addresses
- Item details (description, value, origin)

2. System Actions:

- User creates a shipment → Call carrier rate APIs → Fetch rates → Display to user.
- Selects preferred carrier and service based on rates → Makes Payment → Generates label.
- Call HTS code API → Fetch HTS code → Embed in commercial invoice.
- Retrieve carrier invoices → Compare with initial shipment data.

3. Outputs:

- Real-time shipping rates for multiple carriers
- Downloadable shipping label PDF
- Downloadable Commercial invoice with embedded HTS codes
- Final Carrier Invoices for admin panel

4. Non-Functional Requirements

- **Performance:** API responses should be fetched in under 2 seconds.
- **Scalability:** System should handle high traffic without performance degradation.
- **Accuracy:** Ensure 99.9% accuracy in HTS code fetching and rate display.
- **Security:**
 - Secure API calls with proper authentication (e.g., API keys, OAuth).
 - Secure user data with SSL encryption.
- **Reliability:** System should recover from API errors with retry mechanisms.

5. User Interface (UI) Requirements

- Display carrier rates in a user-friendly format (grid or list view).
- Allow users to select a preferred shipping carrier and service.
- Provide a button to “Download Label” and “Generate Invoice.”
- Show HTS code in a separate field during the checkout/shipping process.
- Display carrier invoices in a user friendly format for admin to pay and trigger payment reconciliation.

6. Assumptions

- Carrier APIs provide all necessary rate, label, and tracking details.
- The HTS code API/database used is up-to-date and compliant with international standards.
- APIs have sufficient rate limits for user load.

7. Deliverables

1. Fully functional API integrations for carrier rates, shipment creation, label generation, HTS code fetching and carrier invoices.
2. Testing report to ensure accuracy and system reliability.
3. Documentation of API endpoints, integration setup, and user flow.

8. Acceptance Criteria

- Carrier rates are fetched and displayed accurately.
- Labels are generated successfully in PDF format for multiple carriers.
- Commercial invoices include accurate HTS codes for items.
- Carrier Invoices are fetched accurately and payment reconciliation is calculated correctly without any errors.

9. Dependencies

- Access to carrier and HTS APIs with valid API keys.
- User interface on the website for input and output display.