



**DEPARTAMENTO
DE COMPUTACION**

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico Número 2

ReciBarFiesta

Ingeniería del Software I+II

Grupo 7

Integrante	LU	Correo electrónico
Ciruelos Rodríguez, Gonzalo	063/14	gonzalo.ciruelos@gmail.com
Costa, Manuel José Joaquín	035/14	manucos94@gmail.com
Gatti, Mathias Nicolás	477/14	mathigatti@gmail.com
Maddonni, Axel Ezequiel	200/14	axel.maddonni@gmail.com
Thibeault, Gabriel	114/13	gabriel.eric.thibeault@gmail.com

Facultad de Ciencias Exactas y Naturales

Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Índice

1. Introducción	3
2. Casos de uso	4
2.1. Descripción de los Casos de uso	4
3. Riesgos	6
4. Plan de Proyecto	9
4.1. WBS	9
4.2. Primera iteración	9
4.2.1. Justificación	10
4.3. Resto de las iteraciones	11
4.3.1. Casos de uso por iteración	11
5. Especificación de Escenarios	13
5.1. Escenarios de Seguridad	13
5.2. Escenarios de Disponibilidad	13
5.3. Escenarios de Modificabilidad	13
5.4. Escenarios de Performance	13
6. Arquitectura	14
6.1. Diagramas de Componentes y Conectores con Deployment	14
6.1.1. Nivel 0	14
6.1.2. ReciBarFiesta	14
6.1.3. Manejador de Eventos	14
6.1.4. Manejador de Usuarios	14
6.1.5. Manejador de Reservas	14
6.1.6. Manejador de Publicidades	14
6.2. Explicación de la Arquitectura	14
6.2.1. Manejador de Eventos	14
7. Referencias	16

1. Introducción

El trabajo práctico se basa en el desarrollo de una aplicación llamada ReciBarFiesta. El trabajo consiste de dos entregas, la primera concierne a la planificación y la segunda a la arquitectura. La parte de planificación consiste en:

- un plan del proyecto,
- una lista de casos de uso,
- un análisis de riesgos.

En este trabajo presentamos toda esa información (en un orden que nos pareció más sensato) y justificamos nuestras decisiones.

Código	Caso de uso	Horas Hombre
CU01	Incluyendo un evento cultural	30
CU02	Eliminando un evento cultural	8
CU03	Buscando evento	56
CU04	Rankeando evento	60
CU05	Logueándose	59
CU06	Compartiendo información en redes sociales	8
CU07	Editando un evento cultural	8
CU08	Registrándose	20
CU09	Verificando contenido inapropiado automáticamente	48
CU10	Verificando contenido inapropiado manualmente	8
CU11	Reservando hoteles	56
CU12	Reservando vuelos	48
CU13	Enviando publicidad	20
CU14	Visualizando caminos	48
CU15	Realizando reserva	56
CU16	Filtrando resultados	20
CU17	Obteniendo estadísticas sobre los usuarios	32
CU18	Combinando eventos	32

Tabla 1 Casos de uso

2. Casos de uso

2.1. Diagrama de Casos de uso

2.2. Descripción de los Casos de uso

1. **Incluyendo un evento cultural:** se refiere a la funcionalidad para agregar nuevos eventos, como recitales, charlas, festivales, etc. Solo lo pueden hacer usuarios autorizados (inicialmente solo miembros de dependencias del Estado).
2. **Eliminando un evento cultural:** se refiere a la funcionalidad por la cual una dependencia del Estado puede eliminar un evento cultural previamente creado.
3. **Editando un evento cultural:** se refiere a la funcionalidad por la cual una dependencia del Estado puede modificar los datos de un evento cultural previamente creado.
4. **Rankeando evento:** se refiere a la funcionalidad que le permite a un miembro identificado de la comunidad calificar un evento en distintas categorías como, por ejemplo, "Organización", "Calidad" y "Ubicación". Solo usuarios identificados podrán realizar esta acción, y se ponderará la calificación de acuerdo a la categoría del usuario (inicial, intermedio o experto).
5. **Buscando evento:** se refiere a la funcionalidad por la cual los usuarios de la aplicación buscarán eventos, pudiendo filtrar los resultados por una serie de criterios (por ejemplo: fecha, temática, ubicación, etc) usando el caso de uso "Filtrando resultados" (CU16).
6. **Compartiendo información en redes sociales:** se refiere a la funcionalidad por la cual los miembros de la comunidad podrán compartir los diferentes eventos disponibles en

la aplicación en diferentes redes sociales como Facebook o Twitter, de manera que otras personas que no necesariamente tienen la aplicación puedan conocer su existencia.

7. **Logueandose:** se refiere a la funcionalidad por la cual los usuarios anónimos con una cuenta previamente creada podrán identificarse con el sistema, lo cual les habilitará a tener información personalizada (por ejemplo, según el barrio en el que viven), valorar eventos, y poder realizar reservas de vuelos y hoteles.
8. **Registrándose:** se refiere a la funcionalidad a través de la cual los usuarios anónimos pueden crearse una cuenta en el sistema, para eventualmente poder identificarse y aprovechar los privilegios que esto acarrea.
9. **Verificando contenido inapropiado automatizado:** se refiere a la funcionalidad con la que el sistema utiliza la API de moderación de contenidos. La misma pone puntajes de 1 (contenido inapropiado), 2 (contenido dudoso, proceder a verificación manual) o 3 (contenido seguro)
10. **Verificando contenido inapropiado manualmente:** se refiere a la funcionalidad para que un moderador pueda determinar si un determinado contenido es apropiado o no. Este caso de uso solo se activa para aquellos contenidos que son calificados con 2 por CU9.
11. **Enviando Publicidad:** se refiere a la funcionalidad que le permite a los sponsors económicos de la aplicación enviar su material publicitario para ser mostrado por la página.
12. **Visualizando Caminos:** se refiere a la funcionalidad que permite a los usuarios visualizar el o los caminos (en caso de que se hayan seleccionado múltiples eventos) desde su posición actual hasta cada evento. Se ofrecen diferentes tipos de rutas: caminando, en auto particular, en taxi (usando la aplicación del gobierno de la Ciudad), y en transporte público. Eventualmente pueden combinarse eventos (CU18).
13. **Reservando Hoteles:** se refiere a la funcionalidad por la cual la API del sistema de la Cámara de Hoteles y Afines de la Ciudad (CAHAC) podrá registrar reservas de hoteles efectuadas por usuarios de la aplicación.
14. **Reservando Vuelos:** se refiere a la funcionalidad por la cual la API del sistema de Aerolíneas Argentinas podrá registrar reservas de vuelos efectuadas por usuarios de la aplicación.
15. **Obteniendo estadísticas sobre los usuarios:** se refiere a la API desarrollada para que las empresas de Big Data puedan obtener los datos estadísticos sobre los usuarios.
16. **Filtrando resultados:** se refiere a la funcionalidad que le permite a los usuarios de la aplicación filtrar listas de eventos según diversos criterios a determinar.
17. **Realizando Reserva:** se refiere a la funcionalidad por la cual los miembros de la comunidad podrán seleccionar y reservar vuelos/hoteles para participar de eventos publicados en la aplicación utilizando los servicios provistos por Aerolíneas Argentinas, para vuelos, y de la Cámara de Hoteles y Afines de la Ciudad, para alojamientos .
18. **Combinando eventos:** se refiere a la funcionalidad que extiende al caso CU10, permitiendo combinar eventos de forma tal que se obtenga el mejor camino que pasa por todas las ubicaciones.

3. Riesgos

Riesgo 1

- **Descripción:** Dado que se pueden dar problemas de conexión y se requiere que todos los días extraigan los datos (por limitaciones de espacio) se podría llegar a fallar en entregar los datos pedidos a las compañías de big data.
- **Probabilidad:** Baja
- **Impacto:** Medio
- **Exposición:** Baja
- **Mitigación:** Tener el espacio necesario para almacenar, como mínimo, el doble de la información que esperamos almacenar en promedio.
- **Plan de contingencia:** En caso de que el servicio de big data este caído por varios días y no queramos perder información, se debería hacer backup manual de los datos antes de que sean sobrescritos.

Riesgo 2

- **Descripción:** No tener la integración con Aerolíneas y CAHAC hecha para el 2017.
- **Probabilidad:** Media
- **Impacto:** Alto
- **Exposición:** Alta
- **Mitigación:** Completar la integración en las primeras dos iteraciones del desarrollo. Asignar a los mejores programadores y más recursos a realizar esta tarea.
- **Plan de contingencia:** En caso que muchos eventos imprevistos suceden y no se llegue a tiempo, se podría pedir una extensión a Aerolíneas con la suficiente anticipación. Además, se podría hacer que el equipo abandone el desarrollo de otras funcionalidades y se aboque en su totalidad a finalizar la integración.

Riesgo 3

- **Descripción:** Incapacidad del sistema de moderación principal para procesar los pedidos en tiempo real (Tanto de congestión como por caída del sistema).
- **Probabilidad:** Alta (Depende fuertemente del sistema).
- **Impacto:** Baja
- **Exposición:** Media
- **Mitigación:** Realizar estudios para tener una excelente aproximación de tráfico, de tal manera de poder comunicarlo a la empresa que provee el servicio. Tener un servicio secundario, de tal manera que si el primario se cae o funciona mal, podamos usarlo.

- **Plan de contingencia:** En caso que todo falle y no podamos analizar comentarios en tiempo real, podemos almacenarlos en una cola y para que sean analizados más adelante cuando todo vuelva a la normalidad. En caso que el sistema tarde mucho en volver a la normalidad, perderemos comentarios.

Riesgo 4

- **Descripción:** Perder la conexión a internet.
- **Probabilidad:** Baja
- **Impacto:** Alto
- **Exposición:** Media
- **Mitigación:** Tener una conexión secundaria (o más de una) de tal manera que si se cae, podamos usar esa.
- **Plan de contingencia:** Deberíamos tener un sistema rotatorio de on-calls, de tal manera que si se cae internet, en cualquier momento se le avise al on-call y este se ocupe de trasladar el servicio a una nueva conexión.

Riesgo 5

- **Descripción:** Filtración de datos privados
- **Probabilidad:** Baja
- **Impacto:** Alto
- **Exposición:** Media
- **Mitigación:** Diseñar un sistema de detección de intrusiones. Enviar datos al servicio de big data de forma encriptada y des-asociada del nombre del usuario.
- **Plan de contingencia:** Almacenar los datos personales de las cuentas de las personas de forma encriptada, de tal manera que si se filtran no sean legibles.

Riesgo 6

- **Descripción:** Usuarios creados con la intención de manipular la valoración de los eventos
- **Probabilidad:** Alta
- **Impacto:** Bajo
- **Exposición:** Media
- **Mitigación:** Pedir datos personales a la hora de crear la cuenta (Por ejemplo DNI o CUIT).
- **Plan de contingencia:** Diseñar un sistema que detecte que un evento está recibiendo muchas valoraciones y avise al on-call para que realice una revisión manual de la situación.

Riesgo 7

- **Descripción:** Renuncia de personal clave del equipo de desarrollo.
- **Probabilidad:** Baja
- **Impacto:** Medio
- **Exposición:** Baja
- **Mitigación:** Distribuir el trabajo de manera de que nadie esté demasiado sobrecargado y todos conozcan todo lo que se hace.
- **Plan de contingencia:** En caso de renuncias, reorganizar el equipo de tal manera que el programador que más conozco el trabajo del programador que renunció retome sus tareas.

Riesgo 8

- **Descripción:** Que el tráfico promedio de usuarios sea más del previsto y los sistemas internos estén continuamente sobrecargados.
- **Probabilidad:** Baja (Suponiendo una buena estimación inicial)
- **Impacto:** Alto
- **Exposición:** Media
- **Mitigación:** Hacer estimaciones lo mejor posibles.
- **Plan de contingencia:** Diseñar un balanceador de cargas y usarlo. Tener varias instancias del servidor corriendo en paralelo y totalmente replicadas.

Riesgo 9

- **Descripción:** Si la cantidad de ventas de hoteles y boletos de avión no fuera la suficiente, Aerolíneas podría retirar su inversión.
- **Probabilidad:** Baja (Muy difícil de estimar).
- **Impacto:** Medio
- **Exposición:** Media
- **Mitigación:** Hacer que la integración con Aerolíneas sea impecable. Diseñar publicidades llamativas pero poco invasivas.
- **Plan de contingencia:** Cuando la aplicación esté funcionando, invertir tempranamente en hardware y ahorrar dinero para poder hacer una inversión grande en caso que Aerolíneas Argentinas retire la suya.

4. Plan de Proyecto

4.1. WBS

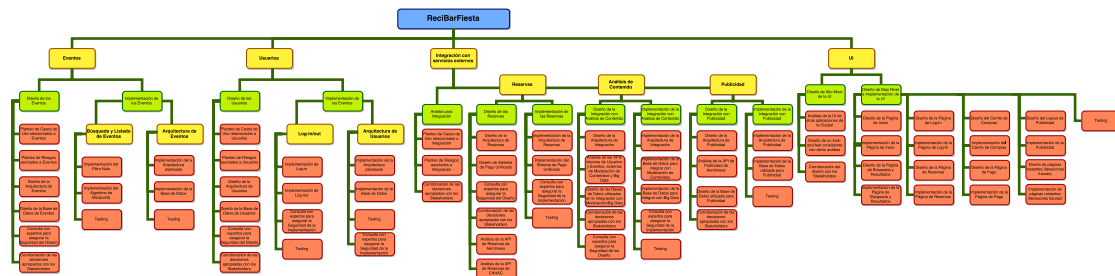


Figura 1: WBS

En la figura 1 puede observarse el diagrama del WBS. Los nodos amarillos representan productos; los verdes, procesos; los rojos, tareas, independientemente de que sean productos o procesos. Cabe realizar algunas aclaraciones respecto del diagrama. El diseño de la arquitectura del sistema se realizará para la próxima entrega; algunas de las subdivisiones en tareas se basan en dicho diseño, por lo que se observan múltiples instancias de tareas placeholder de la forma “Implementación de la Arquitectura de X”. Tras realizar dicha arquitectura, éstas se deberían separar en las tareas más granulares a realizar. Es decir, en el WBS nos limitamos a presentar una visión de alto nivel de las subdivisiones del proyecto, que serían refinadas al proceder con el proceso de diseño. El enfoque más específico se reserva para los productos, procesos y tareas correspondientes a los Casos de Uso que fueron designados a la primera iteración.

4.2. Primera iteración

Para reflejar el orden en que llevaremos a cabo la primera iteración realizamos un diagrama de Gantt (figura 2) el cual modela, entre otras cosas, las dependencias, duraciones y fechas de inicio de cada tarea; sirviendonos de guía al llevar a cabo el proyecto y para identificar puntos delicados en nuestro plan, utilizando criterios como el de camino crítico.

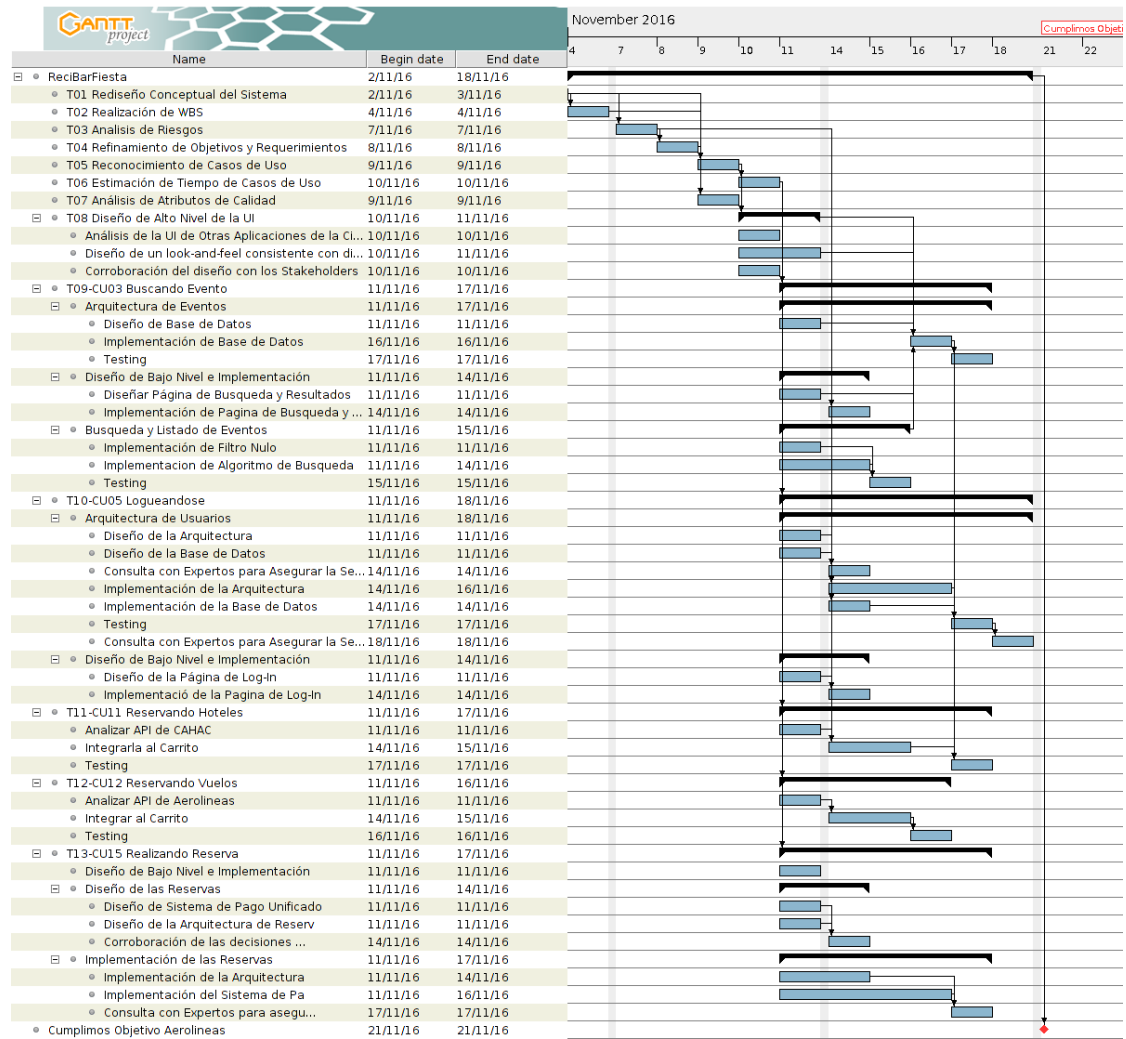


Figura 2: Diagrama de Gantt para la primer iteración del proyecto

4.2.1. Justificación

Como vimos antes, deberíamos priorizar tener la integración con Aerolíneas Argentinas terminada lo más rápido posible, de tal manera de mitigar el riesgo de no tenerla terminada para 2017. Nótese que es el único riesgo que encontramos al cual tenemos exposición alta. De esta manera, pusimos en la primera iteración todo lo relacionado con la compra de pasajes de avión y reserva de hoteles. Esta es la parte más delicada de la integración. El cumplimiento de este objetivo se puede ver reflejado como un hito en nuestro diagrama de Gantt

Además, incluimos los otros dos casos de uso a la iteración porque son fundamentales para toda la aplicación en general, y para la integración con Aerolíneas en particular.

4.3. Resto de las iteraciones

En las siguientes iteraciones deberíamos ir completando el resto de las funcionalidades, prestando atención a implementar los mecanismos de mitigación de los riesgos de exposición media (y bajo, pero más adelante), además de sus planes de contingencia, siempre en caso de que sean por software.

En la Segunda iteración, deberíamos finalizar la integración con Aerolíneas que comenzamos en la primera iteración. Esto se basaría en diseñar y desarrollar la integración de las publicidades de Aerolíneas en nuestra aplicación. Esto no fue incluido en la primera iteración porque el equipo no hubiera dado a basto a completar el trabajo, que hubiera requerido demasiadas horas hombre. Recordemos que está el riesgo de que algún ingeniero clave renuncie. También se agregan funcionalidades como el filtrado de resultados y visualización de caminos para lograr tener un entregable que ponga a prueba algunos de los objetivos principales de la aplicación. En esta iteración dejamos algo de tiempo dedicado al rediseño y reimplementación de la aplicación luego de escuchar el feedback del entregable.

La Tercera y última iteración de Elaboración comprende el diseño e implementación de las interacciones entre los eventos y los usuarios. La creación, edición y eliminación de eventos, rankings y demás.

La Cuarta iteración es de construcción, se espera dedicar gran parte del tiempo a implementar, testear y realizar deployments. Los casos de uso de esta iteración requieren bastante la interacción con agentes externos, ya que entre ellos están la capacidad de compartir información en redes sociales, verificar contenido inapropiado automática y manualmente. También se realiza la obtención de estadísticas de los usuarios y posterior manejo de su envío a las Empresas de Big Data.

4.3.1. Casos de uso por iteración

Primera Iteración (Elaboración)

- CU03 (56 Horas Hombre)
- CU05 (59 Horas Hombre)
- CU11 (56 Horas Hombre)
- CU12 (48 Horas Hombre)
- CU15 (56 Horas Hombre)

Duración: Dos semanas y tres días

Segunda Iteración (Elaboración)

- CU13 (20 Horas Hombre)
- CU14 (48 Horas Hombre)
- CU16 (20 Horas Hombre)

- CU08 (20 Horas Hombre)

Duración: Dos semanas y tres días

Tercera Iteración (Elaboración)

- CU18 (32 Horas Hombre)
- CU04 (60 Horas Hombre)
- CU07 (8 Horas Hombre)
- CU01 (30 Horas Hombre)
- CU02 (8 Horas Hombre)

Duración: Dos semanas y tres días

Cuarta Iteración (Construcción)

- CU06 (8 Horas Hombre)
- CU09 (48 Horas Hombre)
- CU10 (8 Horas Hombre)
- CU17 (32 Horas Hombre)

Duración: Tres semanas

5. Especificación de Escenarios

5.1. Escenarios de Seguridad

5.2. Escenarios de Disponibilidad

5.3. Escenarios de Modificabilidad

5.4. Escenarios de Performance

6. Arquitectura

6.1. Diagramas de Componentes y Conectores con Deployment

6.1.1. Nivel 0

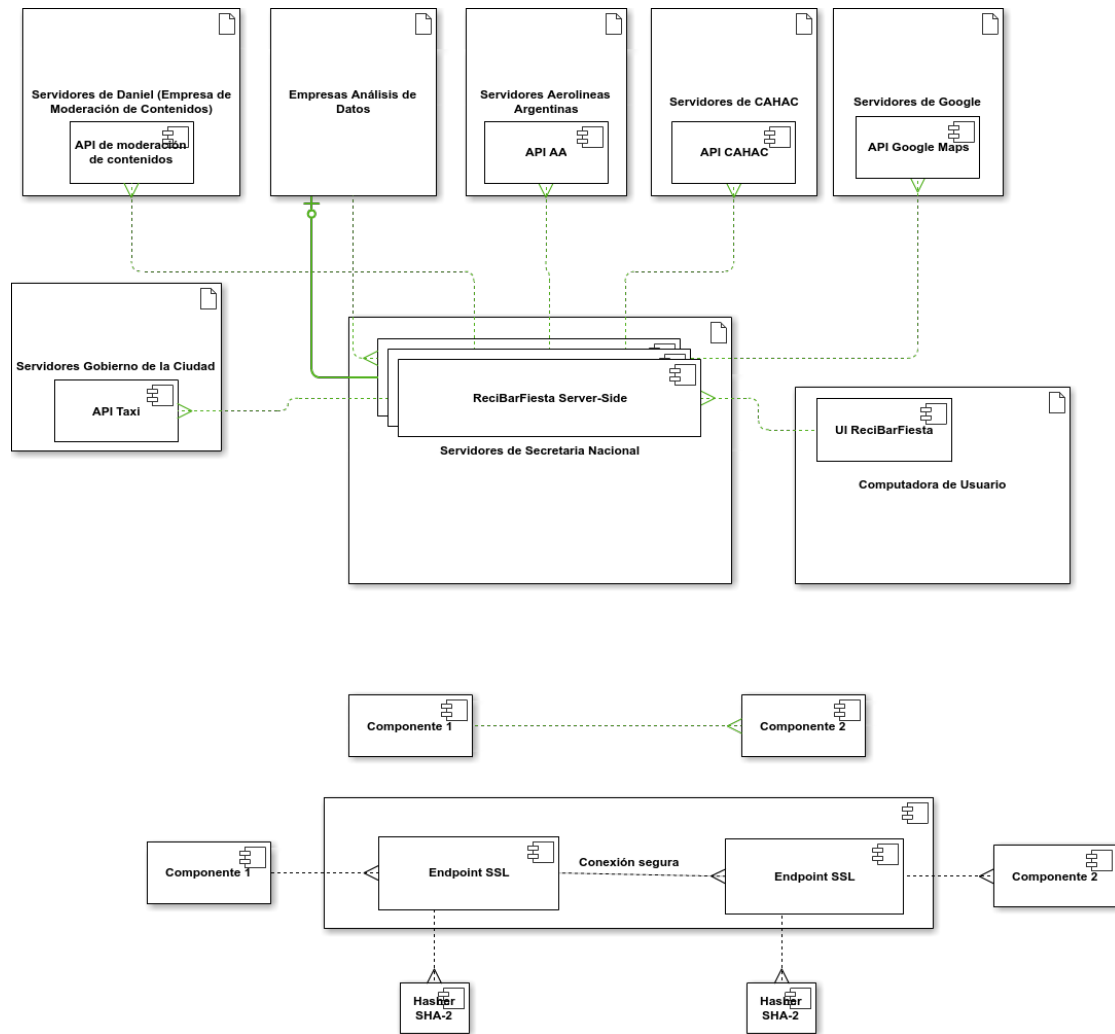


Figura 3:

6.1.2. ReciBarFiesta

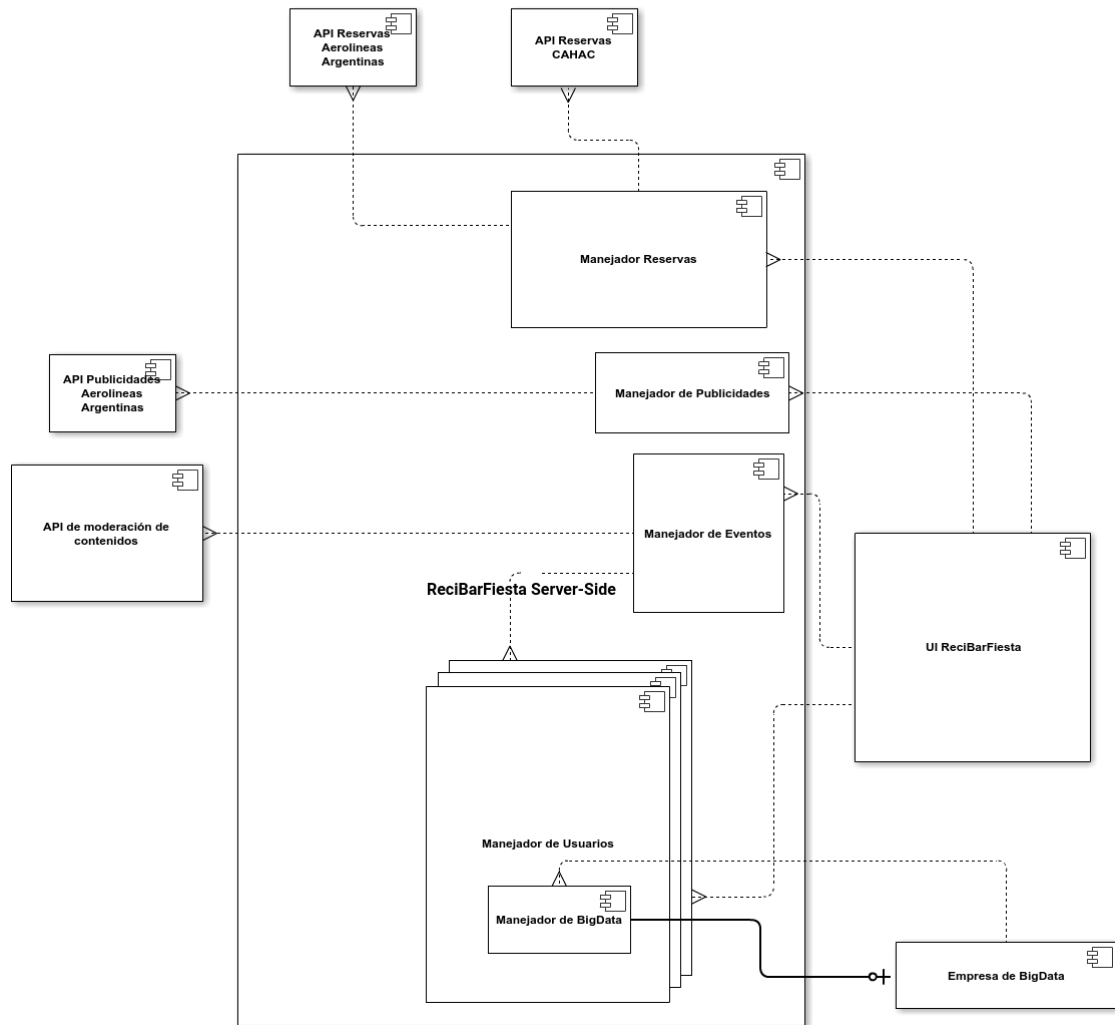


Figura 4:

6.1.3. Manejador de Eventos**6.1.4. Manejador de Usuarios****6.1.5. Manejador de Reservas****6.1.6. Manejador de Publicidades****6.2. Explicación de la Arquitectura****6.2.1. Manejador de Eventos**

El manejador de eventos es el encargado de responder a las distintos requests que envía un usuario de la aplicación desde la interfaz para ejecutar una acción sobre un evento: agregar uno nuevo, modificar, eliminar, valorar, visualizar y búsqueda de eventos.

Notar que los componentes de Visualización, Valoración y Búsqueda de Eventos están replicados para poder realizar muchas operaciones concurrentemente. Sin embargo, el componente de ABM no está replicado para evitar que se generen inconsistencias en la información de los eventos.

Visualización de Eventos

Es el encargado de obtener la información cargada en las bases de datos asociada a un evento en particular, que incluye los datos básicos del evento (así llamaremos a datos como el Nombre, Ubicación, Horario, Tipo de evento, etc.) y los datos del "Perfil del Evento" (que incluye valoraciones por cada feature según el tipo de evento y comentarios asociados a esas valoraciones realizados por los usuarios).

Supongamos que un usuario quiere acceder a la Vista de un evento en particular, al hacer click en un evento el sistema envía el ID del mismo al Manejador, en particular al componente Driver Visualizador de Eventos. Este componente recibe el request y procede a buscar la información asociada al evento (perfil y datos básicos). Para ambos casos, hay un componente encargado de realizar las consultas en la base correspondiente. Además, cada uno tiene una caché asociada con los últimos eventos visitados por los usuarios. Una vez obtenida la información, en caso de que el usuario desee ver el recorrido hasta el evento, se envía al generador de recorridos la dirección actual del usuario y la dirección del evento. Este componente se encargará de interactuar con la API necesaria: GoogleMaps en caso de que sea recorrido a pie, en auto o en colectivo, o la API de Taxis de la Ciudad, para recorridos en taxi. A través de dichas APIs generará un código HTML que se enviará al Comunicador. Luego, el Comunicador del Driver de Visualización envía toda la información al Renderer HTML para generar la página correspondiente y enviársela a la interfaz para que el usuario pueda verla.

Alta/Baja/Modificación de Eventos

Desde la interfaz, un usuario decide realizar una de las operaciones de ABM para algún evento en particular. Primero, el usuario debe ser autorizado por el Autorizador de Usuarios si es que tiene los permisos necesarios para realizar la operación. Una vez chequeados sus permisos, el request llega al Driver ABM de Eventos del Manejador. El Manejador de operaciones deriva el request parseado con los datos necesarios para realizar la operación al componente correspondiente: Carga de nuevos Eventos, Eliminación, o Modificación. En caso de tratarse de Carga de un Evento, el componente chequea que el evento sea válido, y en caso afirmativo pro-

cede a agregarlo a la base de datos de Información básica y a la base de Perfiles, generando un perfil vacío. En caso de que sea una baja de un evento, el componente elimina todas las entradas del mismo en las bases de perfiles e información básica, y luego envía un request asincrónico al Sincronizador de Cambios en Cachés, encargado de actualizar todas las cachés de los componentes del Manejador de Eventos, para que no queden con datos inconsistentes. Por último, en caso de ser una modificación, se actualiza la información básica del evento una vez que haya sido validada, y luego se avisa al sincronizador de cachés para que ejecute las actualizaciones. Una vez finalizada la operación, la misma se guarda en el historial de acciones por usuario.

Valorador de Eventos

Este componente es el encargado de almacenar las valoraciones de los usuarios en el perfil del evento correspondiente. Una valoración debe incluir al menos un valor seleccionable para los features de un evento, que dependen del tipo del mismo, y además puede o no incluir un comentario asociado a dicha valoración. En caso de contener un comentario, éste se encolará para que sea validado interactuando con la API de moderación de contenidos, y luego guardado por el persistidor de Comentarios. En caso de que la conexión con la API se vuelva lenta, se guardará por un lado la valoración y el comentario asociado quedará encolado hasta que sea validado, y luego almacenado. En caso de ocurrir esto se avisará al usuario con mensaje al retornar la respuesta. Si el comentario no es apropiado, se devolverá un mensaje de error. Por otro lado, al persistir las valoraciones, éstas son guardadas temporalmente en un repositorio de últimas valoraciones efectuadas. Este repositorio es leído continuamente por un componente encargado de detectar si se están produciendo demasiadas valoraciones similares al mismo evento; y, en caso de detectarlo, enviará un mail a través del Mail Sender para que la situación sea revisada manualmente.

Buscador de Eventos

El Driver Buscador de Eventos manejará las búsquedas realizadas por los usuarios, dividiéndolas en búsquedas por Features o búsquedas por Información Básica del evento. El Manejador de Búsquedas parseará el request del usuario y enviará la búsqueda al componente Buscador correspondiente según el tipo de la misma. Cada uno de estos componentes cuenta con una Caché donde se almacenarán las búsquedas más populares realizadas por los usuarios, que serán actualizadas periódicamente. Las cachés cuentan con tres niveles (primario, secundario, y terciario) según el nivel de popularidad de las búsquedas para garantizar performance. Una vez realizada la búsqueda, en caso de ser requerido por el usuario, el manejador enviará los resultados al Componente buscador de Eventos Combinables, para que genere búsquedas de eventos combinables (ceranos, temáticas relacionadas, etc.), que luego serán efectuadas por el Buscador correspondiente según se trate de búsquedas en la base de datos de perfiles o de información básica. Una vez obtenidos los resultados, se actualizan las cachés, se notifica al historial de usuario la acción realizada, y se devuelve una página con los resultados y links a las vistas de los eventos encontrados.

7. Referencias