

SFG ALUMINIUM APP REGISTRATION - IMPLEMENTATION SUMMARY

Version: 1.6.0

Implementation Date: November 5, 2025

Status:  Complete and Operational



OVERVIEW

This document summarizes the implementation of the **SFG Aluminium-specific App Registration System** with real-time orchestration capabilities. This is a major enhancement to the existing satellite registration system (v1.5.0), adding webhook endpoints and message handlers for business-critical apps.

WHAT WAS IMPLEMENTED

1. SFG Aluminium Registration Guide

File: /satellite-registration/SFG_ALUMINIUM_APP_REGISTRATION.md

A comprehensive registration prompt specifically for SFG Aluminium business apps that require real-time communication with NEXUS and other SFG systems.

Key Features:

- Complete step-by-step registration process
- GitHub App credentials included
- Webhook setup instructions with signature verification
- Message handler implementation patterns
- SFG-specific business rules documentation
- Troubleshooting guides
- Success criteria and approval workflow

2. Webhook Handler Examples

Python Implementation

File: /satellite-registration/examples/webhook-handler-python.py

Features:

- FastAPI-based webhook endpoint
- HMAC signature verification for security
- Event routing system for 7 event types
- Business logic implementation for each event type
- Health check endpoint

Event Types Handled:

- `enquiry.created` - New customer enquiry received

- `quote.requested` - Quote generation requested
- `order.approved` - Order approval notification
- `customer.registered` - New customer registration
- `credit.check_required` - Credit check trigger
- `invoice.due` - Payment due notification
- `payment.received` - Payment confirmation

Business Logic Examples:

- SharePoint folder creation on enquiry
- Estimator assignment based on workload
- Credit check automation for orders > £10k
- Quote generation with 15% minimum margin enforcement
- Tier-based approval routing
- Production scheduling
- Invoice creation in Xero

Node.js Implementation

File: /satellite-registration/examples/webhook-handler-nodejs.js

Same functionality as Python version, implemented using Express.js.

3. Message Handler Examples

Python Implementation

File: /satellite-registration/examples/message-handler-python.py

Features:

- FastAPI-based message endpoint
- Request/response message pattern
- Message routing system for 6 message types
- Structured response format with request tracking

Message Types Handled:

- `query.customer_data` - Fetch customer information (tier, credit limit, balance)
- `query.quote_status` - Get quote details and validity
- `query.order_status` - Get order tracking and production progress
- `action.create_quote` - Generate new quote with margin validation
- `action.approve_order` - Approve order and trigger production
- `action.send_invoice` - Send invoice via email

Data Returned:

- Customer data with tier information and credit status
- Quote status with expiration and PDF URLs
- Order status with production timeline
- Quote creation with margin enforcement
- Order approval with production scheduling
- Invoice generation with Xero integration

Node.js Implementation

File: /satellite-registration/examples/message-handler-nodejs.js

Same functionality as Python version, implemented using Express.js.

4. SFG Aluminium Example Registration

File: /satellite-registration/examples/sfg-aluminium-example.json

A complete, production-ready registration example for “SFG Customer Portal” including:

App Information:

- Name, version, platform, status, URL
- Webhook URL and events
- Message handler URL and supported messages

Capabilities:

- Customer self-service portal
- Real-time enquiry submission
- Instant quote requests
- Live order tracking
- Document library access
- Credit limit visibility
- Real-time notifications
- Mobile-responsive design

Workflows:

- Customer Enquiry Submission (7 steps)
- Quote Request Processing (9 steps)
- Order Tracking (7 steps)
- Each with SFG-specific business rules

Business Rules:

- Minimum margin enforcement (15%)
- Credit check requirement (> £10k)
- Tier-based approval limits (T1-T5)
- Customer tier discounts (Platinum to Crimson)

Integrations:

- NEXUS (orchestration)
- MCP servers (SALES, FINANCE, OPERATIONS, COMMUNICATIONS, DATA)
- SharePoint (documents)
- Xero (accounting)
- Experian (credit checks)
- Companies House (company verification)

API Endpoints:

- POST /api/enquiries (submit enquiry)
- POST /api/quotes/request (request quote)
- GET /api/orders/:id/status (get order status)
- GET /api/customer/profile (get customer profile)

Data Models:

- Customer (tier, credit limit, balance)
- Enquiry (customer, description, status)
- Quote (items, amount, margin, status)
- Order (status, production dates, progress)

SFG BUSINESS RULES IMPLEMENTED

1. Margin Enforcement

```
# Minimum margin: 15%
# Target margin: 25%
# Warning threshold: 18%

if margin < 0.15:
    return {"status": "rejected", "reason": "Margin below minimum (15%)"}
```

2. Tier-Based Approval Limits

```
tiers = {
    "T1": {"name": "Director", "limit": 1000000},
    "T2": {"name": "Senior Manager", "limit": 100000},
    "T3": {"name": "Manager", "limit": 25000},
    "T4": {"name": "Supervisor", "limit": 10000},
    "T5": {"name": "Staff", "limit": 1000}
}

# Escalate if value exceeds tier limit
if order_value > tier_limit:
    escalate_to_higher_tier()
```

3. Credit Check Automation

```
# Required for orders > £10,000
# Valid for 90 days
# Use Experian API via MCP-FINANCE

if order_value > 10000:
    if credit_check_age > 90:
        request_credit_check_via_experian()
```

4. Customer Tier System

```
tiers = {
    "platinum": {"color": "purple", "discount": 0.10},
    "sapphire": {"color": "blue", "discount": 0.07},
    "steel": {"color": "gray", "discount": 0.03},
    "green": {"color": "green", "discount": 0.00},
    "crimson": {"color": "red", "discount": 0.00}
}

# Apply tier discount to pricing
price_with_discount = base_price * (1 - tier_discount)
```

5. Document Stage Tracking

ENQ → QUA → SENT → ACC → ORD → FAB → INS → INV → PAID

ENQ = Enquiry received
 QUA = Quote generated
 SENT = Quote sent to customer
 ACC = Quote accepted by customer
 ORD = Order placed
 FAB = In fabrication
 INS = Installation scheduled/complete
 INV = Invoice sent
 PAID = Payment received

REGISTRATION WORKFLOW

For SFG Aluminium Business Apps

Step 1: Choose the SFG-specific registration guide

- File: `/satellite-registration/SFG_ALUMINIUM_APP_REGISTRATION.md`
- Use for: Customer portals, operations apps, finance apps

Step 2: Implement webhook endpoint

- Copy code from `webhook-handler-python.py` or `webhook-handler-nodejs.js`
- Deploy to your app server
- Register webhook URL in metadata

Step 3: Implement message handler

- Copy code from `message-handler-python.py` or `message-handler-nodejs.js`
- Deploy to your app server
- Register message handler URL in metadata

Step 4: Extract business logic

- Follow the template in `sfg-aluminium-example.json`
- Include SFG-specific rules (margins, tiers, approval limits)
- Document workflows with steps and triggers

Step 5: Create registration issue on GitHub

- Title: [Registration] [Your App Name]
- Body: Include webhook URL, message handler URL, capabilities, workflows
- Labels: `registration`, `satellite-app`, `sfg-aluminium-app`, `pending-approval`

Step 6: NEXUS reviews and tests

- Webhook endpoint tested (sends test event)
- Message handler tested (sends test message)
- Business logic validated
- Approval within 24 hours

Step 7: Begin orchestrated operation

- Receive real-time events from NEXUS
- Respond to messages from other apps
- Participate in orchestrated workflows

USE CASES

Use SFG Aluminium Registration For:

- Customer portals (enquiries, quotes, orders)
- Operations apps (fabrication, installation, scheduling)
- Finance apps (invoicing, credit checking, payments)
- Sales apps (CRM, lead management, quote generation)
- Any app that needs real-time NEXUS orchestration

Use Universal Registration For:

- Analytics dashboards
- Reporting tools
- Admin panels
- Configuration tools
- Utility apps



FILES ADDED IN v1.6.0

```

satellite-registration/
├── SFG_ALUMINIUM_APP_REGISTRATION.md      # Main guide (38KB)
└── examples/
    ├── webhook-handler-python.py            # Python webhook (12KB)
    ├── webhook-handler-nodejs.js          # Node.js webhook (10KB)
    ├── message-handler-python.py          # Python message handler (9KB)
    ├── message-handler-nodejs.js        # Node.js message handler (8KB)
    └── sfg-aluminium-example.json       # Complete example (15KB)

```

Total Files: 5 new implementation files

Total Code: ~92KB of production-ready code

Languages: Python (FastAPI) and Node.js (Express)



SECURITY FEATURES

Webhook Security

- HMAC-SHA256 signature verification
- Webhook secret stored securely in environment variables
- Request validation on every webhook call
- Invalid signatures rejected with 401 Unauthorized

Message Handler Security

- Request ID tracking for audit trails
- Structured error responses
- Input validation on all parameters
- No sensitive data in logs

GitHub Integration Security

- GitHub App authentication (not personal access tokens)
 - Private key stored securely in .env file
 - Scoped permissions (issues only)
 - Automated credential rotation supported
-

SUCCESS CRITERIA

Your app is successfully registered when:

1.  GitHub issue created with [Registration] title
 2.  Files created in /apps/[your-app-name]/ directory
 3.  Business logic documented with SFG rules
 4.  **Webhook endpoint deployed and URL registered**
 5.  **Message handler deployed and URL registered**
 6.  Issue has labels: registration, satellite-app, sfg-aluminium-app, pending-approval
 7.  NEXUS tests webhook (receives test event successfully)
 8.  NEXUS tests message handler (receives test message successfully)
 9.  Issue label changed to approved by NEXUS
 10.  Real-time events begin flowing from NEXUS
-

INTEGRATION REQUIREMENTS

Required Integrations (MUST HAVE)

-  NEXUS - Central orchestration hub
-  MCP-SALES - Sales tools and CRM
-  MCP-FINANCE - Finance tools (Experian, Xero)
-  MCP-OPERATIONS - Production tracking
-  MCP-COMMUNICATIONS - Notifications
-  MCP-DATA - Data tools

Recommended Integrations (SHOULD HAVE)

-  SharePoint - Document storage
 -  Xero - Accounting and invoicing
 -  Experian - Credit checking
 -  Companies House - Company verification
-

NEXT STEPS

For New SFG Apps

1. Read /satellite-registration/SFG_ALUMINIUM_APP_REGISTRATION.md

2. Copy webhook handler code (Python or Node.js)
3. Copy message handler code (Python or Node.js)
4. Customize business logic for your app
5. Deploy webhook and message endpoints
6. Create GitHub registration issue
7. Wait for NEXUS approval (24 hours)
8. Begin receiving events and messages

For Existing Apps

1. Add webhook endpoint to your app
 2. Add message handler to your app
 3. Update registration with webhook URL and message handler URL
 4. NEXUS will test endpoints
 5. Begin receiving orchestrated events
-

SUPPORT

GitHub Repository: <https://github.com/sfgaluminium1-spec/sfg-app-portfolio>

Create Issue: <https://github.com/sfgaluminium1-spec/sfg-app-portfolio/issues/new>

Contact: warren@sfg-innovations.com

Review Time: 24 hours for webhook/message testing and approval

VERSION TRACKING

Current Version: v1.6.0

Previous Version: v1.5.0 (Satellite Registration System)

Version Increment: MINOR (new feature, backward compatible)

Breaking Changes: None

Deprecations: None

Migration Required: No (existing apps continue to work)

Universal Registration Still Supported: Yes

- Simple apps can still use basic registration
 - No webhooks or message handlers required for utility apps
-

IMPACT

Business Impact

- Real-time orchestration across SFG apps
- Automated workflows between apps
- Faster quote generation and order processing
- Better customer experience with instant updates
- Reduced manual handoffs between systems

Technical Impact

- Event-driven architecture
- Loosely coupled microservices
- Scalable integration pattern
- Standardized communication protocol
- Easier to add new apps to ecosystem

Developer Impact

- Clear implementation examples
 - Production-ready code templates
 - Comprehensive documentation
 - Fast onboarding (webhook + message handler < 1 day)
 - Automated testing by NEXUS
-

Implementation Summary Completed: November 5, 2025

Status:  All components operational and documented

Next Deployment: v1.7.0 (TBD based on business needs)

SFG Aluminium Real-Time Orchestration System
Empowering seamless communication across the SFG app ecosystem