2018

Sistema para la venta de entradas

www.



Elias, Federico.

Maduro, Alan.

Felice, Karina

Metodologia de Sistemas I

16/11/2018

INDICE

| 1. Introducción | 3 |
|--|----|
| 1.1 Propósito | 3 |
| 1.2. Alcance | 3 |
| 1.3. Referencias | 3 |
| 2. Descripción general del sistema | 4 |
| 2.1. Perspectiva del producto | 4 |
| 2.2. Objetivos del sistema | 4 |
| 2.3. Funciones del sistema | 5 |
| 3. Definición de Requisitos del Sistema | 6 |
| 3.1. Definición de Requisitos Funcionales | 6 |
| 3.2. Definición de Requisitos No Funcionales | 14 |
| 4. Especificación de Requisitos del Sistema | 15 |
| 5. Diagrama de caso de uso | 16 |
| 5.1 Especificación caso de uso Cargar Calendario a un Event | 20 |
| 5.2. Especificación Caso de Uso Comprar una entrada para un evento , | |
| calendario y tipo de asiento especifico | 22 |
| 6. Diagrama de Clases con la arquitectura implementada | 24 |
| 7. Diagrama de entidades y relaciones | 26 |

1. Introducción

Este documento contiene la descripción detallada de los diferentes requisitos de software que debe cumplir el sistema para permitir a los clientes comprar tickets para los eventos que la empresa organiza a través de un sitio web.

1.1. Propósito

El propósito de este documento es presentar de manera formal la especificación de requisitos de este sistema, para su discusión y aceptación, por parte de los usuarios que utilizaran dicho sistema. En esta especificación se detallan los requerimientos funcionales, las restricciones y los atributos de calidad que deberá satisfacer el sistema.

El documento va dirigido en primera instancia a los usuarios directos de este sistema, es decir aquellas personas que se encargan de administrar los eventos. En segunda instancia, al equipo de desarrollo de este sistema.

1.2. Alcance

Este sistema proveerá las herramientas necesarias para la gestión de todos los eventos que la empresa ofrece así como también permitirá la compra de tickets a los clientes y todo mediante la web. El sistema llevara como nombre GoToEvent, el cual permitirá:

A los administradores:

- ✓ Administrar los artistas, eventos, categorías, las plazas, los tipo de plazas, los lugares y los calendarios.
- ✓ Consultar cantidades vendidas y remanentes de entradas para un determinado evento.
- ✓ Consultar totales vendidos en pesos.
- ✓ No se ha plantea el escenario de cancelación de un evento.

A los clientes:

- ✓ Registrarse.
- ✓ Consultar eventos.
- ✓ Seleccionar eventos para su compra.
- ✓ Consultar los tickets adquiridos.

1.3. Referencias

Para la elaboración de este escrito se han seguido los lineamientos presentados en los siguientes documentos:

• IEEE-STD-830-1998: Especificaciones de los Requisitos de Software.

2. Descripción General del Sistema

2.1. Perspectiva del producto

Este sistema de venta de tickets deberá ser visualmente atractivo con un diseño intuitivo y fácil de usar con botones de acción bien definidos y visibles para que el usuario los encuentre fácilmente y sepa identificarlos.

Debe asegurar los siguientes aspectos: estabilidad, seguridad, soporte y escalabilidad.

Estabilidad para poder estar operativo las 24 horas todos los días del año. Siempre se producirán errores y fallos, pero deben ser los menos posibles, debe ser lo más estable que técnicamente sea posible.

Seguridad sobre los datos, no debe poner en riesgo la seguridad de los visitantes al sitio web.

Soporte en donde la plataforma es capaz de proporcionar el mejor servicio de ayuda ya sea directamente, a través de una amplia comunidad activa de foros u otros.

Escalabilidad para que la plataforma sea capaz de asimilar y absorber el crecimiento del negocio en caso de ser necesario (en funcionalidades, servicios, facturación, etc).

2.2. Objetivos del Sistema

El sistema GoToEvent deberá cumplir con los siguientes objetivos:

- Permitir que los usuarios puedan realizar a través del sistema, las diversas solicitudes para registrarse, consultar eventos por fecha, categoría o artista.
- ➤ Permitir a los clientes generar la compra de entradas a eventos así como también consultar las ya adquiridas.
- ➤ Permitir a los administradores gestionar todos los datos necesarios para incorporar un nuevo evento al sistema. El sistema deberá hacer todas las validaciones necesarias a medida que se van ingresando los datos e ir informando los problemas que vayan surgiendo.
- Permitir a los administradores realizar todo tipo de consultas con respecto a los eventos ya cargados

2.3. Funciones del Sistema

Entre las funciones generales que debe realizar el sistema podemos enunciar las siguientes:

➤ A los administradores:

- ✓ Administrar los artistas.
- ✓ Administrar los eventos.
- ✓ Administrar categoría de evento determinada (obra teatral, concierto, exposición, festival de música) y tipo de plaza tipo (general, platea lateral, platea central, palco, etc).
- ✓ Administrar la plaza evento que contiene la cantidad disponible , el remanente y el precio.
- ✓ Administrar el tipo de plaza (platea, palco, césped, etc.) que se sacan a la venta en un determinado lugar en donde ocurre un determinado evento.
- ✓ Administrar el lugar del evento que contiene la capacidad y el nombre del lugar (monumental, gran rex, etc).
- ✓ Administrar el calendario que son las distintas fechas que tiene un artista en diferentes lugares.
- ✓ Consultar cantidades vendidas y remanentes de entradas para un determinado evento.
- ✓ Consultar totales vendidos en pesos (por fecha y/o categoría de evento).

A los clientes:

- ✓ Permitir que un usuario se registre mediante un mail y una clave o vía facebook
- ✓ Consultar eventos por fecha del evento.
- ✓ Consultar eventos por categoría del evento.
- ✓ Consultar eventos por determinado artista.
- ✓ Seleccionar eventos para su compra detallando cantidad de plazas y tipo, estas se irán agregando a un carrito. Al confirmarse la misma se almacena la compra y se generan los tickets, enviando una copia por email. El ticket tendrá un número y un código QR que le permitirá ingresar al evento.
- ✓ Consultar los tickets adquiridos ordenados por fecha de evento.

3. Definición de Requisitos del Sistema

En este apartado se presentan los requisitos funcionales y no funcionales que deberán ser satisfechos por el sistema.

Contiene los requisitos a un nivel de detalle suficiente como para permitir a los diseñadores diseñar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas planificar y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aquí especificado describirá comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas

El documento es perfectamente legible por personas de muy distintas formaciones e intereses. Todos los requisitos están unívocamente identificados. Son correctos ya que reflejan las necesidades reales, no son ambiguos porque cada requisito tiene una sola interpretación y son completos gracias a que todos los requisitos relevantes han sido incluidos.

3.1. Definición de Requisitos Funcionales

La siguiente tabla muestra la lista de requisitos funcionales del sistema GoToEvent. Cada requisito contiene un identificador donde las dos primeras letras indican que es un requisito funcional y un número que corresponde a la secuencia de los requisitos. Las columnas nombre y descripción definen el requisito, la columna usuario indican quien lo utiliza. Finalmente, la columna hecho por indica la persona que realizo el requisito.

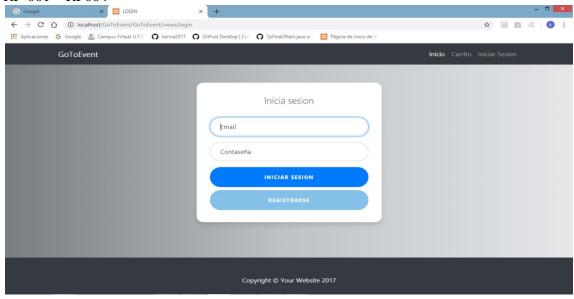
Lista de Definición de Requisitos Funcionales de GoToEvent

| Id | Nombre del | Descripción del requisito | Hecho | Usuario |
|-----------|----------------|---|-------|---------------|
| Requisito | Requisito | | por | |
| RF-001 | Registrar | El sistema debe permitir que el usuario se | | Usuario |
| | usuario | registre con su email y una clave o vía su | | Anónimo |
| | | cuenta de facebook | | |
| RF-002 | Verificar rool | El sistema verifica el tipo de usuario y de | | Cliente/ |
| | | acuerdo a esto le da acceso al menú | | Administrador |
| | | correspondiente dependiendo si es | | |
| | | administrador o cliente | | |
| RF-003 | Login | El sistema permite verificar que los datos | | Cliente/ |
| | | ingresados se correspondan con los datos | | Administrador |
| | | registrados en la base de datos y luego | | |
| | | verifica el tipo de usuario (RF-002) | | |
| RF-004 | Consultar | El sistema permite ingresar una fecha y | | Cliente |
| | eventos por | lista todos los eventos para esa fecha | | |
| | fecha | | | |
| RF-005 | Consultar | El sistema permite seleccionar una | | Cliente |
| | eventos por | categoría y lista todos los eventos de esa | | |
| | categoría | categoría | | |
| RF-006 | Consultar | El sistema permite seleccionar un artista y | | Cliente |
| | eventos por | lista todos los eventos en los cuales | | |
| | artista | interviene ese artista | | |

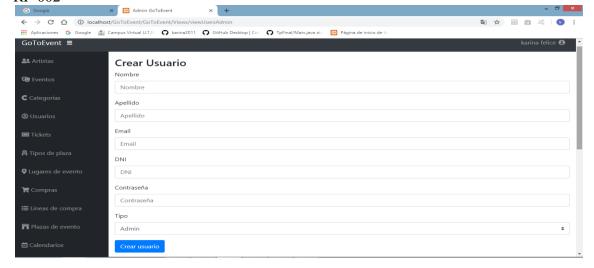
| RF-007 | Consultar | El sistema permite ver los tickets | Cliente |
|--------|---|---|---------------|
| | tickets adquiridos | adquiridos ordenados por fecha | |
| RF-008 | Comprar entrada para evento | El sistema permite seleccionar el evento que desea comprar. Las compras se agrupan por tipo de plaza, es decir el sistema le lista las opciones de tipos de plaza disponibles para el evento, al seleccionar una se debe indicar la cantidad y luego al presionar el botón comprar genera una línea en el carrito y vuelve al evento para comprar más entradas con otro | Cliente |
| | | tipo de plaza o para finalizar la compra. Cuando finaliza la compra el sistema comprueba que haya disponibilidad de plaza, en caso afirmativo, se generan un tickets con número y código QR (RF-016) por cada entrada y se envía por mail los comprobantes. | |
| RF-009 | ABM de artista | El sistema permite dar de alta un nuevo artista para lo cual se necesita nombre, apellido y dni o modificarlo o borrar uno ya existente | Administrador |
| RF-010 | ABM de eventos | El sistema permite dar de alta un evento para lo cual se necesita el nombre del evento, una imagen del evento y la categoría a la cual pertenece. También permite modificar o borrar uno ya existente | Administrador |
| RF-011 | ABM de categorías | El sistema permite dar de alta una categoría para lo cual se necesita el nombre de la categoría o modificarla o borrar una ya existente | Administrador |
| RF-012 | ABM de usuarios y administradores | El sistema permite dar de alta un usuario cliente o administrador para lo cual se necesita el nombre, apellido, email, dni, tipo y contraseña. También permite modificar o borrar uno ya existente | Administrador |
| RF-013 | ABM de tipos de plaza | El sistema permite dar de alta un tipo de plaza para lo cual se necesita el nombre. También permite modificar o borrar una ya existente | Administrador |
| RF-014 | ABM de los lugares para los eventos | El sistema permite dar de alta un lugar evento para lo cual se necesita el nombre y la capacidad total. También permite modificar o borrar uno ya existente | Administrador |
| RF-015 | ABM de plaza evento | El sistema permite dar de alta una plaza evento para lo cual se necesita precio, cantidad disponible y remanente. También permite modificar o borrar uno ya existente | Administrador |
| RF-016 | Generar código QR | El sistema permite generar un código QR para cada ticket vendido para cada evento. | Cliente |

| RF-017 | Crear | El sistema permite crear un calendario para | Administrador |
|--------|-----------------|--|---------------|
| | calendario | un evento para una determinada fecha, | |
| | | para lo cual previamente debe haber dado | |
| | | el alta del evento (RF-010), el o los artistas | |
| | | (RF-009) y el lugar del evento (RF-014). | |
| | | El sistema le permite seleccionar mediante | |
| | | un desplegable el evento, seleccionar el o | |
| | | los artistas involucrados y de otro | |
| | | desplegable elige el lugar del evento. Al | |
| | | presionar el botón de crear calendario pasa | |
| | | a otra plantilla que le permite ingresar las | |
| | | plazas que tendrá el evento (RF-015) | |
| RF-018 | Retornar el | El sistema permite ingresar una fecha y | Administrador |
| | monto total por | retorna el monto total recaudado por la | |
| | fecha | venta de las entradas para todos los | |
| | | eventos en la fecha especificada. | |
| RF-019 | Retornar el | El sistema permite ingresar una categoría y | Administrador |
| | monto total por | retorna el monto total recaudado por la | |
| | categoría. | venta de las entradas para todos los | |
| | | eventos que pertenecen a la categoría | |
| | | especificada. | |

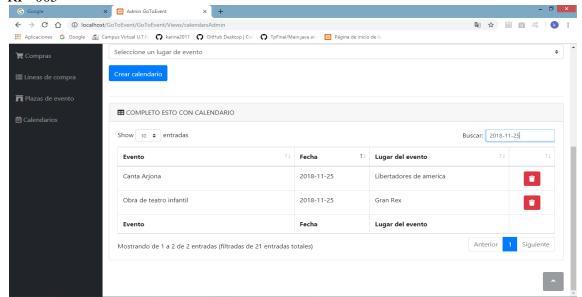
RF-001 - RF004



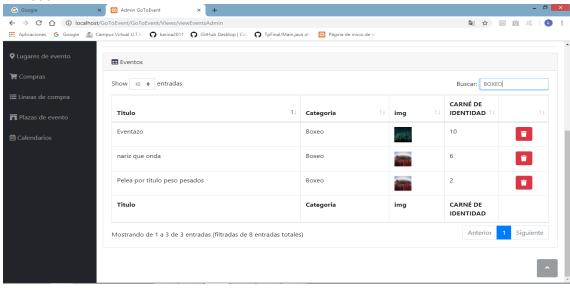
RF-002



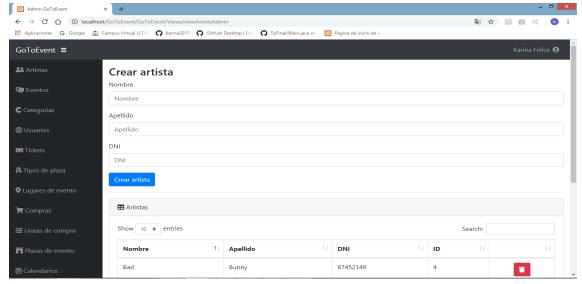
RF- 005



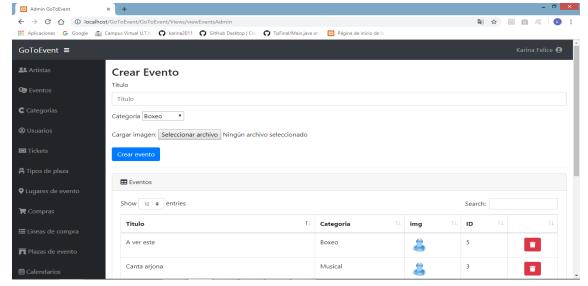
RF-006

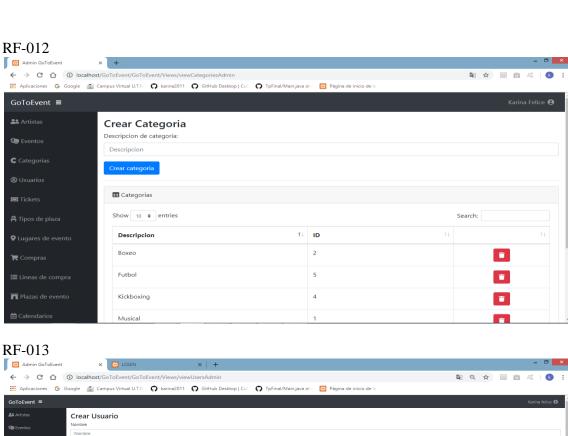


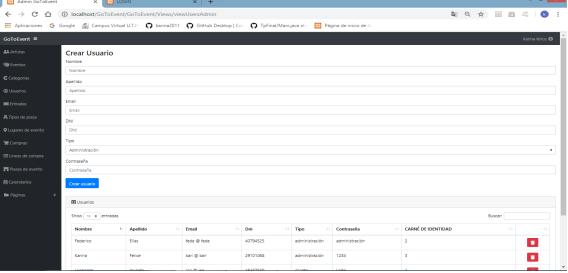
RF-010

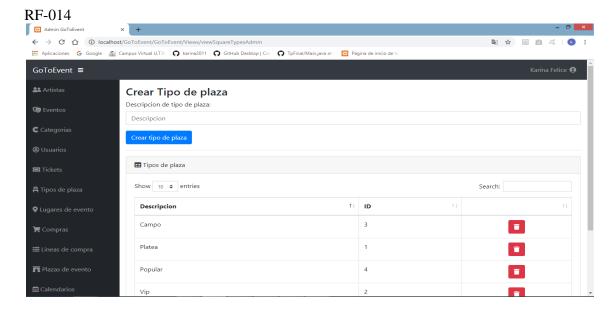


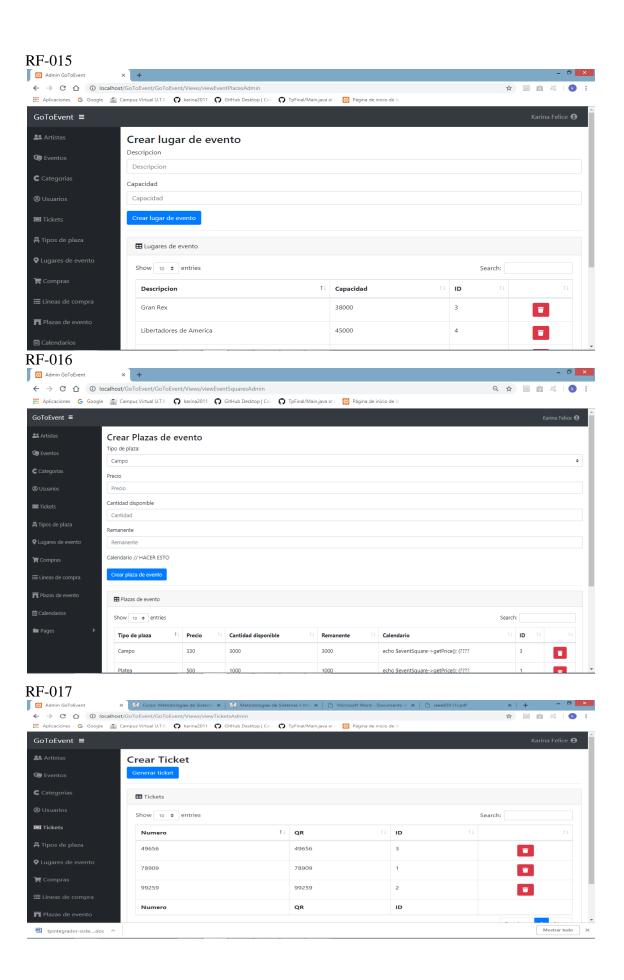
RF-011



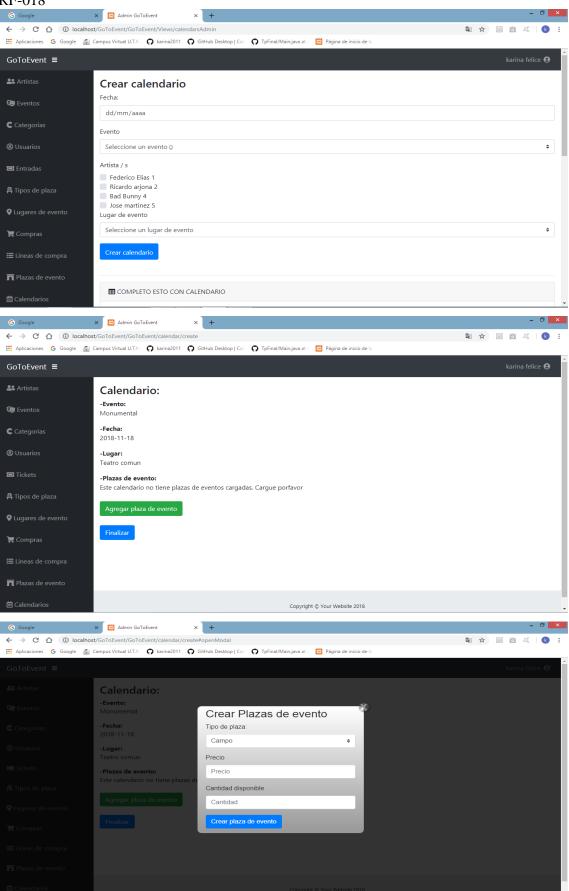








RF-018



3.2. Definición de Requisitos No Funcionales

La siguiente tabla muestra la lista de la definición de requisitos no funcionales de este sistema, las cuales restringen o condicionan el desarrollo e implantación del sistema.

Lista de Definición de Requisitos No Funcionales de GoToEvent

| Id | Descripción del requisito |
|-----------|---|
| Requisito | |
| RNF-001 | El sistema debe ser programados en capas respetando la arquitectura de tres capas logias. Esto implica el desarrollo de las clases que representan las entidades del modelo, controladoras de los casos de uso, las vistas y la capa de acceso a datos. |

4. Especificación de Requisitos del Sistema

Esta sección describe con mayor detalle cada uno de los requisitos funcionales identificados en la sección anterior, los cuales serán especificados usando un modelo de casos. Dicho modelo contendrá una serie de diagramas de casos de uso y un conjunto de plantillas llamadas escenarios, usadas para detallar cada caso de uso, donde se muestra la interacción del usuario-sistema.

En este sentido, los casos de uso representarán las funciones u operaciones que cada actor puede realizar dentro del sistema.

5. Diagrama de caso de uso

caso de uso diagram

Use Case diagram in package 'caso de uso'

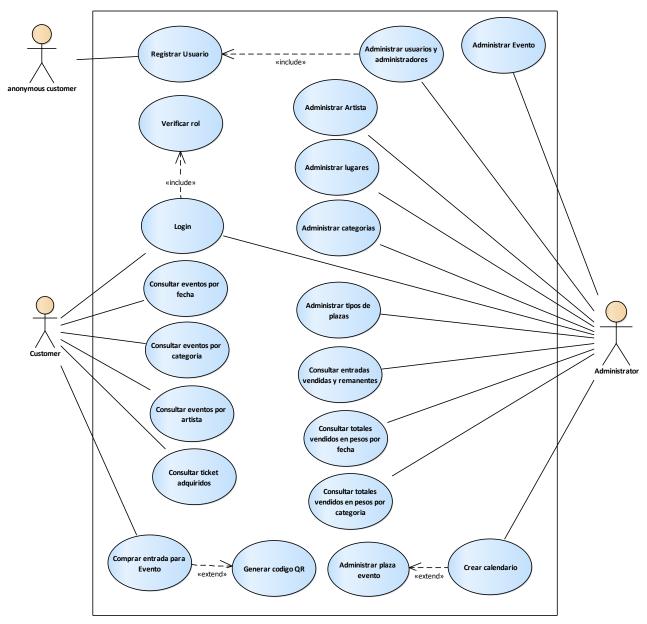


Figure 1: caso de uso

Administrar plaza evento



Administrar usuarios y administradores

RESPONSIBILITIES (INTERNAL REQUIREMENTS) ☑ Functional. RF-012

Generar codigo QR

RESPONSIBILITIES (INTERNAL REQUIREMENTS) ☑ Functional. RF-016

Administrar Artista

RESPONSIBILITIES (INTERNAL REQUIREMENTS) ☑ Functional. RF-009

Administrar Evento

RESPONSIBILITIES (INTERNAL REQUIREMENTS) ☑ Functional. RF-010

Administrar categorias

RESPONSIBILITIES (INTERNAL REQUIREMENTS) ☑ Functional. RF-011

Administrar lugares

RESPONSIBILITIES (INTERNAL REQUIREMENTS)



Consultar totales vendidos en pesos por categoria

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

✓ Functional. RF-018

Consultar totales vendidos en pesos por fecha

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

Functional. RF-018

Login

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

If the functional is a second contained as the second contai

Registrar Usuario

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

Functional. RF-001

Verificar rol

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

Functional. RF-002

5.1 Especificacion caso de uso Cargar Calendario a un Evento.

Crear calendario

RESPONSIBILITIES (INTERNAL REQUIREMENTS) Functional. RF-009 Functional. RF-010 Functional. RF-014 Functional. RF-017

SCENARIOS

- Basic Path. Basic Path
- 1. Seleccionar la fecha para el evento.

Alternate: 1a. El administrador selecciona una fecha anterior a la actual

- 2. Seleccionar el evento
 - Alternate: 2a. El administrador desea ingresar un evento que no se encuentra en la lista
- 3. Seleccionar el o los artistas que participan en el evento

Alternate: 3a. El administrador desea seleccionar un artista que no se encuentra en la lista

- 4. Seleccionar el lugar donde se producira el evento.
 - Alternate: 4a. El administrador desea seleccionar un lugar que no se encuentra en la lista
- 5. Presionar el boton "crear calendario".
 - Alternate: 5a. El administrador dejo un campo del calendario sin datos
 - Alternate: 5b. El administrador selecciono una fecha que no se encuentra disponible en lugar seleccionado
- 6. Presionar el boton "agregar plaza evento"

Alternate: 6a. El administrador desea incluir en el evento una plaza que no se encuentra en la lista

- 7. Seleccionar tipo de plaza.
- 8. Ingresar el precio que corresponde con la plaza.
- 9. Ingresar la cantidad de plazas disponibles para el tipo que se esta creando.
- 10. Presionar el boton "crear plaza evento".
- 11. Repetir los pasos 6,7,8,9 y 10 tantas veces como plazas tenga el evento.
- 12. Presionar el boton "finalizar" para crear el calendario.

Alternate: 12a. El total de todas las plazas supera el tope de capacidad del lugar

- Alternate. Seleccionar una fecha anterior a la actual
- 1. Al seleccionar una fecha anterior a la actual aparece un mensaje en la pantalla indicando el error cometido
- Alternate. Selecciona una fecha anterior a la actual
- 1. El administrador al seleccionar una fecha anterior a la actual
- 2. El sistema muestra un mensaje indicando el error del sistema y vuelve a la pantalla de crear calendario para que vuelva a iniciar la carga del calendario

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

- Alternate. El evento al cual se le desea generar el calendario no se encuentra en la lista
- Alternate. El administrador selecciona una fecha anterior a la actual
- 1. El administrador al seleccionar una fecha anterior a la actua
- 2. El sistema muestra un mensaje indicando el error del sistema y vuelve a la pantalla de crear calendario para que vuelva a iniciar la carga del calendario
- Alternate. El administrador desea ingresar un evento que no se encuentra en la lista
- 1. El administrador debe seleccionar en el menu la opcion de dar de alta un evento para crearlo
- 2. El sistema carga la pagina de crear un evento para que pueda ser generado
- 3. El administrador debe volver a la pagina de crear un calenadrio donde ahora si estara disponible el evento que desea.
- 🛱 Alternate. El administrador desea seleccionar un artista que no se encuentra en la lista
- 1. El administrador debe seleccionar del menu la opcion de dar de alta un artista.
- 2. El sistema carga la pagina que le permite dar de alta un artista
- 3. El administrador debe volver a seleccionar en el menu crear calendario donde ahora si aparecera el artista deseado.
- Alternate. El administrador desea seleccionar un lugar que no se encuentra en la lista
- 1. El administrador debe seleccionar en el menu dar de alta un lugar.
- 2. El sistema debe cargar la pagina que le ermite al administrador dar de alta el nuevo lugar.
- 3. El adminitrador debe seleccionar nuevamante dar de alta un calendario donde ahora si aparecera el lugar en la lista
- Alternate. El administrador dejo un campo del calendario sin datos
- 1. El administrador dejo un campo sin completar
- 2. El sistema muestra un mensaje indicando el error y vuelve a la pagina para poder generar el calendario con todos los datos.
- Alternate. El administrador desea incluir en el evento una plaza que no se encuentra en la lista
- 1. El administrador debe seleccionar en el menu la opcion de dar de alta una plaza evento para generarla.
- 2. El sistema debe cargar la pagina para que el adminitrador pueda generar la nueva plaza.
- 3. El administrador debe seleccionar nuevamente dar de alta calendario donde ahora si estara disponible la plaza deseada.
- Alternate. El administrador selecciono una fecha que no se encuentra disponible en lugar seleccionado
- 1. El administrador selecciono una fecha que o se encuentra disponible en el lugar seleccionado
- 2. El sistema debe mosrar un mensaje indicando el error y volver a cargar la pagina de crear calendario para volver a empezar

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

- 🛱 Alternate. El total de todas las plazas supera el tope de capacidad del lugar
- 1. El administrador cargo las diferentes plazas pero eltotal supera la capacidad del lugar del evento
- 2. El sistema debe mostrar un mensaje de error indicando que no se pudo crear el calendario ya que se ha superado la capacidad maxima del lugar y vuelve a cargar la pagina de dar de alta un calendario para volver a empezar.

5.2. Especificación Caso de Uso Comprar una entrada para un evento, calendario y tipo de asiento especifico.

Comprar entrada para Evento

RESPONSIBILITIES (INTERNAL REQUIREMENTS)

☑ Functional. RF-008

SCENARIOS

- Basic Path. Basic Path
- 1. El cliente inicia sesion

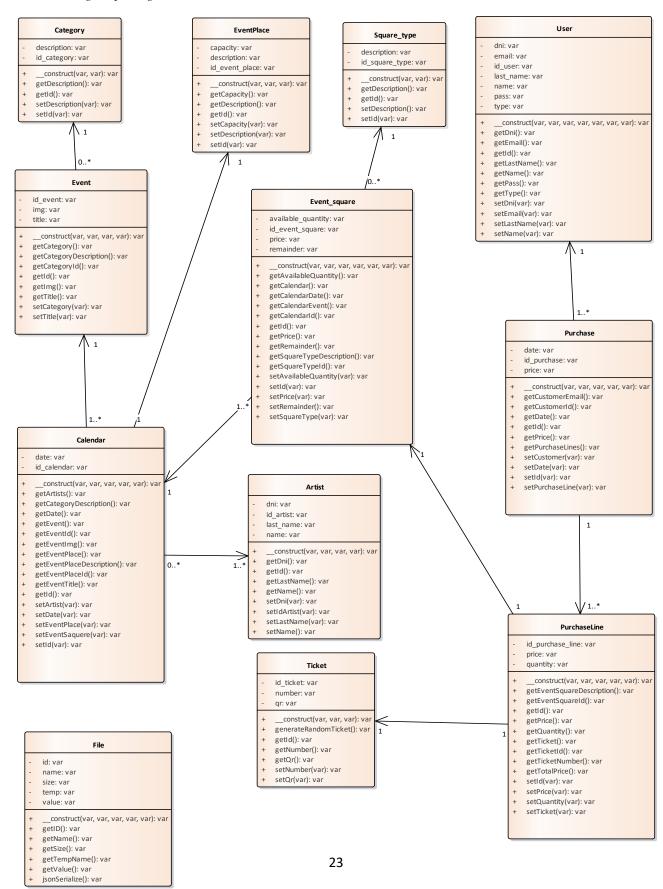
Alternate: 1a. El cliente no puede iniciar sesion

- 2. El cliente selecciona el evento que le interesa
- 3. El cliente selecciona el lugar que desea y la fecha
- 4. El cliente selecciona la plaza que desea comprar
- 5. El cliente ingresa la cantidad de entradas que desea comprar6. El cliente presiona el boton comprar para que sean cargadas en su carrito.
- Alternate: 6a. No se encuentran disponibles la cantidad de entradas que el cliente desea comprar
- 7. El cliente repite los pasos 5,6 y 7 tantas veces como cantidad de plazas distintas desea.
- 8. El cliente repite desde el paso 4 para comprar entradas en otra fecha
- 9. El cliente repite desde el paso 3 para comprar entradas en otro lugar.
- 10. El cliente repite desde el paso 2 para comprar entradas para otro evento.
- 11. El cliente selecciona la opcion finalizar compra para hacer efectiva todas las compras cargadas en su carrito
- Alternate. El cliente no puede iniciar sesion
- 1. El cliente no recuerdad sus datos para iniciar sesion
- 2. El sistema debe tener una opcion para recuperar los datos, de manera que con el mail del usuario y previa validacion le envie la informacion al cliente.
- Alternate. No se encuentran disponibles la cantidad de entradas que el cliente desea comprar
- 1. El cliente ingresa una cantidad de entradas superior a las disponibles en el remanente del evento.
- 2. El sistema debe mostrar un mensaje indicando el motivo por el cual no es posible comprar las entradas y volver a la pagina del evento seleccionado para que pueda elegir otra fecha u otro lugar.

6. Diagrama de Clases con la arquitectura implementada.

models

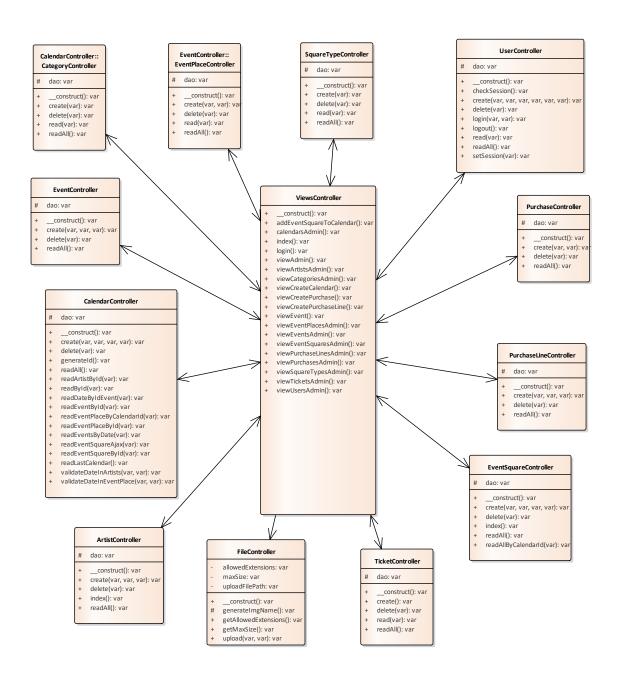
Package in package 'GoToEvent'



controllers

Package in package 'GoToEvent'

controllers Version 1.0 Phase 1.0 Proposed Karina created on 16/11/2018. Last modified 16/11/2018



interfaces

Package in package 'GoToEvent'

«interface»

Crud

- + create(var): var
- + delete(var): var
- read(var): var
- + readAll(): var
- + update(var, var): var

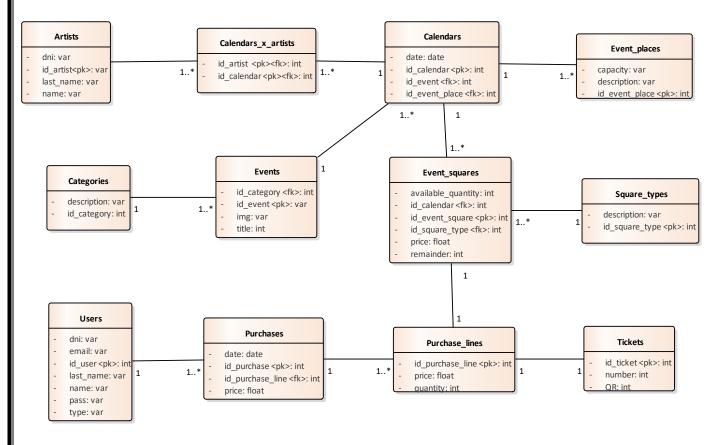
Crud

INCOMING STRUCTURAL RELATIONSHIPS

- → Realization from EventSquareDao to Crud
- Realization from PurchaseDao to Crud
- → Realization from EventDao to Crud
- → Realization from CategoryDao to Crud
- Realization from CalendarDao to Crud
- Realization from ArtistDao to Crud
- → Realization from UserDao to Crud
- → Realization from CalendarArtistDao to Crud
- → Realization from PurchaseLineDao to Crud
- → Realization from EventDao to Crud
- → Realization from TicketDao to Crud
- → Realization from ArtistDao to Crud
- → Realization from EventPlaceDao to Crud
- → Realization from SquareTypeDao to Crud

7. Diagrama de entidades y relaciones

DER



Artists

| ATTRIBUTES | |
|---|--|
| dni : var Private | |
| v diii . vai Tiivate | |
| ✓ id_artist <pk>: var Private ✓ id_artist<pk>: var Private</pk></pk> | |
| ✓ last_name : var Private | |
| name : var Private | |

Calendars

| ATTRIBUTES |
|---------------------------------------|
| ♦ date : date Private |
| id_calendar <pk>: int Private</pk> |
| id_event <fk>: int Private</fk> |
| id_event_place <fk>: int Private</fk> |

Calendars_x_artists

ATTRIBUTES

- id_artist <pk><fk>: int Private
- id_calendar <pk><fk>: int Private

Categories

ATTRIBUTES

- description : var Private
- id_category <pk> : int Private

Event_places

ATTRIBUTES

- capacity: var Private
- description : var Private
- id_event_place <pk>: int Private

Event_squares

ATTRIBUTES

- available_quantity : int Private
- id_calendar <fk>: int Private
- id_event_square <pk> : int Private
- id_square_type <fk>: int Private
- price : float Private
- remainder: int Private

Events

ATTRIBUTES

- id_category <fk> : int Private
- id_event <pk> : var Private
- img: var Private
- vitle: int Private

Purchase_lines

ATTRIBUTES

- id_purchase_line <pk>: int Private
- price : float Private
- quantity: int Private

Purchases

ATTRIBUTES

- date : date Private
- id_purchase <pk>: int Private
- id_purchase_line <fk>: int Private
- price : float Private

Square_types

ATTRIBUTES

- description : var Private
- id_square_type <pk>: int Private

Tickets

ATTRIBUTES

- id_ticket <pk>: int Private
- number: int Private
- QR : int Private

Users

ATTRIBUTES

- dni: var Private
- email: var Private
- id_user <pk> : int Private
- last_name : var Private
- name : var Private
- pass : var Private
- var Private