

Índice de Métodos - Referencia Rápida

Índice Alfabético de Métodos

A

- `atvAdapter.consultarComprobante(clave)` - Consulta estado de comprobante
- `atvAdapter.emitirComprobante(facturaData)` - Emite comprobante electrónico
- `atvAdapter.enviarComprobante(clave)` - Envía comprobante a Hacienda
- `atvAdapter.getStatus()` - Obtiene estado del adaptador ATV
- `atvAdapter.init(customConfig)` - Inicializa adaptador ATV
- `atvAdapter.validarComprobante(clave)` - Valida comprobante por clave

C

- `config.isDevelopment()` - Verifica si está en desarrollo
- `config.isProduction()` - Verifica si está en producción

E

- `EnglishInvoiceController.convertToSpanish(englishInvoice)` - Convierte factura inglés→español
- `EnglishInvoiceController.issueInvoice(req, res)` - Emite factura en inglés
- `EnglishInvoiceController.listInvoices(req, res)` - Lista facturas (inglés)
- `EnglishInvoiceController.queryInvoice(req, res)` - Consulta factura (inglés)
- `EnglishInvoiceController.sendInvoice(req, res)` - Envía factura (inglés)
- `EnglishInvoiceController.validateInvoice(req, res)` - Valida factura (inglés)
- `EnglishInvoiceValidator.convertFromSpanish(spanishInvoice)` - Convierte español→inglés
- `EnglishInvoiceValidator.convertToSpanish(englishInvoice)` - Convierte inglés→español
- `EnglishInvoiceValidator.validateInvoice(invoiceData)` - Valida factura en inglés
- `EnglishInvoiceValidator.validateIssuer(issuer)` - Valida emisor (inglés)
- `EnglishInvoiceValidator.validateReceiver(receiver)` - Valida receptor (inglés)
- `EnglishInvoiceValidator.validateServiceDetail(serviceDetail)` - Valida servicios (inglés)
- `EnglishInvoiceValidator.validateInvoiceSummary(summary)` - Valida resumen (inglés)

F

- `FacturaController.consultarFactura(req, res)` - Consulta factura específica
- `FacturaController.eliminarFactura(req, res)` - Elimina factura (desarrollo)
- `FacturaController.emitirFactura(req, res)` - Emite nueva factura
- `FacturaController.enviarFactura(req, res)` - Envía factura a Hacienda
- `FacturaController.listarFacturas(req, res)` - Lista facturas con filtros
- `FacturaController.obtenerEstadoSistema(req, res)` - Obtiene estado del sistema
- `FacturaController.validarFactura(req, res)` - Valida factura por clave/payload
- `FacturaValidator.getSchemas()` - Obtiene esquemas Joi
- `FacturaValidator.validateBusinessLogic(factura)` - Valida lógica de negocio
- `FacturaValidator.validateClave(clave)` - Valida formato de clave
- `FacturaValidator.validateConsecutivo(consecutivo)` - Valida formato consecutivo
- `FacturaValidator.validateDetalle(detalle)` - Valida líneas de servicio
- `FacturaValidator.validateEmisor(emisor)` - Valida datos del emisor
- `FacturaValidator.validateFactura(factura, options)` - Validación completa
- `FacturaValidator.validateReceptor(receptor)` - Valida datos del receptor
- `FacturaValidator.validateTotales(totales)` - Valida resumen de totales

G

- `generateConsecutivo()` - Genera número consecutivo único
- `generateTimestamp()` - Genera timestamp para archivos

I

- `invoiceStorage.deleteInvoice(consecutivo)` - Elimina factura del storage
- `invoiceStorage.getInvoice(consecutivo, includeContent)` - Recupera factura específica
- `invoiceStorage.getStatistics()` - Obtiene estadísticas de almacenamiento
- `invoiceStorage.listInvoices(options)` - Lista facturas con filtros
- `invoiceStorage.markAsSent(consecutivo, envioMeta)` - Marca factura como enviada
- `invoiceStorage.saveInvoiceJSON(consecutivo, facturaData)` - Guarda factura JSON
- `invoiceStorage.saveInvoiceXML(consecutivo, xmlContent)` - Guarda factura XML

L

- `logger.debug(message, meta)` - Log de debugging
- `logger.error(message, meta)` - Log de errores
- `logger.info(message, meta)` - Log de información
- `logger.logError(error, requestMeta)` - Log estructurado de errores
- `logger.logRequest(req, res, next)` - Middleware logging HTTP
- `logger.warn(message, meta)` - Log de advertencias

S

- `sanitizeFilename(filename)` - Sanitiza nombres de archivo

V

- `validateConsecutivoFormat(consecutivo)` - Valida formato de consecutivo

Métodos por Categoría

Gestión de Facturas

Método	Descripción	Archivo
<code>emitirFactura()</code>	Emite nueva factura	<code>facturaController.js</code>
<code>issueInvoice()</code>	Emite factura en inglés	<code>englishInvoiceController.js</code>
<code>validarFactura()</code>	Valida factura	<code>facturaController.js</code>
<code>validateInvoice()</code>	Valida factura (inglés)	<code>englishInvoiceController.js</code>
<code>enviarFactura()</code>	Envía a Hacienda	<code>facturaController.js</code>
<code>sendInvoice()</code>	Envía a Hacienda (inglés)	<code>englishInvoiceController.js</code>
<code>consultarFactura()</code>	Consulta específica	<code>facturaController.js</code>
<code>queryInvoice()</code>	Consulta específica (inglés)	<code>englishInvoiceController.js</code>
<code>listarFacturas()</code>	Lista con filtros	<code>facturaController.js</code>
<code>listInvoices()</code>	Lista con filtros (inglés)	<code>englishInvoiceController.js</code>

✔ **Validación**

Método	Propósito	Archivo
validateFactura()	Validación completa	facturaValidator.js
validateInvoice()	Validación completa (inglés)	englishInvoiceValidator.js
validateClave()	Formato de clave	facturaValidator.js
validateConsecutivo()	Formato consecutivo	facturaValidator.js
validateEmisor()	Datos del emisor	facturaValidator.js
validateIssuer()	Datos del emisor (inglés)	englishInvoiceValidator.js
validateReceptor()	Datos del receptor	facturaValidator.js
validateReceiver()	Datos del receptor (inglés)	englishInvoiceValidator.js
validateDetalle()	Líneas de servicio	facturaValidator.js
validateServiceDetail()	Líneas de servicio (inglés)	englishInvoiceValidator.js
validateTotales()	Resumen de totales	facturaValidator.js
validateInvoiceSummary()	Resumen de totales (inglés)	englishInvoiceValidator.js
validateBusinessLogic()	Lógica de negocio	facturaValidator.js

🔄 **Conversión de Idiomas**

Método	Conversión	Archivo
convertToSpanish()	Inglés → Español	englishInvoiceController.js
convertFromSpanish()	Español → Inglés	englishInvoiceValidator.js

💾 **Almacenamiento**

Método	Función	Archivo
saveInvoiceJSON()	Guarda JSON	invoiceStorage.js
saveInvoiceXML()	Guarda XML	invoiceStorage.js
getInvoice()	Recupera factura	invoiceStorage.js
listInvoices()	Lista facturas	invoiceStorage.js
markAsSent()	Marca como enviada	invoiceStorage.js
deleteInvoice()	Elimina factura	invoiceStorage.js
getStatistics()	Estadísticas	invoiceStorage.js

🔗 **Integración ATV**

Método	Función	Archivo
init()	Inicializa adaptador	atvAdapter.js
emitirComprobante()	Emite comprobante	atvAdapter.js
validarComprobante()	Valida comprobante	atvAdapter.js
enviarComprobante()	Envía comprobante	atvAdapter.js
consultarComprobante()	Consulta comprobante	atvAdapter.js
getStatus()	Estado del adaptador	atvAdapter.js

Logging

Método	Nivel	Archivo
info()	Información	logger.js
warn()	Advertencia	logger.js
error()	Error	logger.js
debug()	Debug	logger.js
logRequest()	HTTP Request	logger.js
logError()	Error estructurado	logger.js

Utilidades

Método	Función	Archivo
generateConsecutivo()	Genera consecutivo	filenames.js
generateTimestamp()	Genera timestamp	filenames.js
sanitizeFilename()	Sanitiza nombres	filenames.js
validateConsecutivoFormat()	Valida formato	filenames.js

Configuración

Método	Función	Archivo
isDevelopment()	Verifica desarrollo	config/index.js
isProduction()	Verifica producción	config/index.js

Rutas de API

Facturas en Español

Método HTTP	Ruta	Controlador
-------------	------	-------------

POST	/api/facturas/emitir	emitirFactura()
POST	/api/facturas/validar	validarFactura()
POST	/api/facturas/enviar	enviarFactura()
GET	/api/facturas/status	obtenerEstadoSistema()
GET	/api/facturas	listarFacturas()
GET	/api/facturas/:consecutivo	consultarFactura()
DELETE	/api/facturas/:consecutivo	eliminarFactura()

Invoices en Inglés

Método HTTP	Ruta	Controlador
POST	/api/en/invoices/issue	issueInvoice()
POST	/api/en/invoices/validate	validateInvoice()
POST	/api/en/invoices/send	sendInvoice()
GET	/api/en/invoices/health/check	Health check
GET	/api/en/invoices	listInvoices()
GET	/api/en/invoices/:consecutive	queryInvoice()

Rutas de Sistema

Método HTTP	Ruta	Función
GET	/health	Health check básico
GET	/info	Información del sistema
GET	/	Información principal

Flujos de Trabajo Principales

1. Emisión de Factura Completa

1. validateFactura() → Validación estructural
2. atvAdapter.init() → Inicialización
3. atvAdapter.emitirComprobante() → Emisión
4. saveInvoiceJSON() → Guardar JSON
5. saveInvoiceXML() → Guardar XML
6. atvAdapter.enviarComprobante() → Enviar a Hacienda
7. markAsSent() → Marcar como enviada

2. Validación por Payload

1. `validateFactura(payload)` → Validación estructural
2. `validateBusinessLogic()` → Validación de negocio
3. `atvAdapter.validarComprobante()` → Validación ATV
4. Retorno de resultado combinado

3. Consulta de Factura

1. `validateConsecutivo()` → Validar formato
2. `getInvoice()` → Buscar en storage
3. `atvAdapter.consultarComprobante()` → Estado ATV
4. Combinar y retornar datos

4. Conversión Inglés-Español

1. `validateInvoice()` → Validación en inglés
2. `convertToSpanish()` → Conversión a español
3. Procesamiento normal en español
4. Guardar datos bilingües

Patrones de Uso por Escenario

Desarrollo y Testing

- `logger.debug()` para información detallada
- `config.isDevelopment()` para funciones específicas
- `eliminarFactura()` para limpiar datos de prueba
- Modo `SIMULATED` para evitar calls reales

Validación Robusta

- `validateFactura()` + `validateBusinessLogic()`
- Validación en capas (estructura → negocio → ATV)
- Manejo granular de errores por campo

Integración Bilingüe

- `EnglishInvoiceValidator` para datos en inglés
- `convertToSpanish()` para compatibilidad ATV
- Almacenamiento de datos originales + convertidos

Monitoreo y Debugging

- `getStatistics()` para métricas de almacenamiento
- `getStatus()` para estado de ATV
- `logError()` para debugging estructurado
- `obtenerEstadoSistema()` para overview completo

Esta referencia rápida proporciona acceso inmediato a todos los métodos del proyecto, organizados alfabéticamente y por categorías funcionales para facilitar la navegación y el desarrollo.