2. Plan del proyecto

En este capítulo se definen los planes que se han realizado para llevar a cabo este trabajo. Los planes se han llevado a cabo en colaboración con las partes interesadas que participan durante el desarrollo de este trabajo y que utilizarán la modificación de reservas una vez que se despliegue en producción.

A lo largo de los apartados de este capítulo se exponen los planes de:

Alcance: donde se define qué ha de realizarse en el proyecto y qué queda fuera de él. Se aporta una visión general del proyecto y se define la lista de entregables.

Requisitos: en el que se detallará el conjunto de requisitos no funcionales y funcionales del proyecto.

Criterios de aceptación: políticas y criterios de las dos partes interesadas externas para aceptar el proyecto desarrollado por TravelgateX.

Restricciones del proyecto: se definen las dos circunstancias más restrictivas que van a condicionar la planificación y posterior desarrollo del trabajo.

Organización: sirve para dar un vistazo a la cultura y forma de trabajar de los empleados de TravelgateX encargados del desarrollo de integraciones.

Partes interesadas: se enumeran a los implicados en el proyecto y se explica qué relación existe entre ellos.

Riesgos: se detallan los dos mayores riesgos del proyecto y el plan de gestión elaborado para controlarlos.

Tiempo: se detalla la duración de las tareas del proyecto y se exponen todas ellas en un diagrama.

Comunicaciones: se definen cuáles serán los canales y la forma de comunicación a utilizar entre los diferentes interesados externos e internos.

2.1 Alcance del proyecto

La motivación de este proyecto surge de la necesidad de uno de los clientes de TravelgateX, Logitravel, de poder modificar las reservas realizadas con Amadeus. Además, la relación de negocio entre Logitravel y Amadeus, ya consolidada desde hace años, provoca que muchos de los proyectos que Amadeus saca adelante, sean una oportunidad para sus mejores clientes para mejorar y perfeccionar sus relaciones de venta. De esta forma, cuando Amadeus desarrolla un proyecto, los primeros en saberlo e interesarse en él, son las agencias y en este caso, también TravelgateX.

En la planificación establecida por Amadeus y en la que participaron agentes de negocio de Logitravel y el desarrollador encargado de los GDS’s en TravelgateX, se definieron las funcionalidades existentes que puede aportar Amadeus para la necesidad de modificar reservas. En la reunión se destacaron cuales iban a ser las fases del desarrollo y las funcionalidades que deberían desarrollarse para cumplir los objetivos de la agencia. También se enumeraron y explicaron el resto de funcionalidades que podían introducirse en el proyecto si así se consideraba. A continuación se detalla cuál es el alcance y el resto de funcionalidades que en un principio quedaron fuera de éste.

Se define que un usuario ha de poder modificar el itinerario reservado mediante la elección de una opción entre las de una lista de disponibles. Para ello se expusieron dos alternativas: modificar el itinerario sabiendo la nueva fecha para la que reservar, o sin saberla. Estas dos opciones son compatibles y pueden confluir, sin embargo han de desarrollarse ambas por separado y con webservices distintos. Se estableció que la primera opción entraba en el alcance del proyecto y la segunda fuera.

Se estableció que para toda compañía aérea que permita la modificación del itinerario, la funcionalidad a desarrollar debería poder ser capaz de la modificación fuera posible. De esta forma, si una serie de aerolíneas necesitan un flujo de transacciones concretas para la modificación y otras, otro flujo, la aplicación ha de poder adaptarse a ambas y completar una modificación con éxito.

Para certificar el correcto y adecuado funcionamiento del desarrollo, Amadeus impone una serie de pruebas predefinidas en su plan de proyecto, que deberá realizar el desarrollador una vez termine la construcción de la aplicación.

Quedó fuera del propósito de la reunión de planificación, las herramientas o funcionalidades extra a desarrollar que son necesarias para llevar a cabo el proyecto. Por lo tanto, en la reunión no se definió la parte del desarrollo de cara al cliente Logitravel.

Mediante la colaboración entre Logitravel y TravelgateX se definieron los activos a desarrollar: nuevas llamadas en la API de transportes para poder modificar una reserva; código que integre los dos flujos (transacciones con Amadeus y transacciones con Logitravel). Una vez desplegado el desarrollo en producción, tal y como se define en el capítulo 7, se considera por terminado este proyecto, y por lo tanto el mantenimiento de la integración queda fuera del alcance de este proyecto.

Se incluye dentro del alcance establecido solamente por TravelgateX la creación de un formulario de pruebas que agilice el desarrollo y la certificación de Amadeus. Han de realizarse también las gestiones necesarias para permitir la visualización de estadísticas de reservas modificadas por los clientes una vez que el desarrollo esté funcionando en producción.

La planificación del alcance implica entonces la realización de los siguientes entregables:

Tabla 14: Listado de entregables definidos en el alcance del plan de proyecto

|  |  |
| --- | --- |
| Amadeus | Escenarios de test realizados para una correcta certificación por parte del GDS Amadeus |
| Diagramas de flujo de la funcionalidad |
| Logitravel | Integración de Amadeus con la funcionalidad modificar reservas |
| Nuevas llamadas en la API de transportes |
| TravelgateX | Formulario de pruebas |
| Documentación de los nuevos elementos incluidos en la API de transportes |
| Visualización de estadísticas en la web interna de TravelgateX |

2.2 Requisitos del proyecto

En este capítulo se definen los requisitos funcionales y no funcionales del proyecto. En el apartado 2.2.2.1 se detalla la tecnología que se usa para la construcción de las transacciones de la API, la tecnología para desarrollar la integración en 2.2.1.2 y la tecnología para construir el formulario de pruebas en 2.2.1.3. Finalmente se detallan los requisitos funcionales en el apartado 2.2.2.

2.2.1 Requisitos no funcionales

En este capítulo se han separado los requisitos no funcionales en las tres grandes áreas que separan el proyecto:

Transacciones de la API de transportes

Integración

Formulario de pruebas

2.2.1.1 Tecnología usada para las transacciones de la API de transportes

Las peticiones y respuestas, así como elementos y atributos que las forman deberán ser desarrolladas en el lenguaje de programación Visual Basic.

Las nuevas llamadas implementadas han de permitir la compatibilidad entre el lenguaje de programación y la creación de objetos XML.

Las nuevas llamadas han de permitir la creación en última instancia de archivos .wsdl y .xsd, de tal forma que el sistema de TravelgateX pueda interpretar el contenido creado.

Las llamadas han de ser compatibles con el resto de contenido ya existente en la API de transportes.

Todos los elementos y atributos creados deberán ser correctamente explicados en el código mediante comentarios que los definan.

La nomenclatura de las llamadas y su contenido deberá seguir acorde con el estilo de la API de transportes.

Para permitir la serialización y deserialización de objetos a xml’s, los elementos creados deberán ser del tipo XmlElement y los atributos del tipo XmlAttribute.

Todas las clases creadas en la API deberán ser públicas.

Todas las clases creadas en la API deberán heredar de las clases básicas de la API, Transportation y estas a su vez de Base, conteniendo ya éstas elementos generales.

Las clases deberán ser desarrolladas con un constructor que permita la declaración de objetos vacíos, así como objetos que inicialicen todas sus variables.

Las clases, elementos y atributos que vayan a ser visibles en las transacciones XML han de permitir su visualización en español y en inglés.

2.2.1.2 Tecnología usada para la construcción de la integración

Las nuevas clases en la integración para la modificación de reservas deberán desarrollarse mediante el lenguaje de programación Visual Basic.

La integración ha de controlar las excepciones que se produzcan durante la ejecución.

La integración ha de controlar los errores que devuelva cualquier transacción realizada con el proveedor y tipificar adecuadamente dichos errores para describir su causa.

La integración ha de controlar los errores de servidor y de cliente que se produzcan durante la ejecución de las transacciones.

La integración ha de controlar el tiempo transcurrido desde el comienzo de la transacción hasta su finalización y cortar la ejecución en el tiempo previamente estipulado para dicha transacción.

La integración ha de registrar en la respuesta de la transacción de cliente (ej: DisponibilidadRS) todas las transacciones que se hayan realizado con el proveedor.

La integración ha de abrir una sesión con las credenciales adecuadas y predeterminadas con el proveedor y mantener dicha sesión para todas las transacciones que se realicen, así como cerrar la sesión una vez terminado el flujo de transacciones deseado.

El código desarrollado ha de mantener la compatibilidad con el resto de funcionalidades ya desarrolladas en la integración.

El código desarrollado ha de aprovechar al máximo las funcionalidades desarrolladas en la integración previamente para facilitar su posterior mantenimiento y legibilidad del código.

La integración ha de minimizar el uso de memoria y procesador durante su ejecución, de tal forma que la ejecución de las nuevas llamadas ha de demostrar un uso medio igual al resto de transacciones ya desarrolladas en la integración.

El código desarrollado debe escribirse en inglés para permitir su legibilidad para el resto de integrantes de TravelgateX.

La integración deberá incluir entre un 5 y un 10% de comentarios respecto a la totalidad de nuevo código desarrollado.

La integración debe seguir la estructura estandarizada de transacciones que sigue el equipo de desarrollo de integraciones de TravelgateX: construcción de petición, consulta de tiempo transcurrido, realización de llamada al proveedor, registro de errores durante la llamada y registro de transacción. Finalmente pueden tratarse los datos recibidos en la respuesta de la transacción.

La programación realizada deberá facilitar la compatibilidad con la creación y ejecución de pruebas sobre el proyecto.

2.2.1.3 Tecnología usada para la construcción del formulario de pruebas

El formulario de pruebas deberá desarrollarse mediante el lenguaje de programación Visual Basic.

El formulario ha de permitir el manejo de transacciones (petición respuesta) de forma completamente visual.

El formulario de pruebas ha de controlar y visualizar cualquier error de ejecución arrojado por la integración o durante la ejecución de código del mismo formulario.

Los resultados obtenidos en las transacciones de disponibilidad han de poder visualizarse de manera gráfica en una tabla de opciones.

El formulario no ha de perjudicar ni alterar los resultados obtenidos en ninguna de las transacciones realizadas en él.

El formulario ha de ser compatible con el resto de integraciones de vuelos que existen en TravelgateX.

El formulario no debe usar terminología o tipología del proveedor Amadeus.

El formulario de pruebas ha de poder abrirse mediante un ejecutable.

2.2.2 Requisitos funcionales

En este capítulo se enumeran los requisitos funcionales distinguiendo aquellos requisitos que deberá cumplir la integración y aquellos que deberá cumplir el formulario de pruebas.

En los requisitos funcionales de la integración, se separan aquellos que deberán cumplirse para toda nueva transacción desarrollada, de los requisitos que debe cumplir cada una de las transacciones por separado.

Probablemente todos los requisitos funcionales los acabe metiendo dentro de una tabla, con id’s para cada uno de los requisitos (también para los no funcionales), la tabla sería:

Tabla 15: Requisitos funcionales

Requisitos funcionales de la integración.

En todas las transacciones:

El cliente introduce el tiempo de corte de la transacción.

El cliente introduce las credenciales y las url’s a las que atacará la transacción.

El cliente elige el idioma de las transacciones xml.

El cliente visualiza los errores de la transacción.

El cliente visualiza los warnings de la transacción.

El cliente visualiza las transacciones del proveedor (petición y respuesta).

El cliente visualiza el tiempo transcurrido en la ejecución de la transacción.

El cliente visualiza la fecha en la que se realiza la transacción.

En la transacción DMR:

El cliente indica los tramos del nuevo itinerario.

El cliente indica el tipo de modificación de un tramo.

El cliente indica el origen, destino y fechas de un tramo.

El cliente indica el localizador de la reserva.

El cliente filtra la búsqueda por clase cabina.

El cliente filtra la búsqueda por solo vuelos directos (sin escalas).

El cliente filtra la búsqueda por compañía.

El cliente elige si la búsqueda incluye compañías lowcost.

El cliente elige el tipo de viaje que desea realizar: RT, OW, OJ, CT.

El cliente visualiza los segmentos arrojados en la búsqueda.

El cliente visualiza el número de transporte del segmento.

El cliente visualiza el origen, destino y fecha del segmento.

El cliente visualiza la compañía que vende el vuelo y la compañía que opera el vuelo.

El cliente visualiza las terminales de salida y llegada del segmento.

El cliente visualiza el tipo de transporte (tipo de avión).

El cliente visualiza la duración del segmento.

El cliente visualiza si el segmento tiene parada técnica y sus detalles.

El cliente visualiza el tipo de tarifa: RT, OW, OJ, CT.

El cliente visualiza las condiciones de la tarifa.

El cliente visualiza el importe de la tarifa desglosada por tipo de pasajero: ADT, CHD e INF.

El cliente visualiza la moneda del importe.

El cliente visualiza las tasas del importe y su tipo.

El cliente visualiza los cargos o penalizaciones que deberán abonarse por la modificación de la reserva.

El cliente visualiza los segmentos asociados a cada tarifa.

El cliente visualiza la compañía validadora de la tarifa.

El cliente visualiza la familia de tarifa asociada.

El cliente visualiza el tipo y cantidad de equipajes incluidos en la tarifa.

El cliente visualiza los elementos extra que incluye la tarifa.

El cliente visualiza la clase cabina de cada segmento.

El cliente visualiza la clase de cada segmento.

El cliente visualiza el número de asientos disponibles por cada clase.

El cliente visualiza el tipo de tarifa: pública, privada o negociada.

El cliente visualiza los billetes de la reserva.

El cliente visualiza la cantidad de pasajeros desglosados por tipo de pasajero: ADT, CHD e INF.

En la transacción RMR:

El cliente indica la tarifa seleccionada.

El cliente indica la forma de pago.

El cliente indica los billetes de la reserva.

El cliente indica el localizador de la reserva.

El cliente indica el delta price de la reserva.

El cliente visualiza los datos de un pasajero guardados en la reserva.

El cliente visualiza los billetes de la reserva.

El cliente visualiza la factura de la reserva una vez ha sido ya modificada.

El cliente visualiza los importes desglosados de la factura, por tipo de pasajero y los cargos de la misma.

El cliente visualiza la compañía que cobra la factura.

El cliente visualiza el last ticketing date.

El cliente visualiza el tipo de emisión.

En la transacción EMR:

El cliente indica el tipo de emisión.

El cliente indica los billetes de la reserva.

El cliente indica el localizador de la reserva.

El cliente visualiza los nuevos billetes de la reserva.

Requisitos funcionales del formulario de pruebas.

En todas las transacciones:

El usuario copia la petición y la respuesta xml.

El usuario indica el tiempo de corte de la transacción.

El usuario indica las credenciales y las url’s.

El usuario elige el entorno del proveedor al que se realizará la transacción: test o producción.

El usuario indica el tipo de transacción y pega la petición que posteriormente podrá realizar.

En la transacción de disponibilidad:

El usuario indica el origen, destino y fechas de cada tramo.

El usuario añade una compañía de conexión.

El usuario elimina una compañía de conexión.

El usuario indica el número y la edad de los pasajeros.

El usuario realiza la petición de Disponibilidad.

El usuario elige el tipo de viaje: OW, RT o OJ

En la transacción de Valoración:

El usuario elige las preferencias de Valoración.

El usuario elige las opciones devueltas por la Disponibilidad en una tabla.

El usuario realiza la petición de Valoración.

En la transacción de Reserva:

El usuario indica el nombre, apellidos, email, teléfono, calle, localidad, C.P., país y nacionalidad del cliente titular de la reserva.

El usuario indica el tratamiento de un pasajero y su sexo.

El usuario indica el nombre, apellidos, fecha de nacimiento, tipo, id y fecha de caducidad del documento de identidad y C.P. del municipio en el que reside.

El usuario indica la forma de pago: card o cash.

El usuario indica el titular, número, tipo, CVC y fecha de caducidad de la tarjeta.

El usuario indica el delta Price.

El usuario realiza la petición de Reserva.

En las transacciones de Recuperar, Cancelar, Emitir y Vaciar billete:

El usuario indica el localizador de la reserva.

El usuario indica el número de un billete.

El usuario indica el tipo de emisión.

El usuario indica el tipo de billete: normal o extra.

El usuario realiza las peticiones Recuperar Reserva, Cancelar Reserva, Emitir billetes y Vaciar Billete.

En las transacciones de Modificar Reserva:

El usuario indica el localizador de la reserva.

El usuario elige el tipo de viaje: OW, RT o OJ

El usuario indica el origen, destino y fechas de cada tramo.

El usuario indica la acción a realizar en cada tramo: N, KF, K, C, R o A.

El usuario indica el titular, número, tipo, CVC y fecha de caducidad de la tarjeta.

El usuario elige las opciones devueltas por DMR en una tabla.

El usuario realiza las peticiones DMR, RMR y EMR.

2.3 Criterios de aceptación

En este capítulo se exponen los criterios de aceptación establecidos en el plan, definidos y acordados entre el desarrollador y el proveedor Amadeus 2.3.1 y entre el desarrollador y la agencia Logitravel 2.3.2.

2.3.1 Certificación de Amadeus (criterios de aceptación)

En este capítulo se presenta el checklist de certificación por parte del GDS Amadeus, que deberá cumplir la aplicación desarrollada. En cada uno de los criterios presentados en las tablas *Tabla 16*, *Tabla 17* y *Tabla 18*, se indica el webservice al aplica, la definición de dicho criterio y cómo será tratado el déficit del mismo.

Tabla 16: Criterios de aceptación generales

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Web Service (s) | Criterio | Error o warning |
| GEN-01 | Todos | Los servicios xml implementados no deben estar deprecados | Error |
| GEN-02 | Todos | La aplicación no debe crear bucles infinitos en el host de Amadeus | Error |
| GEN-03 | Todos | Ha de implementarse un control de errores adecuado | Error |
| GEN-04 | Todos | El formato de los datos y sus variables deben comprobarse antes de enviarlos a Amadeus. | Error |
| GEN-05 | Todos | Deben evitarse transacciones innecesarias o redundantes | Warning |
| GEN-06 | Todos | Deben usarse mensajes estructurados cuando sea posible | Warning |
| GEN-07 | Todos | No deben crearse reservas duplicadas | Error |

Tabla 17: Criterios de aceptación de gestión de sesión

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Web Service (s) | Criterio | Error o warning |
| SEM-01 | Todos | Todas las sesiones deben cerrarse adecuadamente en la aplicación | Error |
| SEM-02 | Todos | Los tiempos de inactividad o respuesta deben ser tratados por la aplicación | Error |
| SEM-03 | Todos | La secuencia de números de sesión debe ser incrementada y adecuadamente insertada en el encabezamiento SOAP de la petición: 1.0, 1.1, 2.0, 2.1 | Error |

Tabla 18: Criterios de aceptación ATC

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Web Service (s) | Criterio | Error o warning |
| ATC-01 | Todos los servicios ATC | La aplicación debe proveer un ticket por pasajero (incluidos bebés) | Error |
| ATC-02 | ATC Shopper | Las recomendaciones (tarifas) devueltas por ATC Shopper deben visualizarse correctamente en la aplicación | Error |
| ATC-03 | ATC Shopper | El Web Service Air\_SellFromRecommendation debe usarse solamente sobre tarifas devueltas por el ATC Shopper | Error |
| ATC-04 | ATC Shopper | La aplicación debe tratar correctamente los tipos de pasajero que se envíen en las peticiones a Amadeus | Error |
| ATC-05 | ATC Shopper | La aplicación debe devolver la lista de aerolíneas que se recibirá por parte de Amadeus en ATC Eligibility | Error |
| ATC-06 | ATC Shopper | La aplicación debe recuperar la reserva si esta se ha hecho a través de Amadeus y eliminar los segmentos originales una vez que estos se modifiquen | Warning |
| ATC-07 | ATC Shopper | ATC Eligibility debe ser sistemáticamente utilizado antes de cualquier servicio adicional de ATC Shopper | Warning |

2.3.2 Criterios de aceptación de la agencia

En este capítulo se presentan los criterios que se acordaron entre la agencia y el desarrollador de TravelgateX para aceptar la aplicación desarrollada. No se requiere ningún tipo de certificación para aceptar o no la aplicación desarrollada, como así ocurre con el GDS. Aun así, se espera que los criterios enumerados a continuación se cumplan dentro de unos límites aceptables (dentro de los márgenes que se han ido cumpliendo en todas las aplicaciones desarrolladas para la agencia).

* Control de errores y warnings siguiendo los tipos que ya se utilizan en otras transacciones.
* Control del tiempo de corte.
* Visualización de los resultados obtenidos por parte del proveedor, tratados y sin tratar.
* Visualización de la fecha en la que se ha realizado una petición.

2.4 Restricciones del proyecto

En este capítulo se detallan las restricciones del proyecto. Primero se detallan las restricciones de tiempo establecidas por el GDS en el apartado 2.4.1 y después, en el apartado 2.4.2, los factores ambientales.

2.4.1 Restricción de tiempo

En este apartado se muestra el plan de proyecto establecido por Amadeus que restringe la duración del proyecto desarrollado por TravelgateX. Se indican los periodos de duración del proyecto y los periodos de soporte técnico preestablecidos por el GDS.

Figura 16: (figura project life cycle amadeus)

2.4.2 Factores ambientales: cultura del Dpto. de integraciones

TravelgateX ha desarrollado y mantiene una gran cantidad de integraciones. Esto supone el mantenimiento y actualización de muchos servicios desarrollados por ingenieros que ya no trabajan en el Dpto. de integraciones. De esta forma, es esencial que dichos desarrollos cumplan con una serie de criterios, que ayuden a su posterior mantenimiento.

Entre los requisitos más destacados que el Dpto. de integraciones propone, se encuentran:

* Desarrollar en base a funcionalidades compartidas entre APIs.
* Desarrollar utilizando el vocabulario inglés.
* La integración debe seguir una estructura estandarizada de transacciones: construcción de petición, consulta de tiempo transcurrido, realización de llamada al proveedor, registro de errores durante la llamada y registro de transacción. Finalmente pueden tratarse los datos recibidos en la respuesta de la transacción.
* Desarrollar toda nueva integración en el lenguaje de programación C#.
* Compartir con el Dpto. de integraciones cualquier funcionalidad que se considere útil para toda integración desarrollada.
* Utilizar nomenclatura compartida entre integraciones.
* Realización de code reviews de manera quincenal en los que se persigue:
  + Mejorar la calidad del código analizado.
  + Servir como herramienta a los desarrolladores para aprender cuándo y cómo aplicar técnicas de calidad, consistencia y mantenibilidad.

2.5 Organización

En el apartado 2.5.1 de este capítulo, se verán algunas de las consignas más destacadas de la cultura y estilo de TravelgateX, que no han sido mencionadas anteriormente en el apartado 2.4.2. Finalmente, en el apartado 2.5.2 se indica quien participa internamente en este proyecto.

2.5.1 Cultura y estilo de TravelgateX

Además de los requisitos de desarrollo mencionados en 2.4.2, TravelgateX cuando con una serie de consignas que los empleados tienden a seguir, mejorando así la colaboración entre equipos de desarrollo.

* Documentación de cualquier cambio realizado sobre las APIs.
* Crear y compartir posts de ayuda, solucionadores de problemas, FAQs, notas de reunión, etc. en la página web corporativa (interna).
* De forma semanal, realizar retrospectivas en las que participen todos los integrantes de un Dpto.
* Utilización (en el Dpto. de integraciones) de la metodología scrum para proyectos de larga duración.
* Utilización de la metodología kanban para la gestión de incidencias.
* Realizar semanalmente, reuniones de seguimiento con el cliente.

2.5.2 Participantes internos del proyecto

El proyecto de modificación de reservas a través del servicio ATC Shopper de Amadeus cuenta con un único desarrollador, integrante del Dpto. de integraciones aéreas de TravelgateX. Él será quien planifique, analice, diseñe y desarrolle todas las fases de este proyecto.

2.6 Partes interesadas

En este capítulo se enumeran los interesados internos y los externos, así como la relación que existe entre ellos. Se ahondará en las capas de interesados más bajas, lo que facilitará al lector comprender cómo se ejecutará la comunicación interna y externa entre los diferentes interesados.

2.6.1 Interesados internos y externos

Los interesados internos de este proyecto son:

Dpto. de integraciones aéreas:

Además del desarrollador encargado del proyecto, el resto de integrantes del mismo Dpto. están implicados en la mejora y mantenimiento de las integraciones aéreas. La complejidad e importancia de este desarrollo, mejorará la calidad y capacidad técnica del Dpto.

Dpto. de integraciones hoteleras y de ancillaries:

El resto de departamentos de integraciones también puede verse enriquecida de cualquier mejora o funcionalidad compartida que surja a partir del desarrollo. Un claro ejemplo es la posibilidad de compartir el formulario de pruebas.

Los interesados externos de este proyecto son:

Amadeus:

El proveedor de los servicios y productos aéreos va a empezar una nueva clase de producto.

Logitravel:

La agencia que se usará la aplicación desarrollada.

Otros clientes:

La incorporación del servicio de modificar reservas puede ser un reclamo para otros clientes que decidan conectarse a TravelgateX.

2.6.2 Relación entre interesados

En la *Figura 19* se representa visualmente la relación ente los interesados internos y externos del proyecto.

Figura 19: (figura relación entre interesados)