Sourcing Supply Chain Net - API Architecture & **Technical Implementation**

Core APIs & Integrations Required

1. LinkedIn API Integration (CRITICAL)

Purpose: Identity verification, profile data, company validation

LinkedIn APIs Needed:

LinkedIn Sign In with OAuth 2.0

- Endpoint: https://www.linkedin.com/oauth/v2/authorization
- Scopes needed: r_liteprofile, r_emailaddress, r_organization_social, w_member_social
- Use case: User authentication, verify they work at the company

Profile API

- ∘ Endpoint: GET /v2/me
- Data retrieved: Name, email, profile picture, headline, current position
- Validation: Confirm user's current employer matches company profile

Organization Lookup API

- Endpoint: GET /v2/organizations/{id}
- Data retrieved: Company name, size, industry, headquarters, logo
- Use case: Auto-populate company profiles, verify legitimacy

• Company Updates API (if available)

- Endpoint: GET /v2/organizations/{id}/updates
- Use case: Show recent company activity (posts, articles)

Implementation Flow:



- 1. User clicks "Sign in with LinkedIn"
- 2. OAuth redirect → LinkedIn login
- 3. User grants permissions
- 4. Receive access token + profile data
- 5. Extract: name, email, company, position, profile URL
- 6. Validate: User works at company they claim (check current employer)
- 7. Store: LinkedIn ID, profile URL, last verified date
- 8. Schedule: Re-verification every 90 days

Challenges & Workarounds:



A LinkedIn API Limitations:

- Very restrictive limited access to company data
- Rate limits: ~100 requests per user per day
- May not approve apps for "sourcing" use cases

Mitigation Strategies:

1. Manual verification as backup:

- Email domain verification (<u>user@company.com</u>)
- Company website confirmation
- D&B or other business registries

2. Data enrichment alternatives:

- Clearbit API (company data)
- ZoomInfo API (contact verification)
- Hunter.io (email verification)
- RocketReach (professional contact data)

2. Email Verification & Communication APIs

SendGrid / AWS SES / Mailgun

Purpose: Transactional emails, notifications, fallback messaging

Endpoints needed:

- Send RFQ notifications to suppliers
- Response notifications to buyers
- Weekly digest emails
- POC availability status alerts
- Verification emails

Example Flow:



```
// When RFQ is posted
POST /v3/mail/send (SendGrid)
 "to": "verified-poc@supplier.com",
 "from": "rfqs@sscn.com",
 "subject": "New RFQ Match: Stainless Steel Grade 304",
 "template_id": "rfq_notification",
 "dynamic template data": {
  "buyer company": "ABC Manufacturing",
  "material": "Steel Grade 304",
  "quantity": "5000 kg",
  "rfq link": "https://sscn.com/rfq/12345"
```

Hunter.io or NeverBounce

Purpose: Verify email addresses are real and deliverable

```
javascript
```

```
GET /v2/email-verifier?email=contact@supplier.com&api key=xxx
Response: {
 "data": {
  "status": "valid",
  "score": 95,
  "regexp": true,
  "smtp check": true
```

3. Business Data Enrichment APIs

Clearbit Enrichment API

Purpose: Auto-fill company data from domain



GET /v2/companies/find?domain=advancedmaterials.com Response: { "name": "Advanced Materials Corp", "domain": "advancedmaterials.com", "logo": "https://logo.clearbit.com/...", "description": "Leading supplier of aerospace materials", "location": "Houston, TX", "employees": 450, "industry": "Manufacturing", "tags": ["b2b", "manufacturing"] }

ZoomInfo or Apollo.io

Purpose: Contact data enrichment, verify POCs

- Get employee directory
- Verify job titles
- Find decision makers
- Contact information

Dun & Bradstreet API

Purpose: Business verification, credit checks



```
GET /v1/match/cleanseMatch

{
    "companyName": "Advanced Materials Corp",
    "address": "Houston, TX"

}

Response: {
    "duns": "123456789",
    "registrationStatus": "Active",
    "yearsInBusiness": 31,
    "employeeCount": 450
}
```

4. Search & Matching Engine

Elasticsearch or Algolia

Purpose: Fast, faceted search for suppliers

Index Structure:



```
"company_id": "comp_123",
    "name": "Advanced Materials Corp",
    "materials": ["steel", "aluminum", "titanium", "alloys"],
    "certifications": ["ISO 9001", "AS9100", "ITAR"],
    "location": {
        "city": "Houston",
        "state": "TX",
        "coordinates": [29.7604, -95.3698]
      },
      "capabilities": ["heat_treatment", "custom_alloys", "testing"],
      "naics_codes": ["331110", "332710"],
      "response_rate": 95,
      "avg_response_time_hours": 4.2,
      "rating": 4.8
}
```

Search Query Example:



```
POST /suppliers/_search
{
    "query": {
        "bool": {
            "match": { "materials": "steel" }},
            { "terms": { "certifications": ["ISO 9001", "AS9100"] }}
        ],
        "filter": [
            { "range": { "response_rate": { "gte": 80 } }},
            { "geo_distance": {
                  "distance": "500mi",
                  "location": { "lat": 33.4484, "lon": -112.0740 }
        }}
        ]
        }
    }
}
```

5. File Storage & Document Management

AWS S3 or Google Cloud Storage

Purpose: Store RFQ documents, specs, CAD files

Upload Flow:



```
// 1. Request presigned URL from backend
POST /api/rfq/upload-url
Response: {
    "upload_url": "https://s3.amazonaws.com/sscn-docs/...",
    "file_key": "rfq_12345_spec.pdf"
}

// 2. Upload directly to S3 from browser
PUT https://s3.amazonaws.com/... (presigned URL)
Headers: { "Content-Type": "application/pdf" }
Body: <file data>

// 3. Notify backend upload complete
POST /api/rfq/12345/document
{
    "file_key": "rfq_12345_spec.pdf",
    "filename": "specifications.pdf"
}
```

6. Real-Time Messaging & Notifications

Pusher or Ably (WebSocket service)

Purpose: Real-time notifications, chat, status updates



```
// Subscribe to RFQ responses
const channel = pusher.subscribe('rfq-12345');
channel.bind('new-response', (data) => {
    showNotification(`New quote from ${data.supplier_name}`);
});

// Publish when supplier responds

POST /api/pusher/trigger
{
    "channel": "rfq-12345",
    "event": "new-response",
    "data": {
        "supplier_name": "Advanced Materials",
        "response_id": "resp_789"
     }
}
```

Twilio (Optional - SMS alerts)

Purpose: Critical notifications via SMS



javascript

```
POST /2010-04-01/Accounts/{AccountSid}/Messages.json

{
    "To": "+15555551234",
    "From": "+15555559876",
    "Body": "Urgent RFQ response from verified supplier. View now: https://sscn.com/rfq/..."
}
```

7. Payment Processing (Monetization)

Stripe API

Purpose: Subscription management, billing



```
// Create subscription for supplier
POST /v1/subscriptions
{
    "customer": "cus_supplier123",
    "items": [
        { "price": "price_premium_monthly" }
    ],
    "payment_behavior": "default_incomplete"
}

// Usage-based pricing for transaction fees
POST /v1/usage_records
{
    "subscription_item": "si_xyz",
    "quantity": 1, // One successful RFQ match
    "timestamp": 1234567890
}
```

8. Analytics & Monitoring

Mixpanel or Amplitude

Purpose: Product analytics, user behavior tracking

Events to track:

- RFQ posted
- Supplier viewed RFQ
- Quote submitted
- Conversation started
- Deal closed

Sentry

Purpose: Error tracking, performance monitoring

Backend API Architecture

Custom Backend APIs (FastAPI/Node.js)

Authentication Endpoints



POST /api/auth/linkedin - LinkedIn OAuth callback

POST /api/auth/verify-email - Send verification email

POST /api/auth/refresh - Refresh JWT token

GET /api/auth/me - Get current user

Company Management



POST /api/companies - Create company profile

GET /api/companies/:id - Get company details

PUT /api/companies/:id - Update company

POST /api/companies/:id/pocs - Add POC

PUT /api/companies/:id/pocs/:poc_id - Update POC availability

DELETE /api/companies/:id/pocs/:poc_id - Remove POC

RFQ Management



POST /api/rfqs - Create RFQ

GET /api/rfqs - List RFQs (with filters)

GET /api/rfqs/:id - Get RFQ details
PUT /api/rfqs/:id - Update RFQ
DELETE /api/rfqs/:id - Delete RFQ

POST /api/rfqs/:id/responses - Submit response/quote

GET /api/rfqs/:id/responses - Get all responses

Search & Discovery



GET /api/search/suppliers - Search suppliers

POST /api/search/suppliers - Advanced search with filters

GET /api/search/materials - Search materials catalog

GET /api/recommendations - AI-powered matches

Messaging



POST /api/messages - Send message

GET /api/messages/threads/:id - Get conversation thread

PUT /api/messages/:id/read - Mark as read

Analytics & Reporting



GET /api/analytics/dashboard - Dashboard metrics

GET /api/analytics/rfq/:id - RFQ performance stats

GET /api/analytics/supplier/:id - Supplier performance

Database Schema (PostgreSQL/Supabase)

Core Tables

users



sql

```
CREATE TABLE users (
id UUID PRIMARY KEY,
linkedin_id VARCHAR(255) UNIQUE,
email VARCHAR(255) UNIQUE NOT NULL,
name VARCHAR(255) NOT NULL,
profile_picture_url TEXT,
linkedin_profile_url TEXT,
created_at TIMESTAMP DEFAULT NOW(),
last_linkedin_verification TIMESTAMP,
verification_status VARCHAR(50)
);
```

companies



```
CREATE TABLE companies (
 id UUID PRIMARY KEY,
 name VARCHAR(255) NOT NULL,
 domain VARCHAR(255),
linkedin company id VARCHAR(255),
 industry VARCHAR(255),
 headquarters location VARCHAR(255),
 employee_count VARCHAR(50),
 founded year INTEGER,
 description TEXT,
 logo url TEXT,
 website_url TEXT,
 certifications JSONB,
 created_at TIMESTAMP DEFAULT NOW(),
 verified BOOLEAN DEFAULT FALSE
);
```

pocs (Points of Contact)



```
CREATE TABLE pocs (
   id UUID PRIMARY KEY,
   company_id UUID REFERENCES companies(id),
   user id UUID REFERENCES users(id),
   role VARCHAR(255),
   is primary BOOLEAN DEFAULT FALSE,
   is_on_call BOOLEAN DEFAULT FALSE,
   availability status VARCHAR(50),
   avg_response_time_hours DECIMAL,
   response rate DECIMAL,
   last active at TIMESTAMP,
   created_at TIMESTAMP DEFAULT NOW()
  );
rfqs
  CREATE TABLE rfqs (
   id UUID PRIMARY KEY,
   buyer id UUID REFERENCES users(id),
   buyer company id UUID REFERENCES companies(id),
   title VARCHAR(255) NOT NULL,
   material_category VARCHAR(255),
   quantity VARCHAR(255),
   target price VARCHAR(255),
   specifications TEXT,
   required_certifications JSONB,
   delivery deadline DATE,
```

status VARCHAR(50) DEFAULT 'active',

view count INTEGER DEFAULT 0

expires at TIMESTAMP,

created_at TIMESTAMP DEFAULT NOW(),

rfq_responses



);

```
sql
```

```
CREATE TABLE rfq_responses (
id UUID PRIMARY KEY,
rfq id UUID REFERENCES rfqs(id),
 supplier_company_id UUID REFERENCES companies(id),
 responding_poc_id UUID REFERENCES pocs(id),
status VARCHAR(50),
price quote VARCHAR(255),
lead_time_days INTEGER,
 minimum order quantity VARCHAR(255),
message TEXT,
attachments JSONB,
 created_at TIMESTAMP DEFAULT NOW(),
responded at TIMESTAMP
);
```

messages



```
CREATE TABLE messages (
id UUID PRIMARY KEY,
rfq_id UUID REFERENCES rfqs(id),
 sender id UUID REFERENCES users(id),
recipient id UUID REFERENCES users(id),
content TEXT NOT NULL,
read at TIMESTAMP,
 created at TIMESTAMP DEFAULT NOW()
);
```

Security & Compliance

Required Security Measures

- 1. **SOC 2 Compliance**
 - Use: Vanta or Drata for compliance automation
 - Regular security audits
 - Data encryption at rest and in transit
- 2. Data Privacy

- GDPR compliance (EU users)
- CCPA compliance (California users)
- Data retention policies
- Right to deletion workflows

3. Rate Limiting

- Redis for rate limiting
- Per-user and per-IP limits
- API key management

4. Authentication & Authorization

- JWT tokens (short-lived access + refresh tokens)
- Role-based access control (RBAC)
- Two-factor authentication (optional)



6 Cost Estimates (Monthly)

Service	Tier	Estimated Cost
Supabase (Database + Auth)	Pro	\$25-100
AWS S3 (File Storage)	Pay-as-go	\$50-200
SendGrid (Email)	Essentials	\$20-100
Elasticsearch/Algolia	Startup	\$100-500
LinkedIn API	Partnership	\$0-5000*
Clearbit	Startup	\$99-299
Pusher	Startup	\$49
Stripe	Pay-as-go	2.9% + \$0.30
Sentry	Team	\$26
Hosting (AWS/GCP/Vercel)	-	\$100-500

Total MVP: \$500-2000/month *LinkedIn partnership costs vary significantly



Implementation Phases

Phase 1: MVP (Months 1-3)

- LinkedIn OAuth integration
- Basic company profiles
- RFQ posting and response
- Email notifications
- Simple search

Phase 2: Growth (Months 4-6)

- Advanced search with Elasticsearch
- Real-time messaging
- File uploads
- Analytics dashboard
- Mobile responsive design

Phase 3: Scale (Months 7-12)

• AI-powered matching

- Enterprise integrations (SAP, Coupa)
- Mobile apps
- Advanced analytics
- API marketplace



A Key Risks & Mitigation

Risk 1: LinkedIn API Access Denied

Mitigation:

- Plan B: Email + domain verification
- Use data enrichment APIs (Clearbit, ZoomInfo)
- Manual verification process for high-value suppliers

Risk 2: Data Quality Issues

Mitigation:

- Implement verification workflows
- User-reported flags
- Automated stale data detection
- Regular re-verification reminders

Risk 3: Cold Start Problem (No suppliers)

Mitigation:

- Seed database with top 100 suppliers manually
- Offer free premium for early adopters
- LinkedIn outreach campaigns
- Industry partnerships

Success Metrics to Track

1. Platform Health

- POC verification rate (target: 100%)
- Average POCs per company (target: 3-5)
- POC response rate (target: >30%)
- Average response time (target: <48h)

2. User Engagement

- RFQs posted per month
- Responses per RFQ (target: 8-12)
- Conversation start rate
- Deal closure rate

3. Technical Performance

- API response time (<200ms)
- Search latency (<100ms)
- Uptime (99.9%)
- Error rate (<0.1%)

This platform fills a MASSIVE gap in B2B supply chain operations. The key differentiator is the verified POC layer with LinkedIn authentication - something no other platform does well!		