

Este grupo de pruebas fue realizado por Héctor Herráiz, Miguel Jimeno y Alberto Manchen del grupo 85. Se evaluaron diferentes aspectos de la funcionalidad del sistema, incluyendo la validación de tarjetas de crédito, números de DNI, información de nombres y apellidos, números de teléfono, tipos de habitaciones, número de días de reserva y fechas de llegada.

#### **CREDIT CARD:**

**Clase de Equivalencia Valida (CEV1):** Esta clase de equivalencia representa una tarjeta de crédito válida con 16 dígitos que sigue el algoritmo de Luhn. Se incluye para garantizar que la función maneje correctamente las tarjetas de crédito válidas, lo que es esencial para su correcto funcionamiento en situaciones normales.

**Clase de Equivalencia No Valida (CENV1, CENV2, CENV3, CENV4):** Estas clases de equivalencia representan diferentes escenarios de tarjetas de crédito no válidas. Incluyen casos como tarjetas con 16 dígitos que no siguen el algoritmo de Luhn, tarjetas que contienen caracteres no numéricos, tarjetas con menos de 16 dígitos y tarjetas con más de 16 dígitos. Es importante probar estos casos para garantizar que la función maneje adecuadamente las situaciones de entrada incorrecta y proporcione mensajes de error claros y precisos.

**Valor Límite Valido (VLV1):** Este valor límite representa una tarjeta de crédito con 16 dígitos, que es la longitud máxima permitida y sigue el algoritmo de Luhn. Se incluye para verificar si la función puede procesar correctamente la entrada con la longitud máxima, lo que es crítico para asegurar que no haya errores debido a la longitud incorrecta de la entrada.

**Valores Límite No Validos (VLNV1, VLNV2):** Estos valores límite representan situaciones extremas donde la longitud de la tarjeta de crédito excede el límite superior o es menor que el límite inferior. Prueban la capacidad de la función para manejar entradas que se salen de los límites aceptables y verificar si se proporcionan mensajes de error adecuados en tales casos.

**TC3 INVALID CREDIT CARD (CENV1):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona una tarjeta de crédito con 16 dígitos que no sigue el algoritmo de Luhn. Esperamos que la función devuelva un mensaje de error indicando que el número de tarjeta de crédito no es válido.

**TC4 INVALID CREDIT CARD (CENV2):** Aquí evaluamos cómo se comporta la función cuando se proporciona una entrada que no es un número, es decir, contiene caracteres no numéricos. El resultado esperado es un mensaje de error indicando que el número de tarjeta de crédito no es válido.

**TC5 INVALID CREDIT CARD (CENV3):** En este caso, estamos probando la capacidad de la función para manejar una tarjeta de crédito con menos de 16 dígitos. Esperamos que la función devuelva un mensaje de error indicando que el número de tarjeta de crédito no es válido.

**TC6 INVALID CREDIT CARD (CENV4):** Similar al caso anterior, aquí estamos probando cómo maneja la función una tarjeta de crédito con más de 16 dígitos. Esperamos que la función devuelva un mensaje de error indicando que el número de tarjeta de crédito no es válido.

## **DNI:**

**Clase de Equivalencia Valida (CEV3):** Esta clase de equivalencia representa un DNI válido que sigue el formato estándar, que consiste en 8 dígitos seguidos de una letra al final. Se incluye para garantizar que la función maneje correctamente DNIs válidos, lo que es fundamental para su correcto funcionamiento en situaciones normales.

**Clase de Equivalencia No Valida (CENV5, CENV6, CENV7, CENV8, CENV9, CENV10, CENV11, CENV12, CENV13):** Estas clases de equivalencia representan diferentes escenarios de DNIs no válidos. Incluyen casos como DNIs que no tienen una letra al final, DNIs con más de 8 dígitos, DNIs con menos de 8 dígitos, DNIs que no contienen ningún número, DNIs que no siguen el algoritmo de creación de DNIs, DNIs con 8 dígitos y más de 1 letra, DNIs con menos de 9 caracteres, y DNIs con menos de 9 caracteres. Es importante probar estos casos para garantizar que la función maneje adecuadamente las situaciones de entrada incorrecta y proporcione mensajes de error claros y precisos.

**Valor Límite Valido (VLV2):** Este valor límite representa un DNI con la longitud máxima permitida, es decir, 9 caracteres (8 dígitos seguidos de una letra). Se incluye para verificar si la función puede procesar correctamente la entrada con la longitud máxima, lo que es crítico para asegurar que no haya errores debido a la longitud incorrecta de la entrada.

**Valores Límite No Validos (VLNV4, VLVN5, VLVN6, VLVN7, VLVN8, VLVN9):** Estos valores límite representan situaciones extremas donde la longitud del DNI excede el límite superior o es menor que el límite inferior, o donde la composición del DNI no sigue el formato estándar. Prueban la capacidad de la función para manejar entradas que se salen de los límites aceptables y verificar si se proporcionan mensajes de error adecuados en tales casos.

**TC9 INVALID DNI (CENV5):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona un DNI que no tiene una letra al final. Esperamos que la función devuelva un mensaje de error indicando que el DNI no es válido.

**TC10 INVALID DNI (CENV6):** Aquí evaluamos cómo se comporta la función cuando se proporciona un DNI que ya tiene una letra de reserva al final. El resultado esperado es un mensaje de error indicando que el DNI no es válido.

**TC11 INVALID DNI (CENV7):** En este caso, estamos probando la capacidad de la función para manejar un DNI con más de 8 dígitos. Esperamos que la función devuelva un mensaje de error indicando que el DNI no es válido.

**TC12 INVALID DNI (CENV8):** Similar al caso anterior, aquí estamos probando cómo maneja la función un DNI que no sigue el algoritmo de creación. Esperamos que la función devuelva un mensaje de error indicando que el DNI no es válido.

**TC13 INVALID DNI (CENV9):** En este caso, estamos probando cómo maneja la función un DNI con 8 dígitos y más de 1 letra. Esperamos que la función devuelva un mensaje de error indicando que el DNI no es válido.

**TC14 INVALID DNI (CENV10):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona un DNI que no contiene ningún número. Esperamos que la función devuelva un mensaje de error indicando que el DNI no es válido.

## **NOMBRE Y APELLIDO:**

**Clase de Equivalencia Valida (CEV5):** Esta clase de equivalencia representa un nombre y apellido válidos en el formato estándar, que consiste en una cadena de caracteres que contiene tanto el nombre como el apellido separados por un espacio. Se incluye para garantizar que la función maneje correctamente nombres y apellidos válidos, lo que es fundamental para su correcto funcionamiento en situaciones normales.

**Clase de Equivalencia No Valida (CENV15, CENV16, CENV17):** Estas clases de equivalencia representan diferentes escenarios de nombres y apellidos no válidos. Incluyen casos como nombres y apellidos con una longitud fuera del rango aceptable, nombres y apellidos que no contienen tanto un nombre como un apellido, y nombres y apellidos que no son cadenas de caracteres. Es importante probar estos casos para garantizar que la función maneje adecuadamente las situaciones de entrada incorrecta y proporcione mensajes de error claros y precisos.

**Valor Límite Valido (VLV3):** Este valor límite representa un nombre y apellido con la longitud máxima permitida, es decir, 50 caracteres. Se incluye para verificar si la función puede procesar correctamente la entrada con la longitud máxima, lo que es crítico para asegurar que no haya errores debido a la longitud incorrecta de la entrada.

**Valores Límite No Validos (VLNV10, VLVN11, VLVN12):** Estos valores límite representan situaciones extremas donde la longitud del nombre y apellido excede el límite superior o es menor que el límite inferior, o donde la composición del nombre y apellido no sigue el formato estándar. Prueban la capacidad de la función para manejar entradas que se salen de los límites aceptables y verificar si se proporcionan mensajes de error adecuados en tales casos.

**TC6 INVALID NAME SURNAME (CENV15):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona un nombre y apellido con una longitud fuera del rango aceptable. Esperamos que la función devuelva un mensaje de error indicando que la cadena del nombre y apellidos no es válida.

**TC7 INVALID NAME SURNAME (CENV16):** Aquí evaluamos cómo se comporta la función cuando se proporciona un nombre y apellido que no contiene tanto un nombre como un apellido. El resultado esperado es un mensaje de error indicando que la cadena del nombre y apellidos no es válida.

**TC8 INVALID NAME SURNAME (CENV17):** En este caso, estamos probando la capacidad de la función para manejar una entrada que no es una cadena de caracteres. Esperamos que la función devuelva un mensaje de error indicando que la cadena del nombre y apellidos no es válida.

## **TELEFONO:**

**Clase de Equivalencia Valida (CEV7):** Esta clase de equivalencia representa un número de teléfono válido, que consiste en una secuencia de 9 dígitos. Se incluye para asegurar que la función maneje correctamente números de teléfono válidos, lo que es esencial para su funcionamiento adecuado en situaciones normales.

**Clase de Equivalencia No Valida (CENV18, CENV19, CENV20):** Estas clases de equivalencia representan diferentes escenarios de números de teléfono no válidos. Incluyen casos como

números de teléfono que contienen caracteres que no son dígitos, números de teléfono con más de 9 dígitos y números de teléfono con menos de 9 dígitos. Es importante probar estos casos para garantizar que la función maneje adecuadamente las situaciones de entrada incorrecta y proporcione mensajes de error claros y precisos.

**Valor Límite Valido (VLV4):** Este valor límite representa un número de teléfono con la longitud máxima permitida, es decir, 9 dígitos. Se incluye para verificar si la función puede procesar correctamente la entrada con la longitud máxima, lo que es crítico para asegurar que no haya errores debido a la longitud incorrecta de la entrada.

**Valores Límite No Validos (VLNV10, VLVN11):** Estos valores límite representan situaciones extremas donde la longitud del número de teléfono es menor o mayor que el límite aceptable. Prueban la capacidad de la función para manejar entradas que se salen de los límites aceptables y verificar si se proporcionan mensajes de error adecuados en tales casos.

**TC15 INVALID PHONENUMBER (CENV18):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona un número de teléfono que contiene caracteres que no son dígitos. Esperamos que la función devuelva un mensaje de error indicando que el número de teléfono no es válido.

**TC16 INVALID PHONENUMBER (CENV19):** Aquí evaluamos cómo se comporta la función cuando se proporciona un número de teléfono con más de 9 dígitos. El resultado esperado es un mensaje de error indicando que el número de teléfono no es válido.

**TC17 INVALID PHONENUMBER (CENV20):** En este caso, estamos probando la capacidad de la función para manejar un número de teléfono con menos de 9 dígitos. Esperamos que la función devuelva un mensaje de error indicando que el número de teléfono no es válido.

#### **TIPO DE HABITACION:**

**Clase de Equivalencia Valida (CEV8):** Esta clase de equivalencia representa cadenas de caracteres válidas para el tipo de habitación, que pueden ser "SINGLE", "DOUBLE" o "SUITE". Se incluye para asegurar que la función maneje correctamente los tipos de habitación válidos, lo que es esencial para su funcionamiento adecuado en situaciones normales.

**Clase de Equivalencia No Valida (CENV21, CENV22):** Estas clases de equivalencia representan diferentes escenarios de tipos de habitación no válidos. Incluyen casos como tipos de habitación que no están en la lista dada y casos donde se proporcionan más de una cadena válida. Es importante probar estos casos para garantizar que la función maneje adecuadamente las situaciones de entrada incorrecta y proporcione mensajes de error claros y precisos.

**Valor Límite Valido (VLV5, VLV6, VLV7):** Estos valores límite representan los tipos de habitación válidos permitidos: "SINGLE", "DOUBLE" y "SUITE". Se incluyen para verificar si la función puede procesar correctamente la entrada con los tipos de habitación válidos, lo que es crítico para asegurar que no haya errores debido a tipos de habitación incorrectos.

**TC18 INVALID ROOM TYPE (CENV21):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona un tipo de habitación que no está en la lista dada. Esperamos que la función devuelva un mensaje de error indicando que el tipo de habitación no es válido.

**TC19 INVALID ROOM TYPE (CENV22):** Aquí evaluamos cómo se comporta la función cuando se proporcionan más de una cadena válida para el tipo de habitación. El resultado esperado es un mensaje de error indicando que el tipo de habitación no es válido.

#### **NUMERO DE DIAS:**

**Clase de Equivalencia Valida (CEV10):** Esta clase de equivalencia representa un número de días válido, en este caso, 8. Se incluye para asegurar que la función maneje correctamente los números de días válidos, lo que es esencial para su funcionamiento adecuado en situaciones normales.

**Clase de Equivalencia No Valida (CENV23, CENV24):** Estas clases de equivalencia representan diferentes escenarios de números de días no válidos. Incluyen casos como días fuera del rango permitido y casos donde se proporciona un valor que no es un número. Es importante probar estos casos para garantizar que la función maneje adecuadamente las situaciones de entrada incorrecta y proporcione mensajes de error claros y precisos.

**Valor Límite Valido (VLV8):** Este valor límite representa el rango de números de días permitido, que va desde 1 hasta 10. Se incluye para verificar si la función puede procesar correctamente la entrada con los números de días dentro del rango válido, lo que es crítico para asegurar que no haya errores debido a valores de días incorrectos.

**TC24 INVALID NUMDAYS (CENV23):** Este caso de prueba válida cómo maneja la función la entrada cuando se proporciona un número de días fuera del rango permitido. Esperamos que la función devuelva un mensaje de error indicando que el número de días no es válido.

**TC25 INVALID NUMDAYS (CENV23):** Aquí evaluamos cómo se comporta la función cuando se proporciona un número de días mayor que el máximo permitido. El resultado esperado es un mensaje de error indicando que el número de días no es válido.

**TC26 INVALID NUMDAYS (CENV24):** En este caso, probamos cómo la función maneja la entrada cuando se proporciona un valor que no es un número como número de días. Esperamos que la función devuelva un mensaje de error indicando que el número de días no es válido.

#### **CONCLUSION:**

Se observó que el sistema demostró un buen desempeño al reconocer tarjetas de crédito válidas según el algoritmo de Luhn, así como números de DNI con el formato correcto. Además, pudo identificar de manera efectiva errores comunes, como la presencia de caracteres no numéricos en números de teléfono o la introducción de fechas en un formato incorrecto.

Sin embargo, también se detectaron algunas áreas de mejora. Por ejemplo, en el manejo de nombres y apellidos, se encontraron casos donde la longitud de la cadena excedía el rango permitido, lo que sugiere la necesidad de una validación más estricta en este aspecto. Del mismo modo, se identificaron situaciones en las que el sistema no pudo reconocer tipos de habitación válidos, lo que indica la necesidad de una revisión en la lógica de validación.

En resumen, las pruebas realizadas por el grupo T07 proporcionaron una evaluación completa de la funcionalidad del sistema, destacando tanto sus puntos fuertes como las áreas que requieren atención adicional. Estas conclusiones servirán como base para futuras mejoras y optimizaciones en el desarrollo del sistema.