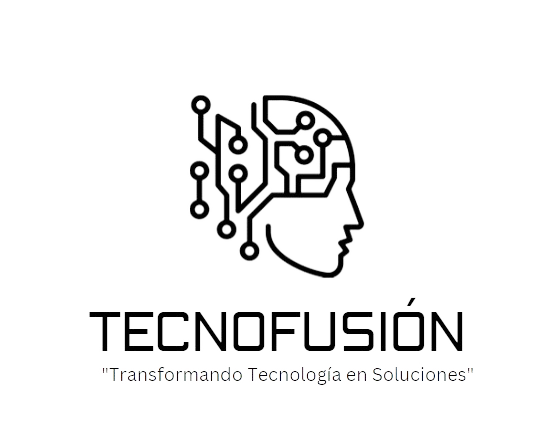
**Desarrollo de plataforma de ventas de entradas con identificación**

**“Documento de arquitectura del software”**

**Autores:**

Allain Alegre, Diego Arturo

**DBA (Administrador de Base de Datos)**

Asencios Gutierrez, Erick Fabrizio

**Arquitecto de Sofware (SA)**

Saavedra Valverde, Joel Ronaldo

**Tester (T) - Analista QA**

**Fecha de creación del documento:**

16/09/2023

**Resumen Ejecutivo:**

La plataforma de venta de entradas con identificación es un sistema diseñado para permitir la compra de boletos de eventos en línea. Este documento de arquitectura de software describe la estructura fundamental y las decisiones clave de diseño que respaldan esta plataforma.

**Objetivos principales:**

* Facilitar una comprensión clara y efectiva de la arquitectura del sistema entre todos los miembros del equipo de desarrollo, los stakeholders y otras partes interesadas.
* Registrar las decisiones de diseño clave que afectan la estructura y el comportamiento del sistema, proporcionando un historial de las elecciones tomadas.
* Servir como guía para el desarrollo del sistema, ayudando a los desarrolladores a implementar componentes y módulos de acuerdo con la arquitectura planificada.
* Permitir una evaluación temprana de la viabilidad técnica del proyecto al identificar posibles desafíos técnicos y requerimientos de recursos.
* Ayudar a identificar riesgos potenciales y proporcionar estrategias para mitigarlos en las etapas iniciales del proyecto.
* Definir una estructura que permita la escalabilidad y la mantenibilidad a largo plazo del sistema, considerando el crecimiento futuro y los cambios en los requisitos.
* Asegurar que la arquitectura esté alineada con los requisitos funcionales y no funcionales del sistema, garantizando que cumpla con las expectativas y necesidades del usuario.
* Fomentar el cumplimiento de estándares y buenas prácticas de desarrollo de software para garantizar la consistencia y la calidad del código.
* Ayudar en la estimación de costos y recursos necesarios para la implementación y el mantenimiento del sistema.

**Decisiones arquitectónicas clave:**

* Uso de una base de datos relacional para el almacenamiento de datos de eventos, usuarios y transacciones.
* Implementación de un sistema de autenticación y autorización robusto para garantizar la seguridad de los datos de los usuarios.
* Uso de microservicios para componentes críticos, como la gestión de inventario de boletos y el procesamiento de pagos.
* Implementación de un sistema de almacenamiento en caché para mejorar el rendimiento de consultas frecuentes.

**Tabla de contenidos**

[**1. Introducción 5**](#_pyw1ezkgpcym)

[1.1. Contexto y antecedentes 5](#_6p6o8bs3aotw)

[1.2. Motivo 5](#_coie75l31100)

[1.3. Audiencia prevista 6](#_h9wui8oo5o9g)

[**2. Visión General de la Arquitectura 7**](#_k3iid4hksyj3)

[2.1. Descripción general 7](#_r8mj7ts5gulz)

[**3. Estilos y Patrones Arquitectónicos 9**](#_o0ji4jjpzwy8)

[**4. Decisión de Tecnología 9**](#_syjyurcuz8b5)

[4.1. Lenguaje de programación, frameworks y herramientas 9](#_xdz49ic9r7gz)

[**5. Vista de Componentes 10**](#_rhtrwqufu283)

[5.1. Descripción de los componentes del sistema 10](#_iio3mwcicgze)

[5.2. Diagrama de componentes 11](#_npywmt16i9wx)

[**6. Vista de Despliegue 11**](#_d80kfnh422t)

[6.1. Descripción 11](#_iqf1lxh9x29o)

[**7. Vista de Secuencia 12**](#_8tqie0rl43u)

[7.1. Ejemplos de interacción 12](#_l2x7z9kvrhmt)

[**8. Consideraciones de Seguridad 14**](#_ecic0decwhit)

[8.1. Medidas de seguridad implementadas 14](#_ffiq93irfhn6)

[**9. Consideraciones de Mantenimiento 15**](#_2m15uafylw71)

[9.1. Consideraciones de Mantenimiento 15](#_dnsiue29qe7b)

[**10. Conclusiones 16**](#_h0o53v8t2rd9)

[**11. Referencias 16**](#_24lhwm2roawo)

# **Introducción**

## **Contexto y antecedentes**

La venta de entradas en línea es una práctica común que permite a los usuarios comprar boletos para conciertos, espectáculos deportivos, películas y otros eventos sin tener que hacer filas en taquillas físicas. Sin embargo, con la creciente preocupación por la seguridad y la autenticación de los asistentes, se ha vuelto necesario implementar un sistema de venta de entradas con identificación.

Algunos de los antecedentes incluyen:

* Fraude y reventa de entradas: Los revendedores a menudo compran grandes cantidades de entradas en segundos y las revenden a precios elevados, lo que afecta negativamente a los consumidores y al negocio legítimo de los eventos.
* Protección de datos personales: La recopilación y gestión de datos personales, como nombres, direcciones y números de teléfono, exige una mayor responsabilidad en términos de privacidad y protección de datos.
* Seguridad en eventos masivos: Los organizadores de eventos necesitan asegurarse de que las personas que asisten estén debidamente identificadas y puedan ser rastreadas en caso de emergencias.

## **Motivo**

***Seguridad del Evento:*** La identificación de los asistentes garantiza un mayor nivel de seguridad en eventos masivos, como conciertos, partidos deportivos y festivales.

***Prevención del Fraude de Entradas:*** Un sistema de identificación sólido reduce significativamente el fraude de entradas y la reventa ilegal. Los boletos pueden vincularse a una identidad única, lo que dificulta la falsificación y la reventa a precios exorbitantes.

***Mejora de la Experiencia del Asistente:*** Los sistemas de venta de entradas con identificación pueden simplificar la experiencia del asistente.

***Acceso a Información en Caso de Emergencia:*** En situaciones de emergencia, como evacuaciones o atención médica, el sistema de identificación puede ayudar a las autoridades y a los equipos de seguridad a localizar y comunicarse con los asistentes de manera más eficaz.

***Facilitación del Control de Acceso:*** Los sistemas de identificación pueden facilitar el control de acceso rápido y eficiente a los eventos, minimizando las colas y las demoras.

## **Audiencia prevista**

La audiencia prevista de un sistema de venta de entradas con identificación puede variar según el tipo de evento, su alcance y los objetivos específicos del sistema. Sin embargo, generalmente se espera que este tipo de sistema atienda a las siguientes audiencias clave:

***Organizadores de eventos***

Son una audiencia crucial para un sistema de venta de entradas con identificación. Necesitan una plataforma confiable para vender y gestionar boletos, así como para recopilar datos de asistentes para mejorar la planificación y la seguridad del evento.

***Asistentes a eventos***

Son usuarios finales del sistema. Deben poder comprar boletos de manera conveniente, recibir confirmaciones de compra y, en algunos casos, utilizar sus identificaciones para acceder al evento.

***Personal de Seguridad y Control de Acceso***

Utiliza el sistema para verificar la identificación de los asistentes y garantizar la seguridad en el evento.

***Autoridades y Cumplimiento Regulatorio***

Pueden requerir acceso a datos de identificación para garantizar la seguridad y el cumplimiento de las regulaciones de seguridad.

***Equipo de Operaciones de Eventos***

Pueden utilizar datos de identificación para coordinar la entrada, salida y experiencia general de los asistentes.

# **Visión General de la Arquitectura**

## **Descripción general**

***Interfaz de usuario***

Proporciona una interfaz web y posiblemente una aplicación móvil para que los usuarios busquen eventos, seleccionen boletos y realicen compras.

***Registro e inicio de sesión***

Permite a los usuarios registrarse, iniciar sesión y administrar sus cuentas, incluyendo la verificación de identidad cuando sea necesario.

***Gestión de eventos***

Administra información sobre eventos, como fechas, ubicaciones, tipos de boletos y precios.

***Carrito de compras***

Permite a los usuarios agregar y gestionar boletos en su carrito de compras antes de la compra final.

***Procesamiento de compras***

Realiza transacciones de compra, incluyendo la verificación de disponibilidad, el procesamiento de pagos y la generación de boletos electrónicos.

***Autenticación de identidad***

Verifica la identidad de los usuarios a través de métodos como la autenticación biométrica, contraseñas o códigos de autenticación de dos factores.

***Almacenamiento de eventos y usuarios***

Base de datos que almacena información sobre eventos, boletos disponibles, precios y ubicaciones e información de los usuarios registrados.

***Control de Acceso***

Garantiza que solo los usuarios autenticados tengan acceso a ciertas funciones y datos sensibles.

***Servicios de Verificación de Identidad***

Puede integrar servicios de verificación de identidad para autenticación biométrica u otros métodos de identificación avanzados.

***Uso de Tecnologías Escalables***

Utiliza tecnologías como contenedores y orquestación para garantizar la escalabilidad horizontal cuando se enfrenta a un alto tráfico de usuarios.

***Herramientas de Monitoreo***

Utiliza herramientas de monitoreo para supervisar el rendimiento del sistema, la seguridad y la disponibilidad.

# **Estilos y Patrones Arquitectónicos**

La arquitectura de software se refiere a la estructura que ha de tener un software, las partes que vamos a construir y la forma en la que las vamos a combinar y juntar para poder trabajar con ellas.

***Patrón Cliente Servidor***

Este patrón consiste en dos partes; un servidor y múltiples clientes . El componente del servidor proporcionará servicios a múltiples componentes del cliente. Los clientes solicitan servicios del servidor y el servidor proporciona servicios relevantes a esos clientes. Además, el servidor sigue escuchando las solicitudes de los clientes.

# **Decisión de Tecnología**

## **Lenguaje de programación, frameworks y herramientas**

Se utilizará un lenguaje capaz de tener la facilidad de uso y una amplia variedad de bibliotecas y frameworks que pueden simplificar el desarrollo de sistemas web. Django y Flask son dos frameworks populares que ofrecen sólido soporte para autenticación y seguridad.

**Frameworks para el Backend**

***Django***

Django es un framework de alto nivel que incluye características de seguridad sólidas, como la protección contra ataques comunes, la gestión de sesiones y una capa de autenticación robusta.

**Frameworks para el Frontend**

***React***

React es ampliamente utilizado en el desarrollo de interfaces de usuario interactivas y altamente responsivas. Es especialmente adecuado cuando se necesita crear una interfaz de usuario dinámica y atractiva que funcione en tiempo real para la selección de entradas y la autenticación.

**Herramientas de Autenticación y Seguridad**

***JSON Web Tokens (JWT)***

JWT es un estándar abierto que permite la autenticación segura y la transmisión de datos entre partes. Puede ser útil para crear tokens de sesión seguros después de que los usuarios se autentiquen.

# **Vista de Componentes**

## **Descripción de los componentes del sistema**

***Componente***

Símbolo para representar los módulos de un sistema (la interacción y la comunicación tienen lugar a través de interfaces).

***Paquete***

Un paquete combina varios elementos del sistema (por ejemplo clases, componentes o interfaces) en un grupo.

***Artefacto***

Los artefactos son unidades físicas de información (por ejemplo código fuente, archivos.exe, scripts o documentos) que se generan en el proceso de desarrollo o el tiempo de ejecución de un sistema o son necesarios para estos.

***Interfaz Ofrecida***

Símbolo para una o más interfaces claramente definidas que proporcionan funciones, servicios o datos del mundo exterior (el semicírculo abierto también se denomina enchufe o *socket*).

***Interfaz Requerida***

Símbolo de una interfaz necesaria para recibir funciones, servicios o datos del exterior (la notación del círculo con palo también se denomina *lollipop* o piruleta).

***Puerto***

Este símbolo indica un punto de interacción independiente entre un componente y su entorno.

***Relación***

Las líneas actúan como conectores e indican las relaciones entre los componentes.

***Relación de dependencia***

Conector especial para expresar una relación de dependencia entre los componentes del sistema (no siempre se indica).

## **Diagrama de componentes**

El diagrama de componentes proporcionará una visión general del sistema y documentará la organización de los componentes del sistema y sus relaciones y dependencias mutuas. Los diagramas de componentes proporcionarán una visión orientada a la ejecución, es decir, dará al desarrollador información sobre si el sistema funciona de forma coherente y cumple sus tareas y objetivos.

Los sistemas de software basados en componentes ahorran tiempo y costes en la fase de planificación y durante la implementación de los sistemas, ya que los sistemas existentes se pueden reutilizar. Además, sus módulos de software de probada eficacia reducen los riesgos y las causas de error, especialmente al implementar proyectos más complejos. Como también se ofrecen módulos de terceros que se pueden adquirir para implementar los sistemas, se puede compensar la falta de conocimientos técnicos internos.

# **Vista de Despliegue**

## **Descripción**

El diagrama de despliegue es un tipo de diagrama UML que sirve para representar la relación de un sistema, utilizando nodos para realizar la expresión gráfica del mismo. Son de gran utilidad para representar sistemas de hardware y software y poder observar cómo se vería su relación de forma real. En la aplicación para la venta de entradas utilizaremos el diagrama de despliegue.

A continuación mostraremos los elementos que contendrán el Diagrama de despliegue:

***Nodos***

Son representados como cubos y hacen referencia a un elemento físico general que engloba subelementos o subconjuntos de ejecución. Puede utilizarse para representar tanto un elemento de hardware como de software.

***Artefactos***

Los artefactos son representaciones mucho más concretas que hacen referencia a un elemento específico. En el caso de los diagramas de despliegue, podría tratarse de archivos ejecutables o archivos de configuración.

***Asociación de comunicación***

Por asociación de comunicación se hace referencia a las líneas que unen dos nodos. Estas líneas son las que permiten conocer la relación que existe entre uno y otro nodo de un diagrama de despliegue, de manera que son muy importantes para dar a conocer las relaciones del sistema.

***Dispositivos***

Los dispositivos también son nodos pero, en este caso, se emplean para representar los recursos programáticos de un sistema. Normalmente se trata de los servidores de una aplicación o sistema, así como también elementos similares.

***Especificación de despliegue***

Finalmente, la especificación de despliegue de un diagrama de despliegue es un tipo de archivo que contiene las indicaciones de desarrollo y funcionamiento de una aplicación o sistema. Por ello, debe ser muy concreta y responder a las características pertinentes y estrictamente necesarias.

# **Vista de Secuencia**

## **Ejemplos de interacción**

***Autenticación y Registro de Usuarios***

* Un usuario se registra en el sistema proporcionando sus datos personales.
* El sistema almacena la información del usuario en la base de datos.
* Cuando el usuario inicia sesión, el sistema verifica las credenciales ingresadas (nombre de usuario y contraseña).
* Si las credenciales son correctas o la autenticación es exitosa, el sistema genera un token de sesión y permite el acceso a las funciones protegidas.

***Búsqueda y Visualización de Eventos***

* El usuario busca eventos disponibles utilizando filtros como ubicación, fecha o categoría.
* El sistema consulta la base de datos de eventos y devuelve resultados coincidentes.
* Los eventos se muestran al usuario en la interfaz de usuario, junto con detalles como fecha, ubicación y opciones de boletos disponibles.

***Selección de Boletos y Carrito de Compras***

* El usuario selecciona un evento al que desea asistir.
* El sistema agrega el boleto al carrito de compras, donde se almacena temporalmente.
* El usuario puede revisar y editar el contenido del carrito de compras antes de proceder al pago.

***Procesamiento de Compras y Pagos***

* El usuario inicia el proceso de compra desde el carrito de compras.
* El sistema verifica la disponibilidad y la información de los eventos en tiempo real para evitar duplicados.
* Luego, el sistema dirige al usuario a una pasarela de pago segura donde se ingresan los detalles de la tarjeta de crédito u otros métodos de pago.
* Una vez que se completa la transacción, el sistema genera boletos electrónicos y los vincula a la cuenta del usuario.

***Generación y Entrega de Boletos Electrónicos***

* Después de la compra, el sistema genera boletos electrónicos en formato digital.
* Los boletos electrónicos se almacenan en el sistema y están disponibles para su descarga o visualización en el sitio web del usuario.

***Acceso al Evento***

* En el día del evento, el usuario presenta el boleto electrónico en la entrada.
* Un escáner de códigos QR o un dispositivo de lectura verifica la autenticidad de los boletos electrónicos.
* Si el boletos es válido, se permite el acceso al evento.

***Historial de Compras y Cuenta de Usuario***

* El sistema actualiza la información del usuario y mantiene un registro de todas las transacciones pasadas.

# **Consideraciones de Seguridad**

## **Medidas de seguridad implementadas**

***Autenticación Fuerte***

Utilizar métodos de autenticación robustos, como contraseñas seguras y autenticación de dos factores, para verificar la identidad de los usuarios al iniciar sesión o realizar transacciones.

***Cifrado de Datos***

Encriptar todos los datos sensibles, como información de identificación y datos de pago.

***Seguridad en Pagos***

Utilizar pasarelas de pago confiables y seguras para el procesamiento de transacciones financieras.

***Gestión de Sesiones Segura***

Implementar medidas de seguridad para gestionar y proteger las sesiones de usuario, como expiración de sesiones inactivas.

***Control de Acceso***

Implementar políticas de control de acceso para garantizar que los usuarios solo tengan acceso a las funciones y recursos para los que están autorizados.

***Respuesta a Incidentes***

Establecer un plan de respuesta a incidentes para manejar posibles violaciones de seguridad de manera eficiente y efectiva.

***Privacidad de Datos***

Cumplir con regulaciones de privacidad de datos, y proteger adecuadamente la información personal de los usuarios.

***Pruebas de Seguridad***

Realizar pruebas de seguridad regulares, como pruebas de penetración y evaluaciones de seguridad, para identificar y corregir vulnerabilidades.

# **Consideraciones de Mantenimiento**

## **Consideraciones de Mantenimiento**

***Documentación detallada***

Mantener documentación completa y actualizada es fundamental para facilitar el mantenimiento. Esto incluye documentación de la arquitectura, el código fuente, los flujos de trabajo y los procedimientos de mantenimiento.

***Pruebas automatizadas***

Implementar pruebas automatizadas, como pruebas unitarias y pruebas de integración, que cubran de manera exhaustiva las funcionalidades críticas del sistema.

***Gestión de versiones***

Utilizar sistemas de control de versiones para rastrear cambios en el código y facilitar la colaboración en el desarrollo.

***Parches y actualizaciones***

Establecer un proceso estructurado para la aplicación de parches de seguridad y actualizaciones de software.

***Seguimiento de problemas***

Utilizar sistemas de seguimiento de problemas para registrar, priorizar y asignar problemas de mantenimiento y correcciones de errores.

***Estrategia de respaldo y configuración***

Implementar una sólida estrategia de respaldo y recuperación de datos para garantizar que la información crítica esté segura y disponible en caso de fallos o desastres.

# **Conclusiones**

Una plataforma de venta de entradas con identificación es una solución tecnológica esencial para la gestión segura y eficiente de la venta de boletos para eventos.

En última instancia, una plataforma de venta de entradas con identificación bien diseñado y gestionado no solo facilita la compra de boletos para los usuarios, sino que también contribuye a la seguridad, la privacidad y la eficiencia en la gestión de eventos y lugares de entretenimiento. La inversión en seguridad y calidad es esencial para construir una base sólida para el éxito a largo plazo del sistema.

# **Referencias**

Pérez, M. A. L. (2023). Reventa de entradas: regulación actual y futura. *Sympathy for the Lawyer*. <https://sympathyforthelawyer.com/hub/reventa-de-entradas-regulacion-actual-y-futura/>

*Venta de entradas por internet. Cuidado con las reventas*. (2022, 30 mayo). Comunidad de Madrid. <https://www.comunidad.madrid/servicios/consumo/venta-entradas-internet-cuidado-reventas>

LegalToday. (2020). La reventa de entradas por internet, al filo de la ley. *Legal Today*. <https://www.legaltoday.com/practica-juridica/derecho-civil/nuevas-tecnologias-civil/la-reventa-de-entradas-por-internet-al-filo-de-la-ley-2017-03-21/>

*Política sobre la reventa de entradas para espectáculos*. (s. f.). eBay. <https://www.ebay.es/help/policies/prohibited-restricted-items/poltica-sobre-la-reventa-de-entradas-para-espectculos?id=4309>

Datos. (2023, 12 mayo). Las 9 medidas de seguridad informática básicas para proteger la empresa | Datos 101. *Datos 101*. <https://www.datos101.com/blog/medidas-de-seguridad-informatica/>

*Seguridad en el ciclo de vida de desarrollo del software*. (s. f.). <https://www.redhat.com/es/topics/security/software-development-lifecycle-security>