

13/03/2025

Grupo: C1.064

Repositorio: <https://github.com/paugonpae67/Acme-ANS>

Autora: Lidia Ning Fernández Casillas (lidfercas@alum.us.es)

ANALYSIS REPORT

ACME ANS – C1.064

TABLA DE CONTENIDOS

N.º	Apartado
1	Resumen ejecutivo
2	Tabla de revisiones
3	Introducción
4	Contenido
5	Conclusión
6	Bibliografía

RESUMEN EJECUTIVO

Este informe presenta un análisis detallado de los requisitos que han requerido una evaluación más profunda durante el desarrollo del entregable 2 del proyecto Acme Ans. No se incluyen todos los requisitos, sino únicamente aquellos que han generado dudas o han necesitado ajustes para su correcta implementación.

Cada registro de análisis contiene la copia literal del requisito evaluado, las conclusiones derivadas del análisis y las decisiones tomadas para corregir o clarificar su alcance. Además, se proporciona un enlace a la validación realizada por el profesor, garantizando que las modificaciones y aclaraciones sean adecuadas y alineadas con los objetivos del proyecto.

Para facilitar este proceso, los registros de análisis se han publicado en el foro de la asignatura, lo que ha permitido recibir retroalimentación adicional del profesor y asegurar que las decisiones adoptadas sean las más acertadas. Gracias a este enfoque, se ha logrado minimizar la ambigüedad en la interpretación de los requisitos, optimizando así la calidad del trabajo final.

TABLA DE REVISIONES

Versión	Fecha	Descripción
1.0	13/03/2025	Versión Inicial. Estructura del documento y Analysis Report

INTRODUCCIÓN

Como bien se mencionó anteriormente en la introducción, este documento presenta un informe detallado sobre el análisis de los requisitos para el entregable 2 dentro del proyecto. Su objetivo es proporcionar una visión clara de los requisitos que han sido evaluados, las conclusiones alcanzadas durante el análisis y las decisiones tomadas para su corrección o clarificación.

El informe se divide en varios registros de análisis, cada uno de los cuales incluye la copia literal del requisito, las conclusiones derivadas del análisis y las decisiones tomadas para ajustarlo. Además, se incorpora un enlace a la validación realizada por el profesor, lo que asegura que las correcciones sean adecuadas y alineadas con los objetivos del proyecto.

Cada registro de análisis se ha publicado en el foro de la asignatura para obtener retroalimentación del profesor, lo que ha facilitado la clarificación de dudas y la optimización de las decisiones tomadas. A través de este informe, se busca documentar y justificar todas las modificaciones realizadas en los requisitos, proporcionando así una base sólida para el desarrollo del entregable final.

Este informe estructuralmente incluye una portada, tabla de contenidos, un resumen ejecutivo, una tabla de revisiones, una introducción, los registros de análisis detallados, una conclusión y la bibliografía usada, que resume los aspectos más relevantes de las correcciones realizadas. Gracias a este enfoque, se ha logrado un análisis exhaustivo y un proceso de mejora continua en la implementación del proyecto.

CONTENIDO

De los requisitos tanto obligatorios como suplementarios I y II, correspondiente a esta entrega 2 se han encontrado ambigüedades en los siguientes:

- **REQUISITO 3**

"Los clientes son quienes compran vuelos. El sistema debe almacenar los siguientes datos sobre ellos: un identificador (único, patrón " $^{\wedge}[A-Z]\{2-3\}d\{6\}$$ ", donde las dos o tres primeras letras corresponden a sus iniciales), un número de teléfono (patrón " $^{\wedge}\backslash+?\d\{6,15\}$$ "), una dirección física (hasta 255 caracteres), además de una ciudad y un país (ambos de hasta 50 caracteres). Opcionalmente, los clientes pueden acumular puntos (hasta 500.000 puntos)."

Conclusiones del análisis del requisito: Tras analizar dicho requisito no queda claramente especificado los límites que hay que establecer para earnedPointr. En el caso de que no se especifique explícitamente si el número debe incluir decimales o cuántas cifras enteras debe tener, no tuve claro si aún así se deben definir los parámetros integer y fraction dentro de ValidNumber, o si estos pueden omitirse. Esta duda no solo aplica a Customer, sino que se repite para todas las entidades que contienen atributos numéricos en el proyecto, por lo que aplica a cualquier otra entidad que tenga algún atributo que deba validarse con ValidNumber

Decisiones: Tras consultar la duda en el foro con el cliente y obtener su respectiva respuesta, en la que se explicaba la no necesidad explícita de incluir los parámetros Integer y Fraction si los límites mínimo y máximo estaban claramente definidos, decidí declarar el atributo como un objeto Integer, ya que este debía poder tomar el valor nulo, dejando únicamente la validación de los valores con los parámetros ValidNumber(min, max).

Link de post en el foro:

[https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id= 89154 1&nav=discussion board&conf id= 426211 1&forum id = 253522 1&message id= 461595 1](https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id=89154_1&nav=discussion_board&conf_id=426211_1&forum_id=253522_1&message_id=461595_1)

- **REQUISITO 4 y 5**

4) *"Una reserva es una reserva realizada por un cliente para comprar un vuelo, garantizando asientos en un itinerario específico y asociando los datos de algunos pasajeros al viaje. El sistema debe gestionar la siguiente información para cada reserva: un código de localizador (único, patrón "[A-Z0-9]{6,8}\$"), el momento de la compra (anterior), la clase de viaje ("ECONOMY", "BUSINESS") y el precio. Opcionalmente, el sistema debe registrar el último pago de la tarjeta de crédito utilizada para el pago."*

5) *"Un pasajero es una persona que toma un vuelo y debe estar registrado en la reserva correspondiente. El sistema debe almacenar los siguientes datos sobre los pasajeros: nombre completo (menos de 256 caracteres), correo electrónico, número de pasaporte (patrón "[A-Z0-9]{6,9}\$"), fecha de nacimiento y, opcionalmente, sus necesidades especiales (menos de 51 caracteres)."*

Conclusiones del análisis del requisito: Tras analizar estos dos requisitos, quedó claro que ambas entidades debían estar relacionadas. Sin embargo, dicha relación era de tipo *many-to-many*, ya que una reserva puede tener muchos pasajeros y un pasajero puede tener compradas muchas reservas. Dicha relación no era posible implementarla en el ámbito de esta asignatura debido a su complejidad. Como resultado, me surgió la duda de cómo implementar las entidades *Booking* y *Customer* sin recurrir a esta relación directa.

Decisiones: Buscando en el foro para ver si algún compañero tenía la misma duda, encontré un post que me proporcionó la respuesta. La solución para implementar esa relación sin utilizar *many-to-many* consistía en crear una entidad intermedia llamada *BookingRecord*, que tendría una relación *many-to-one* tanto con *Booking* como con *Passenger*. No dudé en implementar este consejo, ya que solucionaba el problema de manera eficaz.

Link de post en el foro:

[https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id= 89154 1&conf id= 426211 1&forum id= 253522 1&message i
d= 461394 1&nav=discussion board](https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id=89154_1&conf_id=426211_1&forum_id=253522_1&message_id=461394_1&nav=discussion_board)

- **REQUISITO 16**

"Producir un modelo de dominio UML respecto de los requerimientos de información"

Conclusiones del análisis del requisito: Tras analizar este requisito, surgió la duda en cuanto a cómo debíamos representar las relaciones de las entidades. No quedaba claro si teníamos que integrar todas las entidades (tanto grupales como individuales) en un único modelo de dominio para visualizar mejor las relaciones entre ellas, o si es preferible realizar modelos de dominio separados: uno grupal y varios individuales, cada uno con las entidades específicas de cada estudiante, sin que se relacionasen entre ellos

Decisiones: Tras consultar la duda en el foro con el cliente y obtener su respectiva respuesta, el cliente explicó que la opción A2 era la más adecuada, ya que se basa en asignar un artefacto específico a cada requisito, de manera que cada uno tenga su propia representación individual sin mezclarse con otros, que facilita la auditoría y la evaluación. Debido a ello, se optó por esta opción y se modeló cada estudiante en un fichero diferente, donde el modelo grupal sería independiente a todos ellos.

Link de post en el foro:

[https://ev.us.es/webapps/discussionboard/do/message?action=list_messages
&course_id= 89154_1&nav=discussion board&conf id= 426211_1&forum id
= 253522_1&message id= 460169_1](https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id=89154_1&nav=discussion_board&conf_id=426211_1&forum_id=253522_1&message_id=460169_1)

CONCLUSIÓN

Después de un análisis exhaustivo de los requisitos correspondientes a la segunda entrega, especialmente los requisitos 3, 4, 5 y 16, se identificaron varias ambigüedades que podrían afectar la implementación adecuada del proyecto. Luego de consultar y aclarar estas dudas con el cliente, las expectativas quedaron bien definidas (como se explicó previamente o a través de los enlaces del foro), lo que permitió encontrar soluciones claras y consistentes para cada uno de los requisitos. Esto asegura una alineación más precisa entre las necesidades del cliente y los entregables del proyecto.

En conclusión, el análisis de requisitos es una práctica fundamental para garantizar que el proyecto desarrollado cumpla con las expectativas del cliente, sea funcional y útil en su contexto.

BIBLIOGRAFÍA

Intencionalmente en blanco