**UNIVERSIDAD AUTONOMA**

**“GABRIEL RENE MORENO”  
FACULTAD DE CIENCIAS DE LA COMPUTACIÓN Y TELECOMUNICACIONES**



**­­­**

**GESTIÓN DE CONFIGURACIÓN DE SOFTWARE**

**MATERIA :** INGENIERÍA DE SOFTWARE II

**DOCENTE :** ING. ROLANDO MARTINEZ

**INTEGRANTES :** ROMÁN YÁÑEZ PAOLA ANDREA

VALDEZ COSSIO EDSON GABRIEL

**FECHA : 23/06/2016**

**SANTA CRU­­­­Z – BOLIVIA**

Contenido

[1. INTRODUCCION 3](#_Toc454323199)

[2 OBJETIVOS 4](#_Toc454323200)

[2.1 OBJETIVO GENERAL 4](#_Toc454323201)

[2.2 OBJETIVOS ESPECIFICOS 4](#_Toc454323202)

[3 METODOLOGIA 4](#_Toc454323203)

[4 ALCANCE 4](#_Toc454323204)

[4.1 REQUERIMIENTOS FUNCIONALES 5](#_Toc454323205)

[5 FLUJO DE TRABAJO: REQUISITOS 5](#_Toc454323206)

[5.1 Identificar actores y casos de uso 5](#_Toc454323207)

[5.1.1 Actores 5](#_Toc454323208)

[5.1.2 Casos de Uso 6](#_Toc454323209)

[5.2 Priorizar Casos de Uso 6](#_Toc454323210)

[5.3 Detallar Casos de Uso 7](#_Toc454323211)

[5.4 DIAGRAMA GENERAL DE CASOS DE USO 10](#_Toc454323212)

[5.5 PROTOTIPOS 10](#_Toc454323213)

[6.FLUJO DE TRABAJO: ANALISIS 11](#_Toc454323214)

[6.1 Analisis de la arquitectura 11](#_Toc454323215)

[6.2 Analisis de Casos de Uso 11](#_Toc454323216)

[6.2.1 Diagrama de comunicación 11](#_Toc454323217)

[6.3 Analisis de clases 11](#_Toc454323218)

[7. FLUJO DE TRABAJO: DISEÑO 11](#_Toc454323219)

[7.1 Diseño de clases 11](#_Toc454323220)

[7.2 Diseño de Casos de Uso 11](#_Toc454323221)

[7.2.1 Diagramas de Secuencia 11](#_Toc454323222)

[7.3 Diseño de Base de Datos 11](#_Toc454323223)

[7.3.1 Diseño Conceptual 11](#_Toc454323224)

[7.3.2 Diseño Logico 11](#_Toc454323225)

[7.3.3 Diseño Fisico 11](#_Toc454323226)

[8. REFENCIAS 11](#_Toc454323227)

# INTRODUCCION

En el presente documento detallaremos la documentación sobre el proyecto titulado “Venta De Tickets Para Eventos”, La presente será una aplicación web que estará basada en la Tecnología SAAS (del inglés: **S**oftware **a**s **a** **S**ervice).

Al decir que desarrollaremos la aplicación con SAAS debemos ofrecer una plataforma en línea a los organizadores de eventos para que ellos puedan contar con los servicios y administrar sus eventos.

Además de desarrollar el software y una documentación estará versionada en una plataforma de GITHUB se podrá descargar el documento los modelos y el software en el siguiente repositorio <https://github.com/paolatecno/prodoc>.

Para tener una constancia de los modelos y el código crearemos nuestros modelos con Enterprise Architect 8 con los cuales generaremos el 65% del código del software.

En cuanto a la seguridad implementaremos unos mecanismos y herramientas de software para aplicar métodos de pruebas a los siguientes niveles:

* Pruebas a Nivel Funcional: Estaremos creando casos de prueba para poder testear el software buscando errores de funcionalidad. Usaremos pruebas de caja negra y para ciertos algoritmos importantes usaremos pruebas de caja blanca
* Pruebas de tiempo Real: Emplearemos herramientas para poder medir el tiempo de respuesta del software.
* Pruebas de Seguridad: Usaremos servicios en los cuales probaremos al menos con 3 de ellos. Buscando inseguridad en cuanto inyecion SQL, colapso por boots, accesos no autorizados.

# 2 OBJETIVOS

## 2.1 OBJETIVO GENERAL

Desarrollar un sistema web el cual permita reservar entradas a eventos a través de tickets electrónicos.

## 2.2 OBJETIVOS ESPECIFICOS

* Identificar las funciones que el software debe cumplir.
* Obtener toda la información precisa, mediante investigación sobre la venta de productos a través de internet.
* Diseñar una base de datos con la información y datos recopilados en la investigación.
* Realizar la captura de requisitos; analizando los requerimientos del software, para definir los requisitos a través de casos de uso.
* Diseñar la arquitectura de acuerdo al análisis realizado; diseñando los casos de uso, las clases, los subsistemas y las interfaces.
* Implementar el software según las especificaciones del diseño.
* Realizar las pruebas pertinentes para asegurar el correcto funcionamiento del software.

# 3 METODOLOGIA

* La metodología a usar será el Proceso Unificado de Desarrollo de Software.

# 4 ALCANCE

## 4.1 REQUERIMIENTOS FUNCIONALES

El organizador podrá crearse una cuenta en nuestra plataforma una vez ingrese a la plataforma podrá crear eventos los cuales detallara la ubicación (ciudad, dirección) en qué hora comienza el evento y que hora terminara. Tendrá la opción de poder decir que el evento es todo el día. Además debe colocar información de cuantos tickets estarán disponibles para dicho evento el precio de los tickets en esta sección además especificara si existen tickets de diferente valor.

El organizador podrá generar reportes de quienes, reservaron sus tickets además en el reporte se especificara cuantos tickets, de que tipos y cuando los reservaron.

Los organizadores podrán dar a conocer en que sección se ofrecen los eventos para que sus clientes puedan acceder a la reserva de un ticket.

Los asistentes del evento pueden reservar un evento dando información como nombre, apellido, dirección, número de teléfono, correo electrónico tendrá la posibilidad de reservar más de un ticket dependiendo de la configuración del evento. El asistente recibirá un email a su correo con información del evento y un código de ticket único.

Una vez realizada una reservación el organizador recibirá un email con los datos sobre la reserva con los datos del asistente en el email.

# 5 FLUJO DE TRABAJO: REQUISITOS

## 5.1 Identificar actores y casos de uso

### 5.1.1 Actores

Usuario

Administrador

### 5.1.2 Casos de Uso

1. Crear Ticket
2. Reservar ticket
3. Gestionar evento
4. Gestionar organizadores

## 5.2 Priorizar Casos de Uso

|  |  |  |  |
| --- | --- | --- | --- |
| **Nro.** | **Caso de uso** | **Actores** | **Prioridad** |
| 1  2  3  4 | Crear Ticket  Reservar ticket  Gestionar evento  Gestionar organizadores | Administrador  Usuario  Administrador  Administrador | A  B  A  B |

## 5.3 Detallar Casos de Uso

1. Crear Ticket



|  |  |
| --- | --- |
| **Caso de uso: Crear ticket** | **nro:1** |
| **Descripción:** el Administrador crea ticket | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Precondición:** ingresar como Administrador | |
| **Proceso:**   1. El caso de uso comienza cuando el Administrador selecciona la opcion“ticket”    1. Si el administrador selecciona “Registrar ticket”    2. El administrador Introduce datos del ticket    3. El sistema crea un nuevo ticket | |
| **Poscondición:** ninguna | |
| **Flujos alternativos**:   * 1. Datos Inválidos | |

1. Reservar ticket



|  |  |
| --- | --- |
| **Caso de uso: Reservar ticket** | **nro:2** |
| **Descripción:** el Usuario reserva el ticket al evento que desea asistir | |
| **Actores:** Usuario | |
| **Iniciador:** Usuario | |
| **Precondición:** ingresar como Usuario | |
| **Proceso:**   1. El caso de uso comienza cuando el Usuario selecciona la opcion“reservar ticket”    1. El usuario selecciona el evento al que desea asistir    2. El usuario reserva el ticket para algún evento    3. El sistema guarda la reservación. | |
| **Poscondición:** ninguna | |

1. Gestionar evento



|  |  |
| --- | --- |
| **Caso de uso: Gestionar Evento** | **nro:3** |
| **Descripción:** el Administrador gestiona el evento | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Precondición:** ingresar como Administrador | |
| **Proceso:**   1. El caso de uso comienza cuando el Administrador selecciona la opcion“Evento”    1. Si el administrador selecciona “Registrar evento”    2. El administrador Introduce datos del evento    3. El sistema crea un nuevo evento 2. Si El Administrador selecciona “Modificar evento”   2.1 El administrador selecciona el evento  2.2 El administrador modifica datos del evento  2.3 El sistema Guardar la actualizaciones del evento  3. Si El Administrador selecciona “Eliminar evento”  3.1 El administrador selecciona el evento  3.3 El sistema elimina el evento | |
| **Poscondición:** ninguna | |
| **Flujos alternativos**:  1.2 Datos Inválidos | |

4.Gestionar organizadores

****

|  |  |
| --- | --- |
| **Caso de uso: Gestionar organizadores** | **nro:4** |
| **Descripción:** el Administrador gestiona a los organizadores de los eventos | |
| **Actores:** Administrador | |
| **Iniciador:** Administrador | |
| **Precondición:** ingresar como Administrador | |
| **Proceso:**   1. El caso de uso comienza cuando el Administrador selecciona la opcion“organizadores”    1. Si el administrador selecciona “Registrar organizadores”    2. El administrador Introduce datos de la organizadores    3. El sistema crea una nueva organizadores 2. Si El Administrador selecciona “Modificar organizadores”   2.1 El administrador selecciona la organizadores  2.2 El administrador modifica datos de la organizadores  2.3 El sistema Guardar la actualizaciones de la organizadores  3. Si El Administrador selecciona “Eliminar organizadores”  3.1 El administrador selecciona a la organizadores  3.3 El sistema elimina a la organizadores | |
| **Poscondición:** ninguna | |
| **Flujos alternativos**:  1.2 Datos Inválidos | |

# 5.4 DIAGRAMA GENERAL DE CASOS DE USO



# PROTOTIPOS

# 6.FLUJO DE TRABAJO: ANALISIS

## 6.1 Analisis de la arquitectura

## 6.2 Analisis de Casos de Uso

### 6.2.1 Diagrama de comunicación

## 6.3 Analisis de clases

# 7. FLUJO DE TRABAJO: DISEÑO

## 7.1 Diseño de clases

## 7.2 Diseño de Casos de Uso

### 7.2.1 Diagramas de Secuencia

## 7.3 Diseño de Base de Datos

### 7.3.1 Diseño Conceptual

### 7.3.2 Diseño Logico

### 7.3.3 Diseño Fisico

# 8. REFENCIAS

Jacobson Ivar, Booch Grady, Rumbaugh James. (2005)“El Lenguaje Unificado de Modelado”, (2da Edición). Impreso en España, Madrid: Pearson Educación S.A, Addison Wesley.

Jacobson Ivar, Booch Grady, Rumbaugh James (1999). *El Proceso Unificado de Desarrollo de Software.* (S. Sanchez, M. A. Sicilia, C. Canal, & F. J. Durán, Trads.) Madrid, España: Addison Wesley.