



Ads by Google

Stop seeing this ad

Why this ad? ⓘ



Ads by Google

Stop seeing this ad

Why this ad? ⓘ



Ads by Google

Stop seeing this ad

Why this ad? ⓘ



Ads by Google

Stop seeing this ad

Why this ad? ⓘ



MINISTERIO
DE HACIENDA

Facturación Electrónica

El Salvador

eFacturaSV

DOCUMENTO TECNICO DE
LINEAMIENTOS DE INTEGRACIÓN



FACTURA ELECTRÓNICA EL SALVADOR
DOCUMENTO DE LINIAMIENTO DE INTEGRACIÓN

Nombre del Programa: Programa de Fortalecimiento de la Administración Tributaria
Oficina Patrocinadora: Banco Interamericano de Desarrollo BID
Número de Contrato: N°3852/OC-ES
Unidad Ejecutora: Viceministerio de Ingresos, a través de la Unidad Coordinadora del Programa
Autor: Programa de Fortalecimiento de la Administración Tributaria
Versión: 8.0
Fecha de Publicación: 21 de septiembre de 2021

Descargo de responsabilidad

Las opiniones del autor expresadas en esta publicación no reflejan necesariamente los puntos de vista del Banco Interamericano de Desarrollo BID.

Copyright © [2020] Banco Interamericano de Desarrollo. Este documento se basa en un trabajo financiado en parte por el Banco Interamericano de Desarrollo.

Las opiniones expresadas en esta guía son exclusivamente del autor y no reflejan necesariamente el punto de vista del Banco Interamericano de Desarrollo, de su Directorio Ejecutivo ni de los países que representa.



**Banco Interamericano
de Desarrollo**

Control de Versiones

Versión	Fecha
1.0	09/2020
2.0	11/2020
3.0	01/2021
4.0	02/2021
4.1	03/2021
5.0	
6.0	
7.0	
8.0	09/2021

Control de Cambios

Detalle Versión	Detalle de Cambios
Versión 4.0	Se modifica la sección de servicios, en particular los servicios de recepción, dado que se cambió el json que se envía como petición. Ahora se espera recibir todo el DTE firmado.
Versión 4.1	Se agrega sección de parámetro de Código QR, con la url que debe de ir dentro dicho código.
Versión 5.0	Se modifica la respuesta del servicio de recepción, agregando parámetro de observaciones.
Versión 6.0	Se amplía sección de firma electrónica (pag. 10) Se detalla proceso de contingencia (pag. 30) Se detalla política de reintentos (pag. 14)
Versión 7.0	Se agregan url de servicios productivos Se agrega servicio de anulación de DTE Se agrega detalle de plataforma de soporte vía ticket
Versión 8.0	Se agrega guía explicativa para configuración de Docker image de servicio de firmado

Glosario

Termino	Descripción
DTE	Documento Tributario Electrónico
JWS	JSON Web Signature (abreviado JWS) es un estándar propuesto por IETF para la firma de JSON arbitrario. ¹ Se utiliza como base para una variedad de tecnologías basadas en la web, incluyendo JSON Web Token.
JWT	JSON Web Token (abreviado JWT) es un estándar abierto basado en JSON propuesto por IETF (RFC 7519) para la creación de tokens de acceso que permiten la propagación de identidad y privilegios o claims en inglés
API	Una API es una interfaz de programación de aplicaciones (del inglés API: Application Programming Interface). Es un conjunto de rutinas que provee acceso a funciones de un determinado software.
Docker	Es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.
Cloud Computing	La computación en la nube, conocida también como servicios en la nube, informática en la nube, nube de cómputo o simplemente «la nube», es un paradigma que permite ofrecer servicios de computación a través de una red, que usualmente es internet.



Contenido



MINISTERIO
DE HACIENDA

DIRECCION GENERAL DE IMPUESTOS INTERNOS
UNIDAD COORDINADORA DEL PROGRAMA (UCP)

Introducción

El presente documento detalla información respecto a la infraestructura planificada para la plataforma de Factura Electrónica, así como detalles de los servicios web a utilizar para los procesos de recepción, validación y seguridad.

La arquitectura de factura electrónica se sustenta sobre los siguientes puntos básicos:

- Escalabilidad
- Performance

Tratando de dar cumplimiento a los puntos anteriores y luego de los análisis de la industria tecnológica a nivel mundial, y la imperante necesidad de proveer soluciones robustas y rápidas, la base de toda esta plataforma son los microservicios, de los que se tiene certeza agregarán un valor adicional no solo al resultado final de la implementación, sino también mejoras en la etapa de desarrollo.

Microservicios

La arquitectura de microservicios es una aproximación para el desarrollo de software que consiste en construir una aplicación como un conjunto de pequeños servicios, los cuales se ejecutan en su propio proceso y se comunican con mecanismos ligeros. Para la plataforma de factura electrónica, se ha decidido utilizar este paradigma, dado que nos ayuda a poder obtener mejores resultados en tiempo de implementación, despliegues (acortando los tiempos por medio de despliegues más eficientes) con sus correspondientes pruebas de control de calidad, enfocado en cada módulo/componente que necesite una actualización y/o ajuste, entre otras. Además, proveen el marco de trabajo adecuado para poder obtener los resultados esperados, performance y calidad requeridos de cara a los contribuyentes y el país en general.

Para proveer una infraestructura acorde a las necesidades de los microservicios, la plataforma de factura electrónica, se ejecutará bajo el modelo de infraestructura de nube. La infraestructura de nube describe aquellos elementos necesarios para el cloud computing, entre los que se incluyen el sistema de hardware, los recursos extraídos, el almacenamiento y los recursos de red.

Escalabilidad

Dado el análisis inicial del modelo de infraestructura en nube, se observa que este es uno de los puntos clave para la decisión de adoptar microservicios como paradigma de esta plataforma. Es necesario que todos los componentes garanticen el rendimiento y calidad adecuados en cada transacción, sin afectar a ningún contribuyente. La funcionalidad de una arquitectura desacoplada en N componentes, y sus correspondientes contenedores de ejecución garantizan que los recursos para cada servicio estén siempre disponibles para crecer y adaptarse a la demanda del mismo.

Flexibilidad

Los microservicios permiten la utilización de tecnologías diseñadas para sistemas de un alto volumen de tráfico, también permiten la optimización de contenedores de aplicación de manera individual sin afectar a 1 o N servicios con los cuales convive e interactúa dentro del ecosistema de la aplicación y/o servicio general. Estas nuevas tecnologías nos permiten de cierta manera poder controlar el comportamiento del servicio bajo cierta carga y/o demanda, de tal forma que la respuesta al cliente que consume, permanece optima durante cualquier periodo de tiempo, sin importar si estamos en picos de ejecución o no.

Para lograr este objetivo se implementará un orquestador de los servicios de factura electrónica, a través de Kubernetes, que en conjunto con Docker permitirán ofrecer servicios escalables y altamente adaptables.

Rendimiento

La utilización de aplicaciones y/o sistemas monolíticos es bastante aceptable cuando los ecosistemas (usuarios y datos) permanecen fijos o crecen de manera más lenta, además de tener cierto nivel similar de carga durante un periodo de tiempo. Sin embargo, en sistemas con validaciones y transacciones cercanas al tiempo real y una alta demanda, no podemos utilizar aplicaciones de este tipo, dado que se deben proveer muchas más características de procesamiento y optimizar de 1 o N servicios en nuestra arquitectura de manera dinámica y sin afectar a todo el ecosistema. Lo anterior solamente se obtiene desacoplando funcionalidades y abandonando los esquemas tradicionales de ejecución de aplicaciones.

Modelo de Base de Datos

La utilización de una base de datos relacional (SQL) se descartó luego del estudio de otras soluciones de factura electrónica a nivel latinoamericano, y la necesidad de que la plataforma cumpliera con ser rápida, eficiente y optima. Posterior al análisis se determinó que se necesitaba una arquitectura y/o modelo de base de datos que sea flexible y adaptable a las características de la solución que buscamos brindar. Acorde a lo expuesto anteriormente, se decidió utilizar como motor de base de datos, el esquema de bases NoSQL, utilizando para la plataforma de Factura Electrónica MongoDB, esta decisión se basa en los criterios siguientes:

Funcionalidad

Las bases de datos NoSQL, proveen una cantidad significativa de APIs para facilitar el desarrollo proveyendo tipos de datos según la necesidad de nuestro modelo.

Rendimiento

Por su construcción y diseño, las bases de datos no relacionales están creadas para optimizar los accesos y proveer tiempos de respuesta óptimos ante las consultas recibidas. Lo anterior es clave para la Plataforma de Factura Electrónica, dado que estará al centro de todas las operaciones de los contribuyentes

Escalabilidad

Las bases de datos NoSQL están diseñadas para escalar usando clústeres distribuidos de hardware en lugar de escalar añadiendo servidores caros y sólidos. Algunos proveedores de la nube manejan estas operaciones en segundo plano, como un servicio completamente administrado.

Contenedores

Como se ha comentado, la arquitectura del sistema estará orientada al uso de microservicios, para ello se hará uso de la tecnología de contenedores (Docker) para su ejecución; para el desarrollo de backend se usará JAVA como lenguaje en la capa de negocio mediante el uso de Spring Boot; por último, para la capa del cliente frontend, se utilizará Angular, y para almacenar los datos como se ha detallado se haría uso de la base NoSQL, MongoDB.

Cada contenedor ejecutara una actividad en específico, para la parte de seguridad se tendrá un contenedor que se encargue de gestionar y asegurar todos los demás contenedores, otro para encargarse de recibir los Documentos Tributarios Electrónicos (DTE) enviadas por los contribuyentes, y otro para validar los datos de los DTE.

Los contenedores de aplicaciones son herramientas que ayudaran a gestionar de manera eficiente los recursos, así como la optimización de los servicios de Factura Electrónica. Los puntos que se buscan cubrir mediante la utilización de contenedores son los siguientes:

- Escalabilidad
- Fácil Gestión
- Autonomía de ejecución

Buscando garantizar para cada contenedor un crecimiento automático en base a la utilización del componente (Auto escalamiento) después de cierto rango de utilización, y brindar una respuesta optima de los contribuyentes emisores de cara a la plataforma y sus procesos internos. El auto escalamiento de los contenedores, es estrictamente necesario y los puntos claves necesarios para dicho fin son los siguientes:

- **Escalamiento Horizontal:** Poder agregar mayor número de instancias a los contenedores.
- **Escalamiento Vertical:** Agregar mayor cantidad de recursos (ram, cpu, etc) a cada instancia dentro del contenedor.

Contenedor de Aplicaciones

Es importante mencionar como están elaborados los servicios de la plataforma de factura electrónica, de acuerdo a lo expuesto con anterioridad, tenemos diferentes contenedores con funcionalidades específicas para cada uno de ellos, por lo que en resumen las tecnologías en cada contenedor Docker sería el siguiente:

- Java 8
- Spring boot
- Spring Data
- JWT
- Angular

El framework para tareas de validación, seguridad y procesamiento de datos será Spring; y angular integrado con Spring para temas de front end. El modelo de web service utilizado, y que estará expuesto de cara a los contribuyentes será del tipo Rest.

Contenedor de Portal

Actualmente se han diseñado dos portales: a) informativo cuyo objetivo es proveer información de primera mano referente a factura electrónica, en este portal únicamente se colgaría información de cara a los contribuyentes y notas relacionadas al desarrollo del proyecto; b) operativo, donde los contribuyentes emisores y receptores pueden realizar gestiones administrativas y consultas de información sobre los documentos electrónicos emitidos.

Lineamientos de Seguridad

Los lineamientos descritos a continuación, únicamente se mencionan como referencia de la tecnología y los patrones que se utilizan dentro de la plataforma de eFacturaSV®, y no son una obligación para que los contribuyentes utilicen estas mismas tecnologías en sus desarrollos internos.

Seguridad Tecnológica de los Servicios

Referente al tema de seguridad, se gestionarán los accesos de los diferentes usuarios al sistema (usuarios de aplicación/web service y usuarios web para el portal de factura electrónica), mediante el uso del estándar de **JSON WEB TOKEN**.

¿Qué es JSON Web Token (JWT)?

JSON Web Token (JWT) es un estándar para la transmisión de información entre sistemas de forma segura y compacta, con el uso de objetos JSON ([documento RFC 7519](#)).

El JWT se compone de tres partes separadas por un punto ("."), y codificadas en Base64. Como se muestra en el ejemplo siguiente:

```
"eyJhbGciOiJIUzUxMiJ9.eyJqdGkiOiJzb2Z0dGVrSldUliwic3VljoIMDYxNC0xMzAzODYtMTM2LTQiLCJhdXRob3RpdGllcyI6WyJBZG91pbmizdHJhZG9yIl0sImIhdCI6MTU5NmM3ODU3NSwiZXhwIjoxNTk3Mzg0NTc1fQ.jCHm_2jyBNkshWph8H4m2DSEwul93frkVoB-f-m84bQRRBRTYkhUiHGYO3JpRawx7OBDSx2SU6bsaUaQeB1bpQ"
```



Ads by Google

Stop seeing this ad

Why this ad? ⓘ