



# ANALYSIS REPORT

**D02**

**VIRGINIA MESA PÉREZ**

**ACME-ANS-C1.021**

Repositorio: <https://github.com/DP2-C1-021/Acme-ANS>

Student 2



ÍNDICE

Resumen Ejecutivo.....2

Introducción.....3

Requisitos de Información.....4

Requisitos de prueba.....6

Conclusión.....7



## **Resumen Ejecutivo**

En este documento se examinan el conjunto de requisitos desarrollados en la entrega (D02), con el objetivo de definir correctamente las entidades clave, sus relaciones y la información que debe ser almacenada para garantizar una estructura sólida en la implementación.



## **Introducción**

En este documento analizaremos los requisitos y sacaremos conclusiones de las decisiones tomadas en el desarrollo de estos.

La estructura de este documento consistirá en dos partes, una primera parte de requisitos de información y una segunda parte correspondiente a los requisitos de prueba.



## Requisitos de Información

- **Requisito 03:** “**Customers** are the people who purchase flights. The system must store the following data about them: an identifier (unique, pattern “`^[A-Z]{2-3}\d{6}$`”, where the first two or three letters correspond to their initials), a phone number (pattern “`^\+?\d{6,15}$`”), a physical address (up to 255 characters), plus a city and a country (both up to 50 characters). Optionally, customers may have some earned points (up to 500k points).”

Esta entidad **Customer**, es interpretada como un rol de nuestro sistema, por lo tanto, debería extender **AbstractRole**. El resto de los atributos están explícitamente definidos e implementados.

- **Requisito 04:** “A **booking** is a reservation made by a **customer** to purchase a **flight**, guaranteeing some seats on a specific itinerary and associating some **passengers'** details with the trip. The system must manage the following information for each **booking**: a **locator code** (unique, pattern “`^[A-Z0-9]{6,8}$`”), a **purchase moment** (in the past), a **travel class** (“ECONOMY”, BUSINESS”), and a **price**. Optionally, the system should record the **last nibble** of the credit card used for payment.”

Esta entidad tiene una relación ManyToOne con **Customer** y con **Flight** ya que se especifica que una reserva es una solicitud por un cliente para comprar un vuelo, el resto de los atributos no han presentado problema ya que se especifican sus restricciones.

- **Requisito 05:** “A **passenger** is an individual who takes a flight and he or she must be registered in the corresponding booking. The system must store the following data about passengers: a **full name** (shorter than 256 characters), an **email**, a **passport number** (pattern “`^[A-Z0-9]{6,9}$`”), a **date of birth**, and, optionally, his or her **special needs** (shorter than 51 characters).”

Esta entidad presenta una relación con booking, la cual ha dado lugar a la necesidad de crear una entidad intermedia ya que la relación es del tipo ManyToMany. Dicha entidad se ha llamado **BookingRecord** y contiene el id de booking y el id de passenger con relaciones ManyToOne. Gracias al foro de la asignatura hemos podido solucionarlo.



- **Requisito 15:** “The system must handle **customers dashboards** with the following **indicators**:
- The last five destinations.
  - The money spent in bookings during the last year.
  - Their number of bookings grouped by travel class.
  - Count, average, minimum, maximum, and standard deviation of the cost of their bookings in the last five years.
  - Count, average, minimum, maximum, and standard deviation of the number of passengers in their bookings. “

En este requisito hemos creado una entidad llamada CustomerDashboard que extiende AbstractForm con todos los atributos especificados.



## **Requisitos de Prueba**

Para los requisitos de prueba hemos creado archivos csv de las entidades Booking, Passenger, Customer, BookingRecord, además de las entidades correspondientes al student 1.



## Conclusión

Los principales problemas encontrados en el desarrollo de los requisitos han sido las relaciones entre las entidades ya que dan lugar a confusión. El foro de la asignatura ha sido de gran ayuda para aclarar ciertas relaciones como la de Booking con Passenger y para aclarar las restricciones de los atributos. Por otro lado, destacar la dificultad de tener que implementar también los requisitos del student 1 ya que no hay un rol de student 1 asignado en nuestro grupo de 4 personas.