|  |  |  |
| --- | --- | --- |
| GAUDI SOLUTIONS | INTEGRANTES | CODIGOS |
| ANÁLISIS Y DISEÑO DE SOFTWARE  24 de Octubre de 2013 | Néstor Cruz Hernández | 201310690 |
| Felipe Rojas Echeverri | 201315979 |
| Juan Pedro Mendoza | 200310723 |
| Julián Aguirre Domínguez | 201221709 |
| María Paula Forero | 201310697 |

****

Taller 3

Aplicación gestión de reuniones empresa

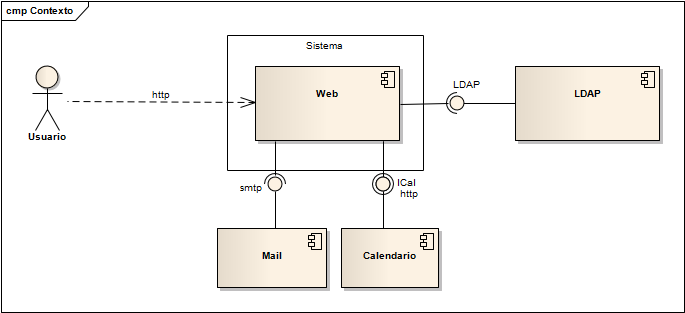


1. **Análisis básico**

Para hacer el diseño de la aplicación de calendario, se comienza por un análisis básico, en donde se definen los siguientes elementos:

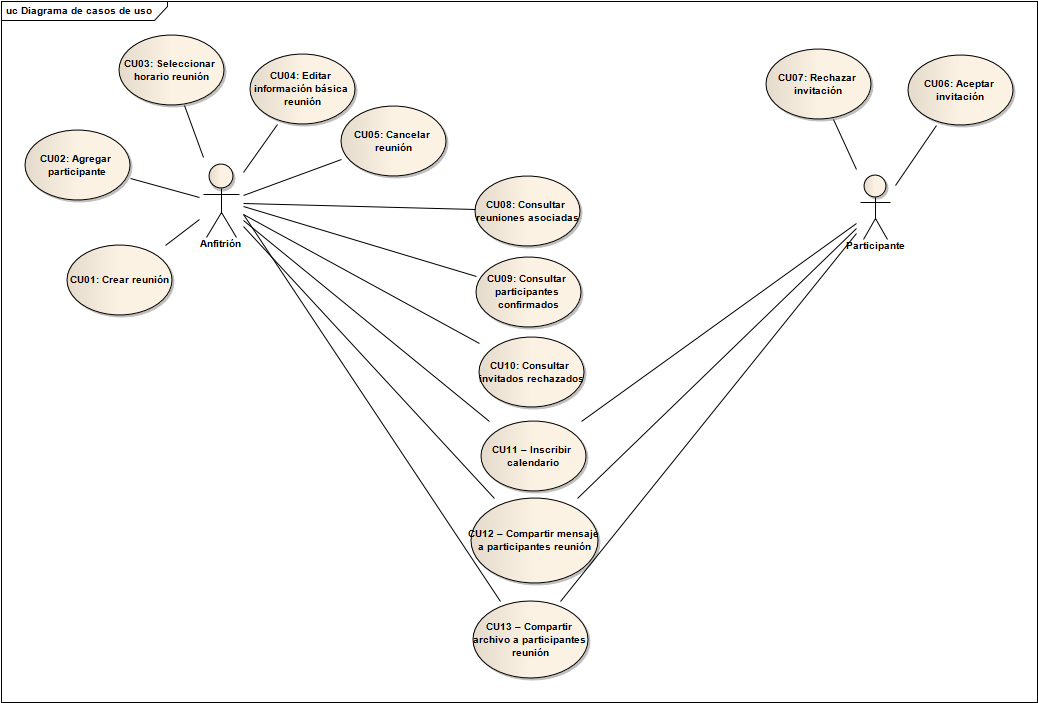
* 1. **Contexto**

A continuación se muestra el diagrama de contexto, en el se puede ver como la aplicación web debe interactuar con diferentes componentes, tales como el mail de cada invitado a las reuniones, los calendarios externos de los usuarios del sistema, y un componente LDAP utilizado para la autenticación de los usuarios



* 1. **Casos de uso**

A continuación se muestran los casos de uso para la aplicación de calendario



|  |  |
| --- | --- |
| **Actor** | **Descripción** |
| Anfitrión | Representa a la persona que administra la reunión |
| Participante | Representa a la persona que asiste a la reunión |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Actor** | **Caso de uso** | **Descripción** | **Entradas** | **Precondiciones** | **Resultados** |
| Anfitrión | CU01 – Crear reunión | Donde se define un rango de tiempo para organizar la reunión, una duración, una ubicación, un nombre y una descripción |  |  |  |
| Anfitrión | CU02 - Agregar participante | Donde el anfitrión indica quienes serán los invitados a la reunión. |  |  |  |
| Anfitrión | CU03 - Seleccionar horario reunión |  |  |  |  |
| Anfitrión | CU04 – Editar información básica reunión. | Permite editar la información básica sobre la reunión (nombre, descripción, etc.) |  |  |  |
| Anfitrión | CU05 – Cancelar reunión. | Permite que el anfitrión cancele una reunión si lo considera conveniente: para esto se envía un correo electrónico a todos los participantes y se cambia el estado de la reunión a cancelada en los calendarios de los participantes. |  |  |  |
| Anfitrión, Participante | CU08 - Consultar reuniones asociadas. Ver información de todas las reuniones en las que esté involucrado. |  |  |  |  |
| Anfitrión, Participante | CU09 – Consultar participantes confirmados. Permite ver quienes han confirmado la invitación |  |  |  |  |
| Anfitrión, Participante | CU10 – Consultar invitados rechazados. Permite ver quienes han declinado la invitación |  |  |  |  |
| Anfitrión, Participante | CU11 – Inscribir calendario |  |  |  |  |
| Anfitrión, Participante | CU12 – Compartir mensaje a participantes reunión |  |  |  |  |
| Anfitrión, Participante | CU13 – Compartir archivo a participantes reunión |  |  |  |  |
| Participante | CU06 - Aceptar invitación |  |  |  |  |
| Participante | CU07 - Rechazar invitación |  |  |  |  |

* 1. **Modelo del dominio**

Hacen falta las entidades en este punto

* 1. **Restricciones**

|  |  |
| --- | --- |
| **Restricción** | **Descripción** |
| **R1** | Aplicación web, con posibilidad de construir aplicaciones móviles nativas en el futuro. |
| **R2** | La autenticación debe hacerse usando el sistema LDAP con el que ya se cuenta. |
| **R3** | Los calendarios utilizados para conocer las actividades de los participantes son externos al sistema y deben estar disponibles en formato iCalendar – ics (ej. Google, iCal, etc.). |
| **R4** | El sistema nunca mostrará un horario en el que no puedan asistir los invitados obligatorios |
| **R5** | Las invitaciones a las reuniones se deben aceptar o rechazar a través del enlace respectivo enviado por correo electrónico, dar clic en el enlace debe ser suficiente para aceptar o rechazar la invitación (no se requiere autenticación) |
| **R6** | Cada persona debe inscribir sus calendarios usando una URL, las credenciales de autenticación y la frecuencia de actualización |
| **R7** | El sistema solo debe almacenar los horarios de las actividades. Cualquier otra información no está permitida |
| **R8** | El sistema debe exponer una URL para exportar los calendarios de la persona en formato iCalendar |

* 1. **Atributos de calidad**

|  |  |  |
| --- | --- | --- |
| **Id** | **Atributo de calidad** | **Descripción** |
| **AC01** | Interoperabilidad | Debido a que el sistema debe interactuar con diferentes tipos de componentes para la sincronización de los calendarios |
| **AC02** | Seguridad | El usuario debe ingresar utilizando un nombre de usuario y una contraseña, y todos los servicios expuestos desde la aplicación deben requerir autenticación a excepción de los servicios para aceptar o rechazar una invitación a una reunión |
| **AC03** | Flexibilidad | El sistema debe ser capaz de soportar dispositivos móviles en el futuro |

1. **Diseño calendario**

El diseño se realizará por medio de diferentes niveles de abstracción, en los cuales, en cada uno se realiza una descomposición de los componentes o módulos de los que se compone cada nivel.

* 1. **Nivel 0 calendario**

En este nivel encontramos los principales componentes del diseño, estos son:

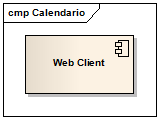
* Web Client
* Server Calendar

Los componentes Calendar Providers, LDAP e EMAIL, son compontes externos al sistema, pero estos son representados para mostrar las dependencia en interfaces de comunicación hacia ellos.



* + 1. **Componente Web Client**

El cliente web es un componente que corre desde el lado del cliente en el navegador, este cliente web estará construido utilizando tecnología Jquery y HTML 5, de forma tal que pueda ser modificado para ser utilizados en aplicaciones nativas de dispositivos móviles, bajo la tecnología HTML 5.



* + - 1. **Responsabilildades funcionales**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Caso de uso | Nombre | Entrada | Salida |
| **WC01** | CU02 - Agregar participante | Consultar usuarios | N/A | Listado de usuarios para invitar a la reunión |
| **WC02** | CU02 - Agregar participante | Seleccionar invitados | * Usuario * Obligatorio | Usuarios indicados en la reunión |
| **WC03** | CU03 - Seleccionar horario reunión | Consultar horarios | * Rango Inicio * Rango Fin * Duración * Asistentes * Cantidad resultados | Horarios en los que se puede crear la reunión indicando cuales de los invitados no pueden asistir |
| **WC04** | CU01 - Crear reunión | Crear reunión | * Nombre * Descripción * Ubicación * Horario * Participantes * Anfitrión | Se crea la reunión en el sistema |
| **WC05** | CU04 - Editar información básica reunión  CU05 - Cancelar reunión | Consultar reunión por anfitrión | * Anfitrión | Listado de reuniones creadas por el anfitrión |
| **WC06** | CU04 - Editar información básica reunión | Modificar reunión | * Nombre * Descripción * Ubicación * Horario * Participantes * Reunión | Se realiza la modificación de la reunión con los datos ingresados |
| **WC07** | CU05 - Cancelar reunión | Cancelar reunión | * Reunión | Se cancela reunión |
| **WC08** | CU08 - Consultar reuniones asociadas | Consultar reuniones | * Participante (anfitrión, invitados) | Se consultan todas las reuniones en la que se encuentra el participante |
| **WC09** | CU08 - Consultar reuniones asociadas | Consultar mensajes | * Reunión | Se consultan todos los mensajes de una reunión |
| **WC10** | CU08 - Consultar reuniones asociadas | Consultar archivos | * Reunión | Se consultan todos los archivos de una reunión |
| **WC11** | CU09 - Consultar participantes confirmados | Consultar participantes confirmados | * Reunión | Se consultan los participantes confirmados de una reunión |
| **WC12** | CU10 - Consultar invitados rechazados | Consultar participantes rechazados | * Reunión | Se consultan los participantes que rechazaron la reunión |
| **WC13** | CU11 - Inscribir calendario  R6 – Inscripción de calendarios externos | Inscribir calendario | * Url de sincronización * Usuario * Contraseña * Intervalo de sincronización | Se registra el calendario |
| **WC14** | CU12 - Compartir mensaje a participantes reunión | Compartir mensajes | * Reunión * Mensajes | Se registra un mensaje en la reunión |
| **WC15** | CU13 – Compartir archivo a participantes reunión | Compartir archivo | * Reunión * Archivo | Se registra un archivo para ser compartido a todos los integrantes de la reunión |
| **WC16** | CU13 – Compartir archivo a participantes reunión | Descargar archivo | * Reunión * Nombre archivo | Se descarga el archivo seleccionado |
| **WC17** | R2 - La autenticación debe hacerse usando el sistema LDAP con el que ya se cuenta | Login | * Usuario * Contraseña | El usuario ingresa al sistema |

* + - 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **WC18** | R1 – Cliente web | El cliente web se desarrollará utilizando la tecnología HTML 5 y Jquery, de forma tal que se facilite la modificación del sistema para aplicaciones nativas móviles |
| **WC19** | R5 – No será necesaria la autenticación del usuario para aceptar o rechazar una invitación | Para estos servicios no se pedirá usuario y contraseña. |
| **WC20** | R6 – Inscripción de calendarios externos | El cliente web debe proveer una interfaz que permita registrar los calendarios externos |

* + - 1. **Responsabilidades no funcionales**

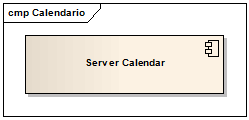
|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **WC21** | AC02 - Seguridad | El sistema solicita usuario y contraseña para acceder a la aplicación, para la invocación de los servicios excepto los servicios de aceptar o rechazar invitación, se requiere que el usuario se autentique. |
| **WC22** | AC03 – Flexibilidad | Debido a uso de HTML 5 se facilita la adaptación de la aplicación web a un aplicación móvil nativa con el uso de estas tecnologías. |

* + - 1. **Ciclo de vida**

El ciclo de vida de este componente está dado por el la sesión web, o sea el tiempo en el que se encuentre el usuario en la sesión del navegador web.

* + 1. **Componente Calendar Server**

Esta es una aplicación web que se encuentra en lado servidor, sus principales interfaces se encuentran expuestas en servicios REST, de forma tal que puedan ser fácilmente consumidas por el cliente web, u otros clientes de calendario mediante el protocolo http.

****

* + - 1. **Responsabilidades funcionales**

Todas las url de los servicios REST, se encontrarán bajo el contexto http://¨﷽﷽﷽﷽﷽﷽﷽﷽an ser facilmente consest, se econtraruentran expuestas en servicios rest, de forma tal que puedan ser facilmente cons{pagina}/{contexto}/Rest/Calendario/{servicio}

Para los servicios que requieren autenticación es necesario que se envíe la cookie de sesión que es respondida por el servicio de autenticación, cuando se ingresa al sistema.

**SC01**

* **Caso de uso:** CU02 - Agregar participante (consulta de usuarios)
* **URL Servicio:** /Usuarios
* **Método:** GET
* **Entradas:** N/A
* **Salida:** Tipo JSON

|  |
| --- |
| {  "usuarios": [  {  "nombre": String,  "login": String  },  {  "nombre": String,  "login": String  }  ]  } |

**SC02**

* **Caso de uso:** CU03 - Seleccionar horario reunión (Consultar horarios)
* **URL Servicio:** /Horarios
* **Método:** GET
* **Entradas:** Tipo JSON

|  |
| --- |
| {  "horarios": {  "inicio": "dd-MM-yyyy",  "fin": "dd-MM-yyyy",  "duracion": int,  "resultados": int,  "invitados": [  {  "login": String,  "oblitagorio": boolean  },  {  "login": String,  "oblitagorio": boolean  }  ]  }  } |

* **Salida:** Tipo JSON

|  |
| --- |
| {  "horarios": [  {  "fecha": "dd-MM-yyyy hh-mm-ss",  "asistentes": [  {  "obligatorio": boolean,  "asiste": boolean  }  ],  "cantidadAsistentes": int  },  {  "fecha": " dd-MM-yyyy hh-mm-ss",  "asistentes": [  {  "obligatorio": boolean,  "asiste": boolean  }  ],  "cantidadAsistentes": int  }  ]  } |

**SC03**

* **Caso de uso:** CU01 - Crear reunión (crear reunión)
* **URL Servicio:** /Reunion
* **Método:** PUT
* **Entradas:** Tipo JSON

|  |
| --- |
| {  "reunion": {  "nombre": String,  "descripcion": String,  "horario": "dd-MM-yyyy hh-mm-ss",  "ubicacion": String,  "anfitrion": String,  "participantes": [  {  "login": String,  obligatorio: boolean  },  {  "login": String,  obligatorio: boolean  }  ]  }  } |

* **Salida:** Tipo JSON

|  |
| --- |
| {  "reunion": "creada",  "mensaje": String  } |

**SC04**

* **Caso de uso:** CU04 - Editar información básica reunión - CU05 - Cancelar reunión (consultar reunión por anfitrión)
* **URL Servicio:** /Reunion/{Anfitrion}
* **Método:** GET
* **Entradas:** parámetros
  + **Anfitrion** – login del anfitrion
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reuniones": [  "reunion": {  "nombre": String,  "descripcion": String,  "horario": "dd-MM-yyyy hh-mm-ss",  "ubicacion": String,  "anfitrion": String,  "id": String,  "participantes": [  {  "login": String,  obligatorio: boolean  },  {  "login": String,  obligatorio: boolean  }  ]  }  ]  } |  |

**SC05**

* **Caso de uso:** CU04 - Editar información básica reunión (modificar reunión)
* **URL Servicio:** /Reunion
* **Método:** POST
* **Entradas:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": {  "nombre": String,  "descripcion": String,  "horario": "dd-MM-yyyy hh-mm-ss",  "ubicacion": String,  "anfitrion": String,  "id": String,  "participantes": [  {  "login": String,  obligatorio: boolean  },  {  "login": String,  obligatorio: boolean  }  ]  }  } |  |

* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": "modificada",  "mensaje": String  } |  |

**SC06**

* **Caso de uso:** CU05 - Cancelar reunión (cancelar reunion)
* **URL Servicio:** /Reunión
* **Método:** DELETE
* **Entradas:** N/A
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": "Cancelada",  "mensaje": String  } |  |

**SC07**

* **Caso de uso:** CU08 - Consultar reuniones asociadas (consultarReuniones)
* **URL Servicio:** /Reunion/{Login}
* **Método:** GET
* **Entradas:** parámetros
  + **Login** – login del usuario del cual se consultan las reuniones
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reuniones": [  "reunion": {  "nombre": String,  "descripcion": String,  "horario": "dd-MM-yyyy hh-mm-ss",  "ubicacion": String,  "anfitrion": String,  "id": String,  "participantes": [  {  "login": String,  obligatorio: boolean  },  {  "login": String,  obligatorio: boolean  }  ]  }  ]  } |  |

**SC08**

* **Caso de uso:** CU08 - Consultar reuniones asociadas (consultar mensajes)
* **URL Servicio:** /Reunion/{Reunion}/Mensajes
* **Método:** GET
* **Entradas:** parámetros
  + **Reunion** – Id de la reunión sobre la cual se desean consultar los mensajes
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": {  "id": String,  "mensajes": [  {  "login": String,  "mensaje": String,  "fecha": "dd-MM-yyyy hh-mm-ss"  },  {  "login": String,  "mensaje": String,  "fecha": "dd-MM-yyyy hh-mm-ss"  }  ]  }  } |  |

**SC09**

* **Caso de uso:** CU08 - Consultar reuniones asociadas (consultar archivos reunión)
* **URL Servicio:** /Reunion/{Reunion}/Archivos
* **Método:** GET
* **Entradas:** parámetros
  + **Reunion** – Id de la reunión sobre la cual se desean consultar los archivos
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": {  "id": String,  "archivos": [  {  "login": String,  "archivo": String,  "fecha": "dd-MM-yyyy hh-mm-ss"  },  {  "login": String,  "archivo": String,  "fecha": "dd-MM-yyyy hh-mm-ss"  }  ]  }  } |  |

**SC10**

* **Caso de uso:** CU09 - Consultar participantes confirmados (consultar participantes confirmados)
* **URL Servicio:** /Reunion/{Reunion}/Confirmados
* **Método:** GET
* **Entradas:** parámetros
  + **Reunion** – Id de la reunión sobre la cual se desean consultar los participantes confirmados
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": {  "id": String,  "confirmados": [  {  "nombre": String,  "login": String  },  {  "nombre": String,  "login": String  }  ]  }  } |  |

**SC11**

* **Caso de uso:** CU10 - Consultar invitados rechazados (consultar participantes rechazados)
* **URL Servicio:** /Reunion/{Reunion}/Rechazados
* **Método:** GET
* **Entradas:** parámetros
  + **Reunion** – Id de la reunión sobre la cual se desean consultar invitados que rechazaron la invitación
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": {  "id": String,  "rechazados": [  {  "nombre": String,  "login": String  },  {  "nombre": String,  "login": String  }  ]  }  } |  |

**SC12**

* **Caso de uso:** CU11 - Inscribir calendario - R6 – Inscripción de calendarios externos
* **URL Servicio:** /Calendario
* **Método:** PUT
* **Entradas:** Tipo JSON

|  |  |
| --- | --- |
| {  "calendario": {  "url": String,  "usuario"String,  "contrasena": String,  "intervaloSinc": int  }  } |  |

* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "calendario": "Agregado",  "mensaje": String  } |  |

**SC13**

* **Caso de uso:** CU12 - Compartir mensaje a participantes reunión (compartir mensaje)
* **URL Servicio:** /Reunion/Mensaje
* **Método:** PUT
* **Entradas:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": String,  "mensaje": String,  "usuario": String,  "fecha": "dd-MM-yyyy hh-mm-ss"  } |  |

* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "mensaje": "Agregado",  } |  |

**SC14**

* **Caso de uso:** CU13 – Compartir archivo a participantes reunión (compartir archivo)
* **URL Servicio:** /Reunion/Archivo
* **Método:** PUT
* **Entradas:** Tipo JSON

|  |  |
| --- | --- |
| {  "reunion": String,  "archivo": String(base64),  "usuario": String,  "fecha": "dd-MM-yyyy hh-mm-ss"  } |  |

* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "archivo": "Agregado",  "mensaje": String,  } |  |

**SC15**

* **Caso de uso:** CU13 – Compartir archivo a participantes reunión (obtener archivo)
* **URL Servicio:** /Reunion/{Reunion}/Archivo/{Archivo}
* **Método:** GET
* **Entradas:** parámetros
  + **Reunion –** id de la reunión a la cual se desea descargar el archivo
  + **Archivo –** Nombre del archivo que se desea descargar
* **Salida:** octet-stream - bytes del archivo a descargar

**SC16**

* **Caso de uso:** CU07 - Rechazar invitación (rechazar invitación)
* **URL Servicio:** /Reunion/{Reunion}/Acepta/false
* **Seguridad:** No requiere autenticación
* **Método:** Post
* **Entradas:** parámetros
  + **Reunión -** id de la reunión
  + **Login –** Login del usuario que rechaza la invitación
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "invitacion": "rechazada",  "mensaje": String,  } |  |

**SC17**

* **Caso de uso:** CU06 - Aceptar invitación
* **URL Servicio:** /Reunion/{Reunion}/Acepta/true
* **Seguridad:** No requiere autenticación
* **Método:** Post
* **Entradas:** parámetros
  + **Reunión -** id de la reunión
  + **Login -** Login del usuario que acepta la invitación
* **Salida:** Tipo JSON

|  |  |
| --- | --- |
| {  "invitacion": "aceptada",  "mensaje": String,  } |  |

**SC18**

* **Caso de uso:** R2 - La autenticación debe hacerse usando el sistema LDAP con el que ya se cuenta
* **URL Servicio:** /Autenticacion
* **Método:** POST
* **Entradas:** parámetros
  + **User –** login del usuario
  + **Password –** password del usuario
* **Salida:** Tipo JSON – este servicio debe retornar la cookie de sesión necesaria para ingresar al sitio

|  |  |
| --- | --- |
| {  "autenticado": boolean,  "mensaje": String,  } |  |

**SC19**

* **Caso de uso:** R8 – Exponer calendarios usuario
* **URL Servicio:** /{Usuario}/Events
* **Método:** GET
* **Seguridad:** Basic authentication
* **Entradas:** parámetros
  + **Usuario –** nombre de usuario para entrar a la aplicación
* **Salida:** octec-stream, Events.ics
  + - 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **SC20** | R1 – Aplicación web, con posibilidad de construir aplicaciones móviles nativas en el futuro. | Todos los servicios expuestos por este componente son servicios REST, de forma tal que pueden ser consumidos fácilmente por cualquier dispositivo. |
| **SC21** | R3 – Los calendarios utilizados para conocer las actividades de los participantes son externos al sistema y deben estar disponibles en formato iCalendar | Se expone un servicio que permite registrar los calendarios externos para ser procesados por este mismo componente en periodos específicos de tiempo |
| **SC22** | R4 – El sistema nunca mostrará un horario en el que no puedan asistir los invitados obligatorios | Este componente se encarga de exponer el servicio para consultar los horarios y se debe asegurar de cumplir con esta restricción |
| **SC23** | R5 - Las invitaciones a las reuniones se deben aceptar o rechazar a través del enlace respectivo enviado por correo electrónico, dar clic en el enlace debe ser suficiente para aceptar o rechazar la invitación | En este componente se exponen los servicios que permiten aceptar o rechazar una invitación a una reunión |
| **SC24** | R6 - Cada persona debe inscribir sus calendarios usando una URL, las credenciales de autenticación y la frecuencia de actualización | Este componente se encarga de exponer un servicio para el registro de url de calendarios externos |
| **SC25** | R7 - El sistema solo debe almacenar los horarios de las actividades. Cualquier otra información no está permitida | Este componente almacena únicamente la información permitida de los calendarios externos |
| **SC26** | R8 - El sistema debe exponer una URL para exportar los calendarios de la persona en formato iCalendar | Este componente expone una url para obtener los calendarios de un usuario |
| **SC27** | Persistencia | Este componente se encentra encargado de la conexión con la persistencia de la aplicación |
| **SC28** | Envió de correos | Este componente se encarga del envió correos electrónicos a los participantes de las reuniones |

* + - 1. **Responsabilidades no funcionales**

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **SC28** | AC01 - Interoperabilidad | Este componente se encarga de comunicarse con los calendarios de los proveedores externos |
| **SC29** | AC02 – Seguridad | La autenticación se hace por medio de este componente, y la comunicación se realiza mediante el protocolo https |
| **SC30** | AC03 – Flexibilidad | Los servicios REST permiten la invocación de servicios desde diferentes tipos de dispositivos |

* + - 1. **Ciclo de vida**
* Este componente se inicia y queda en espera para atender las conexiones de los clientes web, o los calendarios de los clientes.
* Cuando este componente se inicia, se replica la información de los usuarios en la base de datos local del sistema.
  + 1. **Conexión Web Client a Server Calendar**

A continuación se hace referencia a la conexión existente entre el cliente web y el servidor

****

* Esta conexión está definida bajo el estándar REST
* El protocolo de conexión es https, utilizando JSON como el principal formato para el intercambio de información.
  + - 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **WCSC01** | R1 – Aplicación web, con posibilidad de construir aplicaciones móviles nativas en el futuro. | Al estar esta conexión basada en servicios REST, permite que sea fácilmente invocado desde otro dispositivos |
| **WCSC02** | R5 - Las invitaciones a las reuniones se deben aceptar o rechazar a través del enlace | Este conexión permite la invocación para aceptar o rechazar una reunión |
| **WSCS03** | R8 - El sistema debe exponer una URL para exportar los calendarios de la persona en formato iCalendar | Los calendarios del sistema son expuestos mediante esta conexión |

* + - 1. **Responsabilidades no funcionales**

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **WCSC04** | AC02 - Seguridad | Por medio del protocolo https esta conexión se hace de manera segura |
| **WCSC05** | AC03 – Flexibilidad | El uso de servicios REST permite cambiar el componente de presentación fácilmente. |

* + - 1. **Ciclo de vida**

El ciclo de vida de esta conexión está definido por la duración de la sesión web y será contralada por el navegador.

* + 1. **Conexión Server Calendar a LDAP**

Esta conexión especifica la forma en la que interactúan el servidor de calendario y el componente LDAP de la empresa.



* Esta conexión está definida bajo el protocolo LDAP para el acceso a directorios
* El acceso a esta conexión se especifica desde las configuraciones de Server Calendar
* La ruta de conexión para obtener la contraseña del usuarios se debe especificar en las configuraciones de Server Calendar
* Esta conexión será utilizada para replicar los usuarios en la base de datos de la aplicación.
  + - 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **SCLD01** | R2 – La autenticación debe hacerse usando el sistema LDAP con el que ya se cuenta. | Esta conexión asegura la comunicación con el LDAP para la autenticación |

* + - 1. **Ciclo de vida**

Cada vez que se autentique en el sistema, se abrirá una conexión con el componente LDAP y se cerrará cuando termine la autenticación del usuario

* + 1. **Conexión Server Calendar a EMAIL**

Esta conexión especifica la forma en la que interactúan el servidor de calendario y el envío de correos



* Esta conexión se realiza mediante el protocolo SMTP
  + - 1. **Ciclo de vida**

Se crea una nueva conexión para cada correo enviado

* + 1. **Conexión Server Calendar a Calendar Providers**

Esta conexión indica cómo se realiza la sincronización de los calendarios con los proveedores externos.



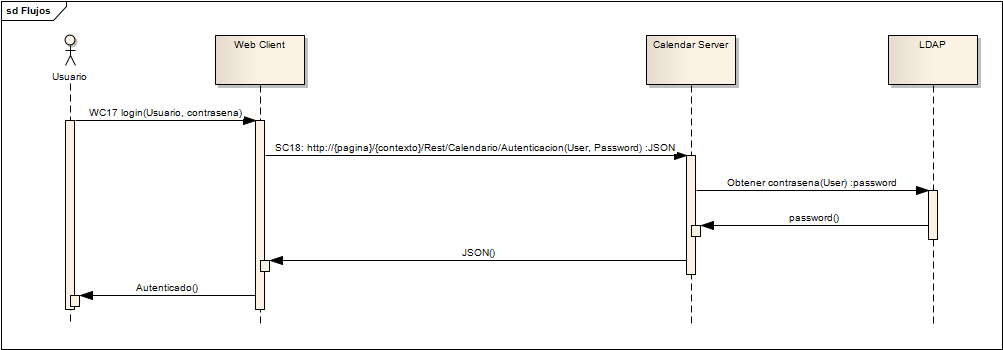
* Esta conexión se realiza mediante http utilizando el formato ICalendar
* Esta conexión es bidireccional, los proveedores externos pueden revisar los calendarios del sistema y el sistema puede leer calendarios externos
  + - 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **SCCL01** | R3 – Los calendarios utilizados para conocer las actividades de los participantes son externos al sistema y deben estar disponibles en formato ICalendar – ics (ej. Google, iCal, etc.). | Esta conexión se utiliza para sincronizar los calendarios externos |
| **SCCL02** | R8 - El sistema debe exponer una URL para exportar los calendarios de la persona en formato iCalendar | Esta conexión expone los calendarios por medio del formato ICalendar |

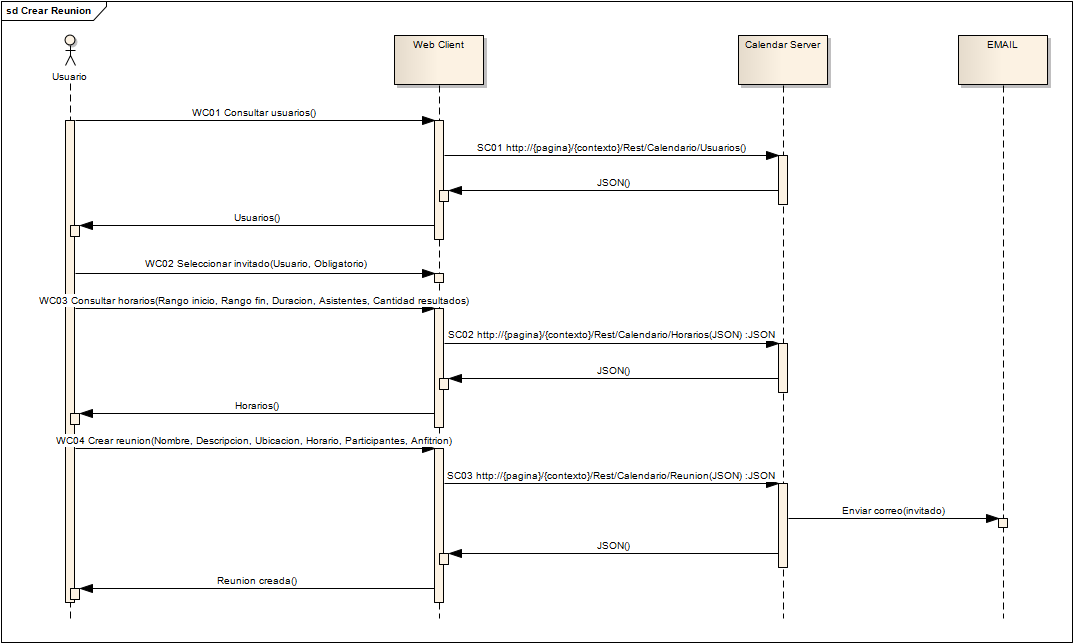
* + - 1. **Ciclo de vida**
* Se crea una conexión independiente para acceder a cada uno de los proveedores de calendarios externos
* Se crea una conexión para cada calendario externo que acceda al sistema para la descarga de los calendarios.
  + 1. **Flujos de ejecución**

A continuación se indican los flujos de ejecución de los principales casos de uso y las restricciones.

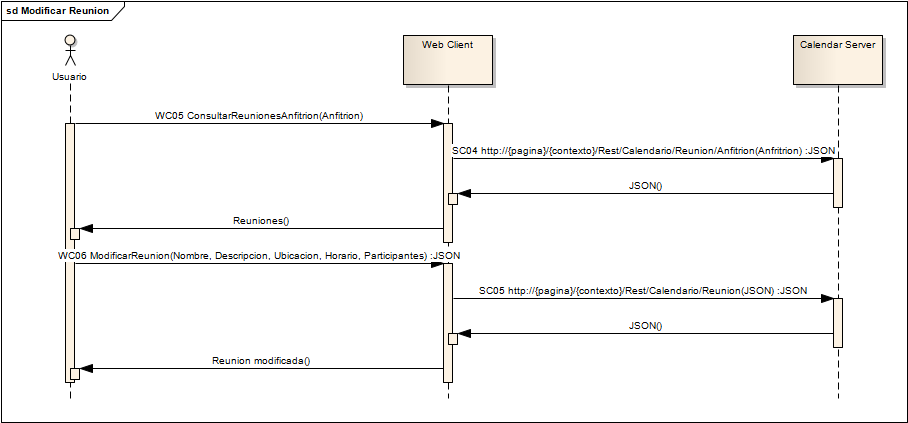
* + - 1. **Autenticación**

****

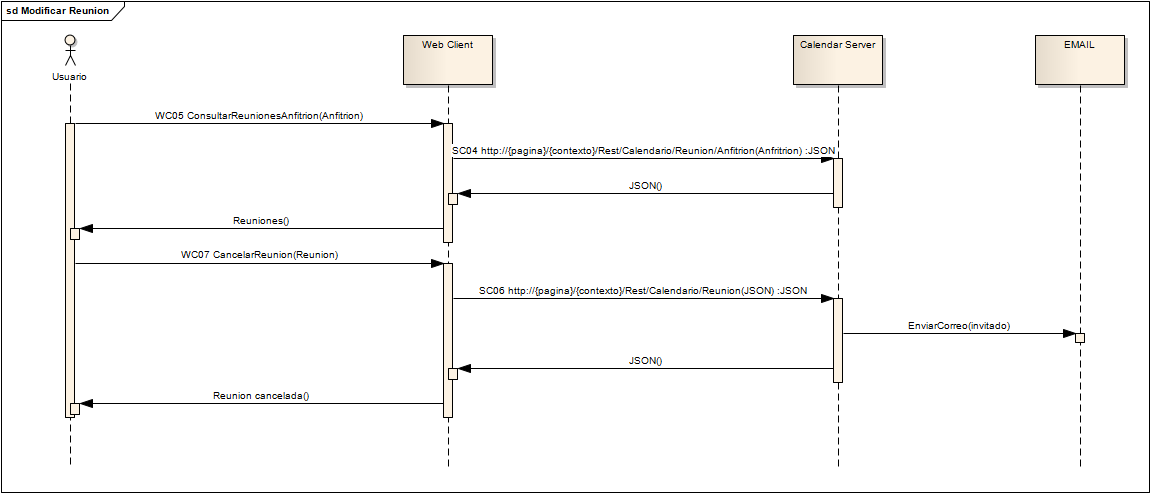
* + - 1. **Crear Reunión**

****

* + - 1. **Modificar Reunión**



* + - 1. **Cancelar Reunión**



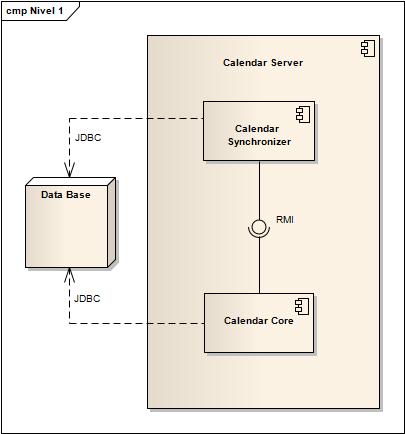
* 1. **Nivel 1**

En este nivel se especifican los componentes Server Calendar y Web Client.

* + 1. **Nivel 1 Web Client**
    2. **Nivel 1 Calendar Server**

El componente Calendar Server se puede divir en los componentes:

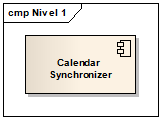
* **Calendar Synchronizer:** Tecnología Java 7, empaquetado en un archivo jar, standalone
* **Calendar Core:** Tecnología Java 7, empaquetado en un archivo war, de forma tal que pueda ser desplegado en un contenedor web (Tomcat, Jetty).
* **Base de datos:** Es externa al sistema, de tipo relacional



* + - 1. **Componente Calendar Syncronizer**

Este componente es una aplicación standalone basa en la tecnología Java 7, es el encargado de realizar la sincronización con los calendarios externos de los usuarios, de forma tal que solo almacene los datos relacionados a los horarios de la reuniones.

Sus interfaces están expuestas por medio del protocolo RMI.



* + - * 1. **Responsabilidades funcionales**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Caso de uso | Nombre | Entrada | Salida |
| **CSY01** | CU11 - Inscribir calendario  R6 – Inscripción de calendarios externos | Programar Sincronización | * Login * Password * URL * TiempoSincronizacion | N/A |
| **CSY02** | Apagar | Apagar | N/A | El componente se apaga cerrando todas las conexiones y dejando los datos consistentes |

* + - * 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **CSY03** | R3 – Los calendarios utilizados para conocer las actividades de los participantes son externos al sistema y deben estar disponibles en formato iCalendar – ics (ej. Google, iCal, etc.). | Este componente es el encargado de obtener los calendarios externos de los usuarios para extraer la información de sus reuniones, por medio del formato ICalendar |
| **CSY04** | R6 - Cada persona debe inscribir sus calendarios usando una URL, las credenciales de autenticación y la frecuencia de actualización | Cuando un calendario es registrado en el sistema, este componente se encarga de realizar su sincronización |
| **CSY05** | R7 - El sistema solo debe almacenar los horarios de las actividades. Cualquier otra información no está permitida | Este componente únicamente almacenará los horarios de las reuniones de los usuarios |

* + - * 1. **Responsabilidades no funcionales**

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **CSY05** | AC01 - Interoperabilidad | Este componente es el encargado de comunicarse con las fuentes de calendario externas, utilizando el protocolo http y el formato ICalendar |

* + - * 1. **Ciclo de vida**
* Este componente se inicia como una aplicación Java standalone.
* Cuando se inicia, se encarga de consultar los usuarios en la base de datos y crear los jobs para la sincronización de los calendarios de cada usuario.
* Este componente se ejecutará hasta que se invoque el método apagar.
  + - 1. **Componente Calendar Core**

Es el componente encargado de exponer las interfaces definidas en el Nivel 0 para el componente Calendar Server y atender las peticiones del componente Web Client, por medio de un proyecto web basado en la tecnología Java 7, este componente se empaqueta en un WAR y se despliega en un contenedor web.

Todas las interfaces expuestas por este componente se encuentran basadas en la metodología REST.



* + - * 1. **Responsabilidades funcionales**

Las responsabilidades funcionales de este componente corresponden a todas las interfaces definidas para el componente Calendar Server.

* + - * 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **CC19** | R1 – Aplicación web, con posibilidad de construir aplicaciones móviles nativas en el futuro. | Todos los servicios expuestos por este componente son servicios REST, de forma tal que pueden ser consumidos fácilmente por cualquier dispositivo. |
| **CC20** | R3 – Los calendarios utilizados para conocer las actividades de los participantes son externos al sistema y deben estar disponibles en formato iCalendar | Se expone un servicio que permite registrar los calendarios externos. |
| **CC21** | R4 – El sistema nunca mostrará un horario en el que no puedan asistir los invitados obligatorios | Este componente se encarga de exponer el servicio para consultar los horarios y se debe asegurar de cumplir con esta restricción, esto lo hace consultando la base de datos para ver las reuniones que tiene pendiente el usuario, que fueron sincronizadas anteriormente por el componente Calendar Synchronizer |
| **CC22** | R5 - Las invitaciones a las reuniones se deben aceptar o rechazar a través del enlace respectivo enviado por correo electrónico, dar clic en el enlace debe ser suficiente para aceptar o rechazar la invitación | En este componente se envían los correos a los invitados a la reunión y se exponen los servicios que permiten aceptar o rechazar una invitación |
| **CC23** | R6 - Cada persona debe inscribir sus calendarios usando una URL, las credenciales de autenticación y la frecuencia de actualización | Este componente se encarga de exponer un servicio para el registro de url de calendarios externos |
| **CC24** | R8 - El sistema debe exponer una URL para exportar los calendarios de la persona en formato iCalendar | Este componente expone una url para obtener los calendarios de un usuario |
| **CC25** | Persistencia | Este componente tiene acceso a la base de datos para realizar la lógica de creación de reuniones, consulta de horarios, crear mensajes, etc. |
| **CC26** | Envió de correos | Este componente se encarga del envió correos electrónicos a los participantes de las reuniones |

* + - * 1. **Responsabilidades no funcionales**

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **CC27** | AC02 – Seguridad | La autenticación se hace por medio de este componente, y la comunicación se realiza mediante el protocolo https |
| **CC28** | AC03 – Flexibilidad | Los servicios REST permiten la invocación de servicios desde diferentes tipos de dispositivos |

* + - * 1. **Ciclo de vida**
* El ciclo de vida de este componente se definido establecido por el contenedor web.
* Cuando este componente se inicia, se replica la información de los usuarios en la base de datos local del sistema.
  + - 1. **Conexión Calendar Core a Calendar Synchronizer**

La conexión entre Calendar Core y Calendar Synchronizer se crea para dar la posibilidad a la aplicación de cargar nuevas direcciones para la sincronización con calendarios de proveedores externos.



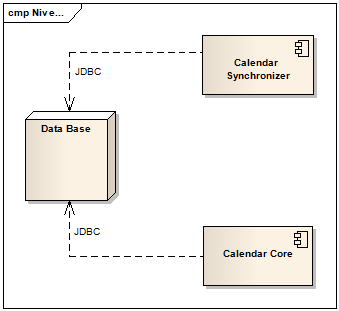
* Este conenxión se realiza mediante el protocolo para la invocación remote de metodos RMI.
  + - * 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **CCCS01** | R6 – Cada persona debe inscribir sus calendarios usando una URL, las credenciales de autenticación y la frecuencia de actualización | Este conexión permite registrar nuevas url para la sincronización de calendarios externos, indicandole al componente Calendar Synchronizer, que se debe registrar una nueva URL. |

* + - * 1. **Ciclo de vida**
* El componente Calendar Core debe abir una conexión RMI cada vez que desee invocar la interfaz externa.
* conexión deberá cerrarse una vez termine la invocación.
* La ruta de conexión se encuentrará en las propiedades del componente Calendar Core
  + - 1. **Conexión Calendar Core, Calendar Synchronizer a Data Base**

Tanto los componentes Calendar Synchronizer como Calendar Core crean conexión a la base de datos, esta conexión se realiza mediante JDBC.

* Calendar Synchronizer la utiliza para sincronizar los horarios de las reuniones de los usuarios del sistema y obtener la información de sincronización (user, password, url).
* Calendar Core la utiliza para llevar a cabo las responsabilidades funcionales, por ejemplo crear una reuión, obtener el listado de usuarios, obtener los invitados que han aceptado la invitación, etc.



* + - * 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **DBCC01** | R4 – El sistema nunca mostrará un horario en el que no puedan asistir los invitados obligatorios | Este conexión permite registrar nuevas url para la sincronización de calendarios externos, indicandole al componente Calendar Synchronizer, que se debe registrar una nueva URL. |

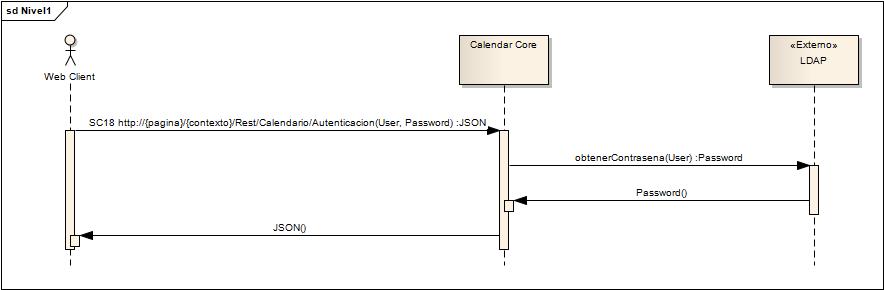
* + - * 1. **Responsabilidades no funcionales**

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **SC28** | AC01 - Interoperabilidad | Esta conexión permite obtener la información de los calendarios externos que ya fueron sincronizados en el sistema, sigue el patron de arquitectura Blackboard |

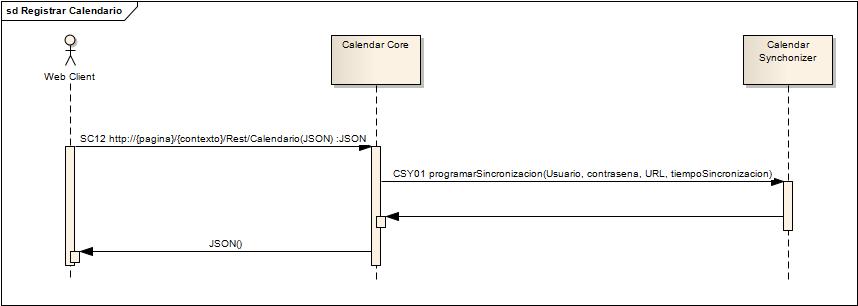
* + - * 1. **Ciclo de vida**
* Cada componente debe obtener un anueva conexión a la base de datos cada vez que requera realizar una operación.
* Una vez terminada la operación se debe cerrar la conexión.
  + - 1. **Flujos de ejecución**

A continuación se indican los flujos de ejecución de los principales casos de uso y las restricciones.

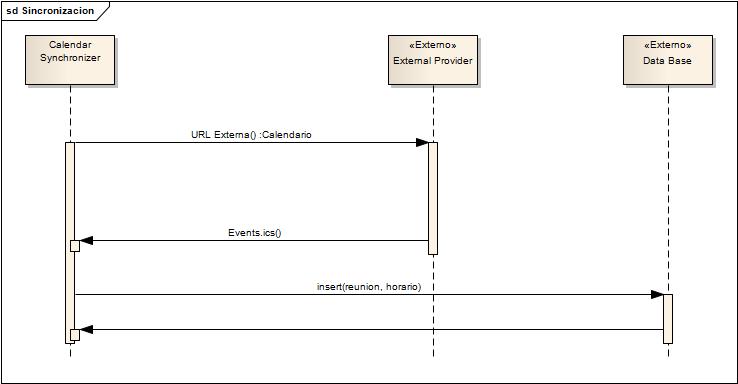
* + - * 1. **Flujos autenticación**

****

* + - * 1. **Flujo registrar calendario**



* + - * 1. **Flujo Sincronizar calendario**



* 1. **Nivel 2**

En este sección se describe los componentes de nivel 2 de Calendar Synchronizer y Calendar Core.

* + 1. **Nivel 2 Calendar Synchronizer**
    2. **Nivel 2 Calendar Core**

En este componente se describen los módulos necesarios para exponer las funcionalidades a los clientes Web y calendarios externos que quieran actualizar las reuniones del sistema



* + - 1. **Módulo Calendar REST**

Es el módulo encargado de recibir las peticiones http y dar respuesta en los formatos indicados en la definición de cada uno de los servicios, utilizando el estilo de arquitectura REST.



* + - * 1. **Responsabilidades funcionales**

Este componente expone las responsabilidades funcionales definidas en el Componente Calendar server que se encuentran en el nivel 0.

* + - * 1. **Restricciones**

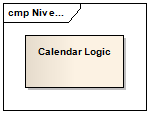
|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **CR20** | R1 – Aplicación web, con posibilidad de construir aplicaciones móviles nativas en el futuro. | Todos los servicios expuestos por este componente son servicios REST, de forma tal que pueden ser consumidos fácilmente por cualquier dispositivo. |
| **CR21** | R3 – Los calendarios utilizados para conocer las actividades de los participantes son externos al sistema y deben estar disponibles en formato iCalendar | Se expone un servicio que permite registrar los calendarios externos. |
| **CR22** | R5 - Las invitaciones a las reuniones se deben aceptar o rechazar a través del enlace respectivo enviado por correo electrónico, dar clic en el enlace debe ser suficiente para aceptar o rechazar la invitación | En este componente se envían los correos a los invitados a la reunión y se exponen los servicios que permiten aceptar o rechazar una invitación |
| **CR23** | R6 - Cada persona debe inscribir sus calendarios usando una URL, las credenciales de autenticación y la frecuencia de actualización | Este componente se encarga de exponer un servicio para el registro de url de calendarios externos |
| **CR24** | R8 - El sistema debe exponer una URL para exportar los calendarios de la persona en formato iCalendar | Este componente expone una url para obtener los calendarios de un usuario |

* + - * 1. **Responsabilidades no funcionales**

|  |  |  |
| --- | --- | --- |
| Id | Nombre | Descripción |
| **CR25** | AC02 – Seguridad | La autenticación se hace por medio de este componente, y la comunicación se realiza mediante el protocolo https |
| **CR26** | AC03 – Flexibilidad | Los servicios REST permiten la invocación de servicios desde diferentes tipos de dispositivos |

* + - * 1. **Ciclo de vida**
* El ciclo de vida de este componente este dado por la invocaciones a los servicios rest, donde cuando se invoca el servicio inicia el ciclo de vida y cuando termina la invocación del serivicio termina el ciclo de vida, esto es devido a que los servicio REST no tiene mantienen el estad
* La sesión del usaurio se mantiene gracias a la cookie de sesión, la cual debe ser enviada en cada invocación para comprar que el usuario ya se haya autenticado.
  + - 1. **Módulo Calendar Logic**

Calendar Logic es el módulo encargado de llevar a cabo la logica de la aplicación web, este logica comprende, la creación de reuniones, la consulta de horarios, la invocación al LDAP para la autenticación, el envio de correos, entre otros.



* + - * 1. **Responsabilidades funcionales**

**CL01**

* **Caso de uso:** CU02 - Agregar participante
* **Descripción:** Se encarga de realizar la consulta de los usuarios registrados en el sistema
* **Nombre:** consultarUsuarios
* **Entradas:** N/A
* **Salidas:**
  + List<Usuario>
    - Nombre (String)
    - Login (String)

**CL02**

* **Caso de uso:** CU03 - Seleccionar horario reunión
* **Descripción:** se encarga de realizar la consulta de los horarios en los que los invitados pueden asistir a una reunión
* **Nombre:** consultarHorarios
* **Entradas:** 
  + Inicio (Date)
  + Fin (Date)
  + Duracion (int)
  + CantidadResultados (int)
  + List<Invitado>
    - Login (String)
    - Obligatorio (boolean)
* **Salidas:**
  + List<Horario>
    - Fecha (Date)
    - cantidadAsistentes (int)
    - List<Asistente>
      * Obligatorio (boolean)
      * Asiste (boolean)

**CL03**

* **Caso de uso:** CU01 - Crear reunión
* **Descripción:** Se encarga de persistir una reunión en la base de datos y enviar los correos a los invitados a la reunión
* **Nombre:** crearReunion
* **Entradas:** 
  + Nombre (String)
  + Descripcion (String)
  + Horario (Date)
  + ubicacion (String)
  + Anfitrion (String)
  + List<Participante>
    - Login (String)
    - Obligatorio (boolean)
* **Salidas:** N/A

**CL04**

* **Caso de uso:** CU04 - Editar información básica reunión - CU05 - Cancelar reunión
* **Descripción:** Se encarga de realizar la consulta de las reuniónes programadas por un anfitrion
* **Nombre:** consultarReunionPorAnfitrion
* **Entradas:** 
  + loginAnfitrion
* **Salidas:** 
  + List<reunion>
    - Id (String)
    - Nombre (String)
    - Descripcion (String)
    - Horario (Date)
    - Ubicacion (String)
    - Anfitrion (String)
    - List<Participante>
      * Login (String)
      * Obligatorio (Boolean)

**CL05**

* **Caso de uso:** CU04 - Editar información básica reunión
* **Descripción:** Se encarga de modificar los datos básicos de una reunión y enviar correo a los invitados
* **Nombre:** modificarReunion
* **Entradas:** 
  + idReunion (String)
  + Nombre (String)
  + Descripción (String)
  + Ubicación (String)
  + Horario (Date)
  + Anfitrion (String)
* **Salidas:** N/A

**CL06**

* **Caso de uso:** CU05 - Cancelar reunión
* **Descripción:** Se encarga de cambiar el estado de la reunión en la base de datos y enviar correos a los responsables invitados a la reunión
* **Nombre:** cancelarReunion
* **Entradas:** 
  + idReunion (String)
* **Salidas:** N/A

**CL07**

* **Caso de uso:** CU08 - Consultar reuniones asociadas
* **Descripción:** Realiza la consulta de la reuniones de un usuario
* **Nombre:** consultarReuniones
* **Entradas:** 
  + Login (String)
* **Salidas:** 
  + List<reunion>
    - Id (String)
    - Nombre (String)
    - Descripcion (String)
    - Horario (Date)
    - Ubicacion (String)
    - Anfitrion (String)
    - List<Participante>
      * Login (String)
      * Obligatorio (Boolean)

**CL08**

* **Caso de uso:** CU08 - Consultar reuniones asociadas
* **Descripción:** realiza la consulta de los mensajes del asociados a una reunión
* **Nombre:** consultarMensajes
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Mensajes>
    - Login (String)
    - Mensaje (String)
    - Fecha (Date)

**CL09**

* **Caso de uso:** CU08 - Consultar reuniones asociadas
* **Descripción:** se encarga de realiza la consulta de los archivos asociados a una reunión
* **Nombre:** consultarArchivos
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Archivo>
    - Login (String)
    - NombreArchivo (String)
    - Fecha (Date)

**CL10**

* **Caso de uso:** CU09 - Consultar participantes confirmados
* **Descripción:** realiza la consulta de los invitados que han confirmado la invitación a la reunión
* **Nombre:** consultarParticipantesConfirmados
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Confirmado>
    - Login (String)
    - Nombre (String)

**CL11**

* **Caso de uso:** CU10 - Consultar invitados rechazados
* **Descripción:** realiza la consulta de los invitados que han rechazado la invitación a la reunión
* **Nombre:** consultarParticipantesRechazados
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Rechazado>
    - Login (String)
    - Nombre (String)

**CL12**

* **Caso de uso:** CU11 - Inscribir calendario - R6 – Inscripción de calendarios externos
* **Descripción:** permite almacenar la información para la sincronización de cuentas de calendario externas de los usuarios.
* **Nombre:** inscribirCalendario
* **Entradas:** 
  + URL (String)
  + Login (String)
  + Contraseña (String)
  + IntervaloSinc (int - minutos)
* **Salidas:** N/A

**CL13**

* **Caso de uso:** CU12 - Compartir mensaje a participantes reunión
* **Descripción:** almacena los datos de un mensaje asociado a una reunión
* **Nombre:** compartirMensaje
* **Entradas:** 
  + IdReunion (String)
  + Mensaje (String)
  + Login (String)
  + Fecha (Date)
* **Salidas:** N/A

**CL14**

* **Caso de uso:** CU13 – Compartir archivo a participantes reunión
* **Descripción:** almacena un archivo asociado a una reunión
* **Nombre:** compartirArchivo
* **Entradas:** 
  + IdReunion (String)
  + Archivo (byte[])
  + Login (String)
  + Fecha (Date)
* **Salidas:** N/A

**CL15**

* **Caso de uso:** CU13 – Compartir archivo a participantes reunión
* **Descripción:** consulta los bytes de un archivo
* **Nombre:** obtenerArchivo
* **Entradas:** 
  + IdReunion (String)
  + Archivo (String)
* **Salidas:** 
  + Archivo (byte[])

**CL16**

* **Caso de uso:** CU07 - Rechazar invitación
* **Descripción:** rechaza una invitación a una reunión, y envia correo al anfitrión
* **Nombre:** rechazarInvitacion
* **Entradas:** 
  + IdReunion (String)
  + Login (String)
* **Salidas:** N/A

**CL17**

* **Caso de uso:** CU06 - Aceptar invitación
* **Descripción:** acepta una invitación a una reunión, y envia correo al anfitrión
* **Nombre:** aceptarInvitacion
* **Entradas:** 
  + IdReunion (String)
  + Login (String)
* **Salidas:** N/A

**CL18**

* **Caso de uso:** R2 - La autenticación debe hacerse usando el sistema LDAP con el que ya se cuenta
* **Nombre:** autenticar
* **Entradas:** 
  + Usuario (String)
  + Constraseña (String)
* **Salidas:** 
  + Autenticado (boolean)

**CL19**

* **Caso de uso:** R8 – Exponer calendarios usuario
* **Nombre:** obtenerCalendario
* **Entradas:** 
  + Usuario (String)
* **Salidas:** 
  + Archivo ics
    - * 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **SC20** | R4 – El sistema nunca mostrará un horario en el que no puedan asistir los invitados obligatorios | Este componente se encarga de la lógica necesaria para determina los horarios de las reuniones |
| **SC21** | Envió de correos | Este componente se encarga del envió correos electrónicos a los participantes de las reuniones |

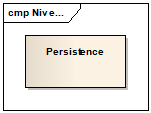
* + - * 1. **Ciclo de vida**

El ciclo de vida de este módulo se encuentra definido por la invocación de los servicios REST, esto significa que desde el módulo Calendar REST, se inicia una instancia del módulo Calendar Logic, para llevar a cabo las tareas requeridas, cuando se retorna al Calendar REST, se termina la instancia del módulo.

* + - 1. **Módulo Persistencia**

Este módulo se encarga de comunicarse con la base de datos, por medio de consultas SQL y es el encargado de preveer los datos del sistema al módulo Calendar Logic.

Los datos de configuración a la base de datos se encuentran en un archivo de configuración.



* + - * 1. **Responsabilidades funcionales**

**PS01**

* **Caso de uso:** CU02 - Agregar participante
* **Nombre:** consultarUsuarios
* **Entradas:** N/A
* **Salidas:**
  + List<Usuario>
    - Nombre (String)
    - Login (String)

**PS02**

* **Caso de uso:** CU03 - Seleccionar horario reunión
* **Descripción:** Esta consulta se realiza para poder evaluar los horarios de las reuniones de una persona y poder determinar si pude o no asistir a determinada reunión
* **Nombre:** consultarReuniones
* **Entradas:** 
  + Login (String)
  + Fecha inicio (Date)
  + Fecha fin (Date)
* **Salidas:**
  + List<Reunion>
    - idReunion (String)
    - Fecha Incio (Date)
    - Fecha fin (Date)

**PS03**

* **Caso de uso:** CU01 - Crear reunión
* **Descripción:** se persiste la reunión en la base de datos
* **Nombre:** persistirReunion
* **Entradas:** 
  + Nombre (String)
  + Descripcion (String)
  + Horario (Date)
  + Ubicacion (String)
  + Anfitrion (String)
  + List<Participante>
    - Login (String)
    - Obligatorio (boolean)
* **Salidas:** N/A

PS**04**

* **Caso de uso:** CU04 - Editar información básica reunión - CU05 - Cancelar reunión
* **Descripción:** Permite realizar la consulta de las reuniones que tiene un anfitrión
* **Nombre:** consultarReunionesPorAnfitrion
* **Entradas:** 
  + loginAnfitrion
* **Salidas:** 
  + List<reunion>
    - Id (String)
    - Nombre (String)
    - Descripcion (String)
    - Horario (Date)
    - Ubicacion (String)
    - Anfitrion (String)
    - List<Participante>
      * Login (String)
      * Obligatorio (Boolean)

**PS05**

* **Caso de uso:** CU04 - Editar información básica reunión
* **Descripción:** permite actualizar la información basica de una reunión
* **Nombre:** actualizarReunion
* **Entradas:** 
  + idReunion (String)
  + Nombre (String)
  + Descripción (String)
  + Ubicación (String)
  + Horario (Date)
  + Anfitrion (String)
* **Salidas:** N/A

**PS06**

* **Caso de uso:** CU05 - Cancelar reunión
* **Descripción:** actualiza una reunion a estado cancelada
* **Nombre:** cancelarReunion
* **Entradas:** 
  + idReunion (String)
* **Salidas:** N/A

**PS07**

* **Caso de uso:** CU08 - Consultar reuniones asociadas
* **Descripción:** realiza la consulta de la reuniones de un usuario basandose en el login
* **Nombre:** consultarReuniones
* **Entradas:** 
  + Login (String)
* **Salidas:** 
  + List<reunion>
    - Id (String)
    - Nombre (String)
    - Descripcion (String)
    - Horario (Date)
    - Ubicacion (String)
    - Anfitrion (String)
    - List<Participante>
    - Login (String)
    - Obligatorio (Boolean)

**PS08**

* **Caso de uso:** CU08 - Consultar reuniones asociadas
* **Descripción:** realiza la consulta de los mensajes del asociados a una reunión
* **Nombre:** consultarMensajes
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Mensajes>
    - Login (String)
    - Mensaje (String)
    - Fecha (Date)

**PS09**

* **Caso de uso:** CU01 - Crear reunión - CU04 - Editar información básica reunión - CU05 - Cancelar reunión
* **Descripción:** realiza la consulta de los correos de los usuarios
* **Nombre:** consultarCorreosUsuarios
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + List<Correo>

**PS10**

* **Caso de uso:** CU08 - Consultar reuniones asociadas
* **Descripción:** se encarga de realiza la consulta de los archivos asociados a una reunión
* **Nombre:** consultarArchivos
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Archivo>
    - Login (String)
    - NombreArchivo (String)
    - Fecha (Date)

**PS11**

* **Caso de uso:** CU09 - Consultar participantes confirmados
* **Descripción:** realiza la consulta de los invitados que han confirmado la invitación a la reunión
* **Nombre:** consultarParticipantesConfirmados
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Confirmado>
    - Login (String)
    - Nombre (String)

**PS12**

* **Caso de uso:** CU10 - Consultar invitados rechazados
* **Descripción:** realiza la consulta de los invitados que se encuentran con estado rechazado en la invitación en la base de datos
* **Nombre:** consultarParticipantesRechazados
* **Entradas:** 
  + idReunion (String)
* **Salidas:** 
  + IdReunion (String)
  + List<Rechazado>
    - Login (String)
    - Nombre (String)

**PS13**

* **Caso de uso:** CU11 - Inscribir calendario - R6 – Inscripción de calendarios externos
* **Descripción:** permite almacenar en la base de datos la información para la sincronización de cuentas de calendario externas de los usuarios.
* **Nombre:** inscribirCalendario
* **Entradas:** 
  + URL (String)
  + Login (String)
  + Contraseña (String)
  + IntervaloSinc (int - minutos)
* **Salidas:** N/A

**PS14**

* **Caso de uso:** CU12 - Compartir mensaje a participantes reunión
* **Descripción:** almacena en la base de datos un mensaje asociado a una reunión
* **Nombre:** persistirMensaje
* **Entradas:** 
  + IdReunion (String)
  + Mensaje (String)
  + Login (String)
  + Fecha (Date)
* **Salidas:** N/A

**PS15**

* **Caso de uso:** CU13 – Compartir archivo a participantes reunión
* **Descripción:** persiste en la base de datos un archivo asociado a una reunión
* **Nombre:** persistirArchivo
* **Entradas:** 
  + IdReunion (String)
  + Archivo (byte[])
  + Login (String)
  + Fecha (Date)
* **Salidas:** N/A

**PS16**

* **Caso de uso:** CU13 – Compartir archivo a participantes reunión
* **Descripción:** consulta en la base de datos los bytes del archivo
* **Nombre:** consultarArchivo
* **Entradas:** 
  + IdReunion (String)
  + Archivo (String)
* **Salidas:** 
  + Archivo (byte[])

**PS17**

* **Caso de uso:** CU07 - Rechazar invitación
* **Descripción:** actualiza la invitación de una reunión a un estado de rechazada
* **Nombre:** rechazarInvitacion
* **Entradas:** 
  + IdReunion (String)
  + Login (String)
* **Salidas:** N/A

**PS18**

* **Caso de uso:** CU06 - Aceptar invitación
* **Descripción:** actualiza la invitación de una reunión a un estado de aceptada
* **Nombre:** aceptarInvitacion
* **Entradas:** 
  + IdReunion (String)
  + Login (String)
* **Salidas:** N/A

**PS19**

* **Descripción:** Realiza la consulta de una reunión por el id
* **Nombre:** consultarReunion
* **Entradas:** 
  + IdReunion (String)
* **Salidas:** 
  + Id (String)
  + Nombre (String)
  + Descripcion (String)
  + Horario (Date)
  + Ubicacion (String)
  + Anfitrion (String)
  + List<Participante>
  + Login (String)
  + Obligatorio (Boolean)
    - * 1. **Restricciones**

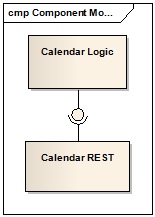
|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **PS20** | Persistencia | Este componente tiene acceso a la base de datos para realizar la lógica de creación de reuniones, consulta de horarios, crear mensajes, etc. |

* + - * 1. **Ciclo de vida**

Para cada una de las invocaciones realizadas por Calendar Logic, se creará una nueva instancia de este módulo.

* + - 1. **Conexión Calendar REST a Calendar Logic**

La conexón entre estos dos módulos, se realiza interfaces java e instancia de clases que permiten acceder a los métodos de Calendar Logic.

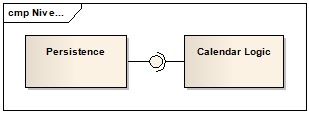


* + - * 1. **Ciclo de vida**

Una instancia del módulo Calendar Logic se crea cada vez que que se realiza una invocación desde el módulo Calendar REST.

* + - 1. **Conexión Calendar Logic a Persistence**

La conexión entre estos dos módulos es aquella que da acceso a la persistencia de los datos que son tratados por el módulo de lógica.



* + - * 1. **Restricciones**

|  |  |  |
| --- | --- | --- |
| Id | Restricción | Descripción |
| **SC22** | R4 – El sistema nunca mostrará un horario en el que no puedan asistir los invitados obligatorios | Esta conexión permite realizar las consultas de las reuniones indicando los horarios en los que pueden realizar las reuniones |
| **SC27** | Persistencia | Esta conexión se encentra encargada de la persistencia de la aplicación |

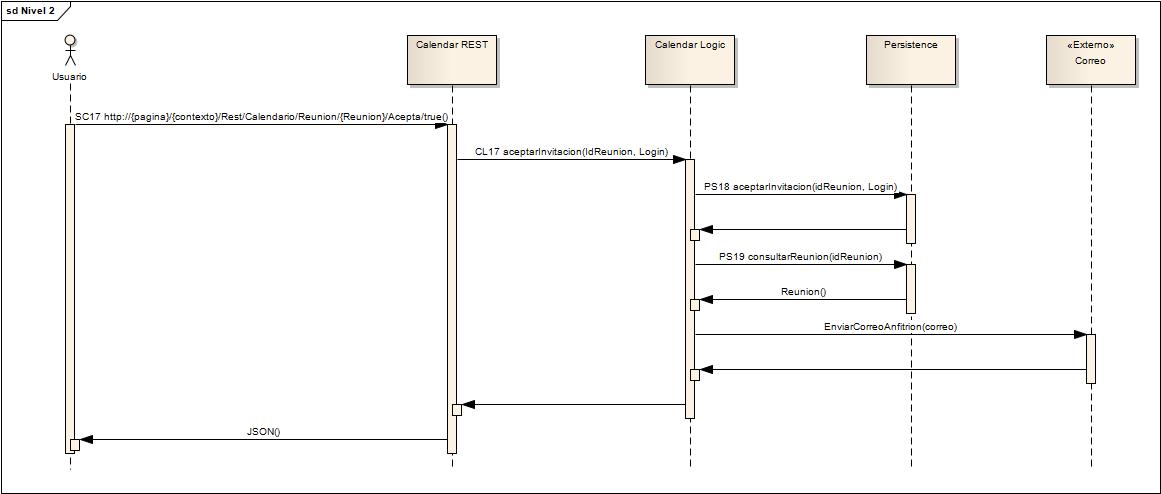
* + - * 1. **Ciclo de vida**

Se crea una instancia de este módulo por cada invocación realizada al módulo Calendar Logic

* + - 1. **Flujos de ejecución**

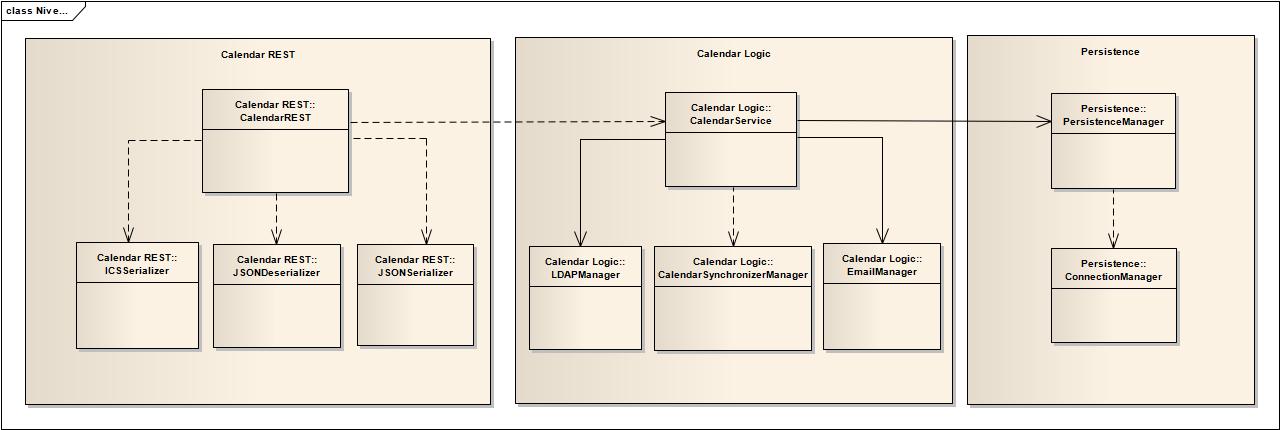
A continuación se indican algunos flujos de ejecución del sistema.

* + - * 1. **Flujo aceptar invitación**

****

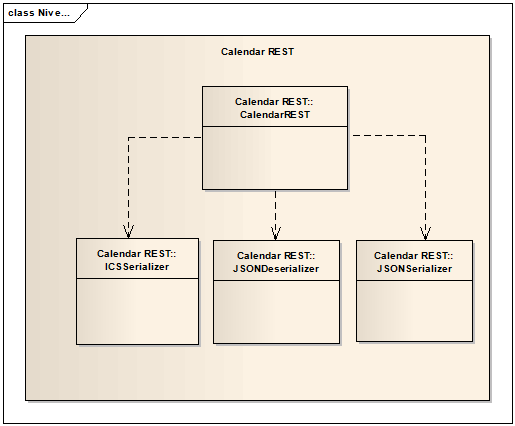
* 1. **Nivel 3**

En este nivel se indican las principales clases de cada uno de los modulos definidos en el nivel 2.

****

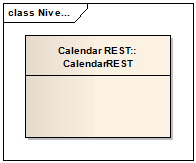
* + 1. **Nivel 3 CalendarREST**

En este nivel se encuentran las clases encargadas de la invocación de los servicios REST y la serialización a los diferentes formatos de respuesta de los servicios.

****

* + - 1. **CalendarREST**

Esta clase es controlada por el contenedor Web, y se encarga de recibir las peticiones de los servicios invocados desde los clientes Web y la seguridad de este componente se encontrará controlada por medio del API JAAS. Para la implementación de los servicios REST se utilizará el API Jersey.

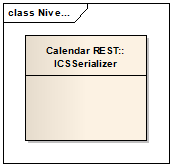


* + - * 1. **Responsabilidades funcionales**

Las responsabilidades funcionales de esta clase se encuentran definidas en las resposabilidades funcionales del componente Calendar Server.

* + - 1. **ICSSerializer**

Este clase se encarga de transformar todas las reuniones de un usuario en un archivo en formato ics.

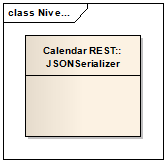


* + - * 1. **Responsabilidades funcionales**

**ICSS01**

* **Descripción:** Se encarga de tomar la lista de reuniones de un usuario y transformarla en un archivo con formato ics
* **Nombre:** serializarICS
* **Entradas:** 
  + List<Reunion>
* **Salidas:** 
  + Archivo formato ics
    - 1. **JSONSerilizer**

Este clase se encarga de tomar los datos retornados por Calendar Logic y transformarlos en formato JSON

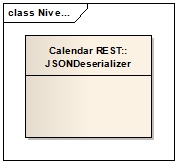


* + - * 1. **Responsabilidades funcionales**

Las responsabilidades funcionales de esta clase se encuentran definidas en las salidas de las resposabilidades funcionales del componente Calendar Server.

* + - 1. **JSONDeserializer**

Este clase se encarga de transformar las entradas de los servicios REST que envian JSON como parametros.

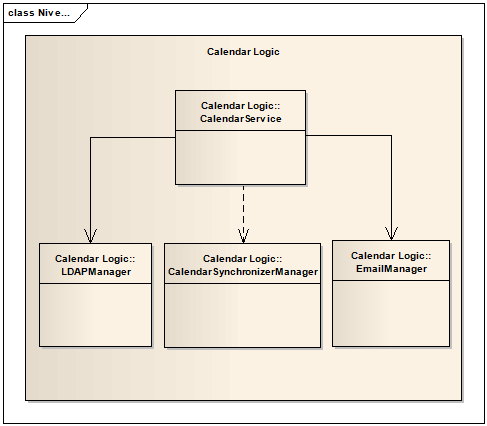


* + - * 1. **Responsabilidades funcionales**

Las reponsibilidades funcionales de este componente se encuentran en la transformación de las entradas de tipo JSON en el componente Calendar Server.

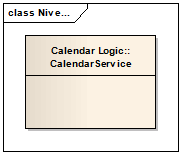
* + 1. **Nivel 3 CalendarLogic**

En este nivel se encuentran las clases encargadas de la logica del sistema para la creación de reuniones, envio de correos, invocación al LDAP, etc.



* + - 1. **CalendarService**

Esta clase se encarga de realizar la logica del sistema,toma las entradas de los servicios rest y realiza consultas necesarios, invoca al LDAP, invoca el envío de emails, etc.

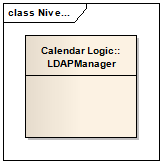


* + - * 1. **Responsabilidades funcionales**

Las responsabilidades funcionales de esta clase se encuentran definidas en las responsabilidades funcionales del módulo Calendar Logic.

* + - 1. **LDAPManager**

Es la clase que se encarga de establcer la conexión con el LDAP donde se encuentran los usuarios del sistema. La conexión al LDAP se encuentra definia en archivos de propiedades.



* + - * 1. **Responsabilidades funcionales**

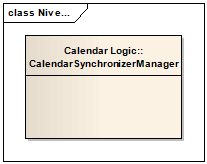
**LDAP01**

* **Descripción:** Se encarga de establecer la conexión y realizar la consulta de los usuarios
* **Nombre:** obtenerUsuarios
* **Entradas:** N/A
* **Salidas:** 
  + List<Usuario>

**LDAP02**

* **Descripción:** Se encarga de consultar el password del usuario
* **Nombre:** obtenerPassword
* **Entradas:** 
  + Usuario
* **Salidas:** 
  + Password
    - 1. **CalendarSyncrhronizerManager**

Esta clase se encarga de administrar la invocación a la interfaz RMI registrar URL de calendarios externos.

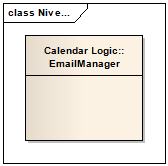


* + - * 1. **Responsabilidades funcionales**

**CSM01**

* **Descripción:** Se encarga de indicarle al componente Calendar Synchronizer que debe registrar una nueva URL de calendarios externos
* **Nombre:** sincronizarURL
* **Entradas:** 
  + Usuario
  + Password
  + URL
  + intervaloSincronizacion
* **Salidas:** N/A
  + - 1. **EmailManager**

Es la clase encargada del envío de correos, esto lo hace por medio del API Java Mail.

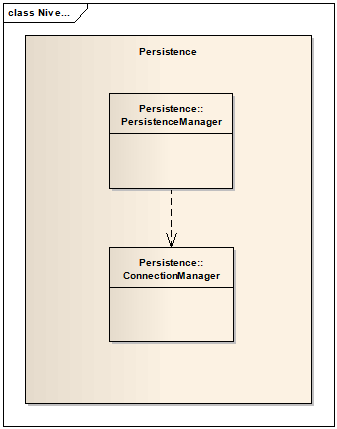


* + - * 1. **Responsabilidades funcionales**

**EM01**

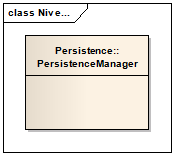
* **Descripción:** Se encarga de enviar un correo.
* **Nombre:** enviarCorreo
* **Entradas:** 
  + Correo
* **Salidas:** N/A
  + 1. **Nivel 3 Persistence**

En este nivel se encuentran definidos los elementos de persistencia.



* + - 1. **PersistenceManager**

Esta es la clase encargada de crear las consultas a la base de datos, y retornar los objetos de tipo Value Object que seran utilizados en la logica.

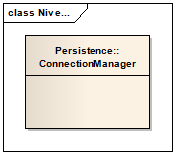


* + - * 1. **Responsabilidades funcionales**

Las responsabilidades funcionales de este componente se encuentran definidas en el módulo de persistencia.

* + - 1. **ConnectionManager**

Es la clase encargada de pasar la conexión a la base de datos a la clase PersistenceManager



* + - * 1. **Responsabilidades funcionales**

**CM01**

* **Descripción:** Se encarga de obtener una conexión y retornarla
* **Nombre:** obtenerConexion
* **Entradas:** N/A
* **Salidas:** Conexion