

REQUERIMIENTO FUNCIONAL: API DE FACTURACIÓN ELECTRÓNICA - PERÚ (UBL 2.1)

1. INFORMACIÓN GENERAL

- **Código del requerimiento:** REQ-FE-GO-UBL21-001
 - **Nombre del requerimiento:** Desarrollo de API REST para Facturación Electrónica con SUNAT
 - **Módulo/Sistema:** API de Facturación Electrónica
 - **Prioridad:** Alta
 - **Versión:** 1.0
-

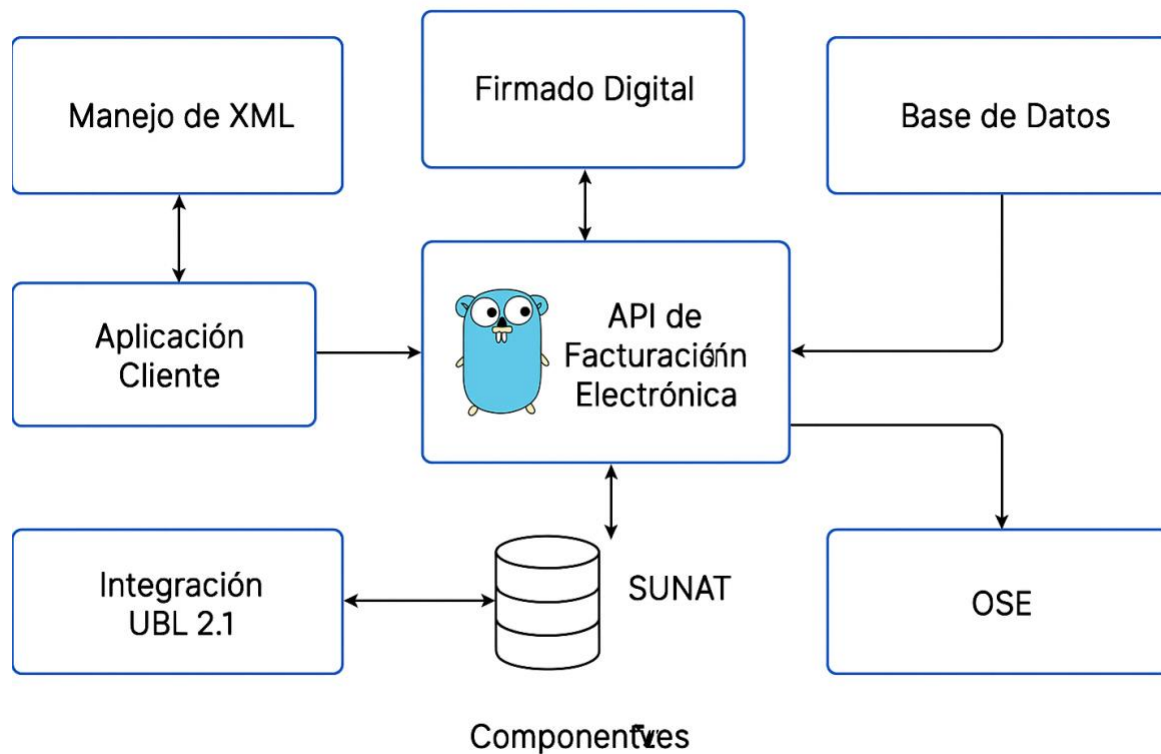
2. DESCRIPCIÓN FUNCIONAL

Propósito: Automatizar la generación, firma, envío y seguimiento de comprobantes electrónicos a SUNAT, cumpliendo con la normativa vigente y utilizando el estándar UBL 2.1.

Alcance: Incluye facturas, boletas de venta, notas de crédito y notas de débito. Aplica a emisores con obligación de emitir comprobantes electrónicos bajo la modalidad directa o mediante OSE.

Normativas aplicables: Resoluciones SUNAT N° 300-2014/SUNAT, 193-2020/SUNAT, y guías técnicas de SUNAT sobre UBL 2.1.

Integración: Estructura de documentos basada en UBL 2.1, con extensiones SUNAT (UBLExtensions) y catálogos oficiales.



3. ACTORES INVOLUCRADOS

- **Actor principal:** ERP/Aplicación de gestión empresarial del contribuyente
- **Actores secundarios:** SUNAT, OSE (Operador de Servicios Electrónicos), Sistema de firma digital, servicios de almacenamiento documental
- **Roles y permisos:** Usuario autenticado, certificado digital vigente, RUC habido

4. PRECONDICIONES

- Certificado digital emitido por entidad certificadora autorizada
- RUC del emisor activo y habido en SUNAT
- Ambiente correctamente configurado (Producción o Beta)
- Validación de datos estructurales y tributarios según guías SUNAT

5. FLUJO PRINCIPAL

1. Recepción del JSON con los datos del comprobante
2. Validación de datos conforme a catálogos SUNAT
3. Generación del XML UBL 2.1 con extensiones SUNAT

4. Firma digital del XML (PKCS#12, algoritmo SHA256)
 5. Envío a SUNAT mediante REST (SOAP para OSE si aplica)
 6. Procesamiento de la respuesta (CDR)
 7. Almacenamiento del XML, CDR, PDF y registro en base de datos
-

6. FLUJOS ALTERNATIVOS

- Rechazo del comprobante por inconsistencias tributarias o estructurales
 - Fallo de conexión a SUNAT (reintentos automáticos y logging)
 - Error en proceso de firmado digital (clave incorrecta, archivo corrupto)
 - Generación de comprobantes de contingencia con resumen diario (para Boletas)
-

7. VALIDACIONES ESPECÍFICAS SUNAT

- Validación de estructura del RUC (11 dígitos)
 - Tipo y número de documento del receptor (DNI, RUC, CE)
 - Tipo de comprobante conforme a catálogo 01 (Factura, Boleta, etc.)
 - Códigos de moneda conforme al catálogo 02 (PEN, USD, EUR)
 - Unidades de medida (catálogo 03) por cada línea del comprobante
 - Cálculo del IGV (18%) y verificación de detracciones si aplica
 - Estructura y correlación de numeración (serie, número)
-

8. ESTRUCTURA UBL 2.1

- Documentos: Invoice, CreditNote, DebitNote
 - Elementos obligatorios: cbc:ID, cbc:IssueDate, cac:AccountingSupplierParty, cac:AccountingCustomerParty, cac:TaxTotal, cac:LegalMonetaryTotal, cac:InvoiceLine
 - Campos extendidos: UBLExtensions, AdditionalInformation, SUNATTransaction, SUNATEmbeddedDespatchAdvice
 - Códigos SUNAT: Catálogos 01 al 51 según corresponda
 - Detalle: cac:InvoiceLine con Item, Price, TaxSubtotal
-

9. ENDPOINTS DE LA API

- POST /api/v1/invoices – Generar factura
- POST /api/v1/credit-notes – Generar nota de crédito
- POST /api/v1/debit-notes – Generar nota de débito
- GET /api/v1/documents/{id}/status – Consultar estado SUNAT

- GET /api/v1/documents/{id}/xml – Obtener XML firmado
 - GET /api/v1/documents/{id}/pdf – Obtener representación impresa
-

10. PARÁMETROS DE ENTRADA

- Datos del emisor (RUC, razón social, certificado, clave)
 - Datos del receptor (tipo y número de documento, nombre)
 - Datos del comprobante (tipo, serie, número, fecha, moneda)
 - Líneas de detalle (código, descripción, cantidad, precio, IGV)
 - Totales e impuestos (importe total, tributos)
 - Campos adicionales según tipo de comprobante (guías, referencias, detracciones)
-

11. RESPUESTAS DE LA API

- **200 OK:** Comprobante aceptado

```
{  
  "estado": "aceptado",  
  "hash": "string",  
  "cdr_zip": "base64",  
  "xml_firmado": "base64",  
  "pdf_url": "url_documento.pdf"  
}
```

- **400 Bad Request:** Error de estructura o validación de datos
 - **401 Unauthorized:** Token de autenticación inválido o vencido
 - **500 Internal Server Error:** Fallo del sistema o integraciones
-

12. POSTCONDICIONES

- XML UBL 2.1 firmado y almacenado
 - CDR de SUNAT archivado
 - Registro del comprobante en base de datos
 - Estado actualizado: aceptado, observado o rechazado
 - Logging de auditoría completado
-

13. CRITERIOS DE ACEPTACIÓN

- XML cumple con esquema XSD SUNAT
- Firmado digital válido (X.509, SHA256)
- CDR recibido sin observaciones
- Rendimiento: < 2 seg por operación en condiciones normales

- Errores gestionados mediante retry y logs centralizados
-

14. CONSIDERACIONES TÉCNICAS GO

- Uso de goroutines y channels para procesos concurrentes
 - Librerías: encoding/xml, crypto/x509, net/http, logrus
 - Estructura de paquetes:
 - /models: estructuras UBL
 - /services: firma, generación XML, envío SUNAT
 - /handlers: endpoints HTTP REST
 - /config: parámetros y certificados
 - Patrones: Clean Architecture o Hexagonal
 - Logging estructurado, trazabilidad con zap o logrus
-

15. ANEXOS

- Resoluciones SUNAT N° 193-2020/SUNAT, 113-2022/SUNAT
- Manual de implementación UBL SUNAT (PDF)
- Esquemas XSD UBL 2.1 SUNAT
- Catálogos SUNAT (CSV/XML)
- Ejemplos de XML válidos (factura, crédito, débito)
- Documentación de servicios REST de SUNAT