ANEXO 1

1. **Cobertura de Pruebas:**
   * **Auth (Autenticación)**
     + **POST /auth**: Verificar la autenticación con credenciales correctas.
     + **POST /auth**: Verificar la autenticación con credenciales incorrectas.
   * **Booking (Reservas)**
     + **POST /booking**: Crear una reserva con datos válidos.
     + **GET /booking/{id}**: Obtener una reserva existente.
     + **PUT /booking/{id}**: Actualizar una reserva existente.
     + **PATCH /booking/{id}**: Actualizar parcialmente una reserva.
     + **DELETE /booking/{id}**: Eliminar una reserva existente.
     + **GET /booking**: Obtener todas las reservas y filtrar por diferentes criterios.
   * **Ping (Verificación del estado del servidor)**
     + **GET /ping**: Verificar que el servidor esté operativo.

**Escenarios de Prueba**

1. **Auth (Autenticación)**
   * **Escenario 1: Autenticación Exitosa**
     + **Dado** que se proporcionan credenciales válidas.
     + **Cuando** se realiza una solicitud de autenticación.
     + **Entonces** el servidor responde con un token de autenticación válido.
2. **Booking (Reservas)**

**Escenario 1: Creación de Reserva Exitosa**

* + - **Dado** que se proporcionan datos válidos para una nueva reserva.
    - **Cuando** se realiza una solicitud de creación de reserva.
    - **Entonces** el servidor responde con los detalles de la nueva reserva creada.

**Escenario 2: Obtener Reserva Exitosa**

* + - **Dado** que existe una reserva con un ID específico.
    - **Cuando** se realiza una solicitud para obtener la reserva.
    - **Entonces** el servidor responde con los detalles de la reserva.

**Escenario 3: Actualización de Reserva Exitosa**

* + - **Dado** que se proporcionan datos válidos para actualizar una reserva existente.
    - **Cuando** se realiza una solicitud de actualización de reserva.
    - **Entonces** el servidor responde con los detalles actualizados de la reserva.

**Escenario 4: Eliminación de Reserva Exitosa**

* + - **Dado** que existe una reserva con un ID específico.
    - **Cuando** se realiza una solicitud para eliminar la reserva.
    - **Entonces** el servidor responde con un código de estado 201 indicando la eliminación exitosa.

**Escenario 5: Filtro de Reservas por Criterios**

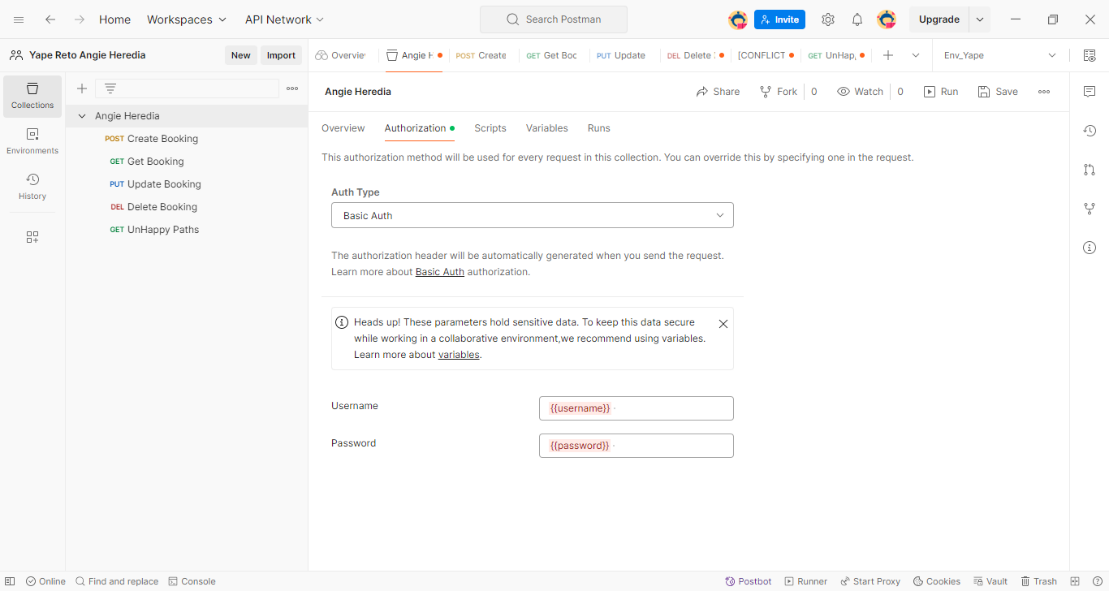
* + - **Dado** que existen reservas con diferentes criterios.
    - **Cuando** se realiza una solicitud para obtener reservas filtradas por un criterio específico.
    - **Entonces** el servidor responde con una lista de reservas que coinciden con el criterio.

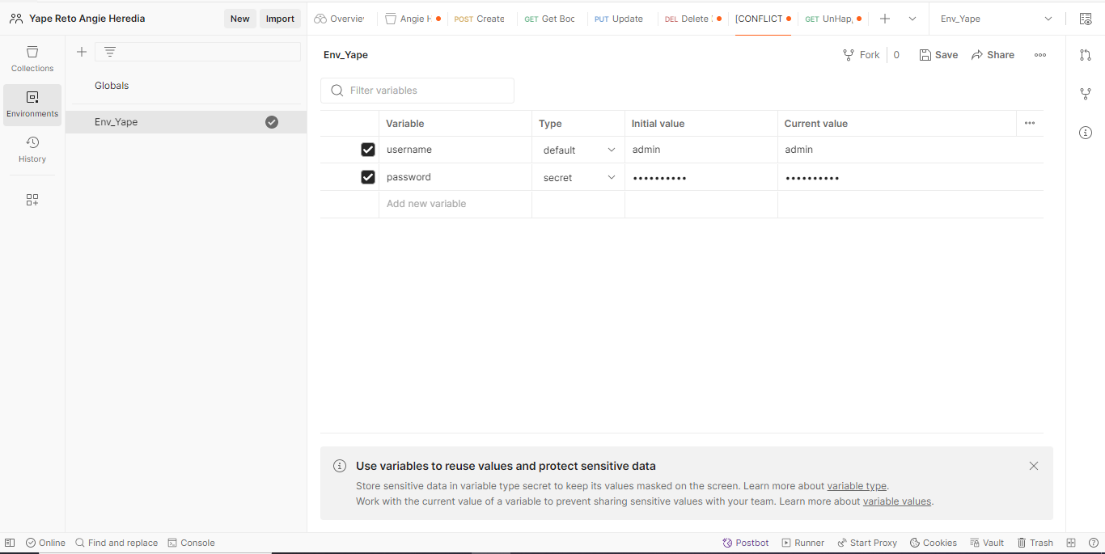
1. **Ping (Verificación del estado del servidor)**

**Escenario 1: Verificación Exitosa del Servidor**

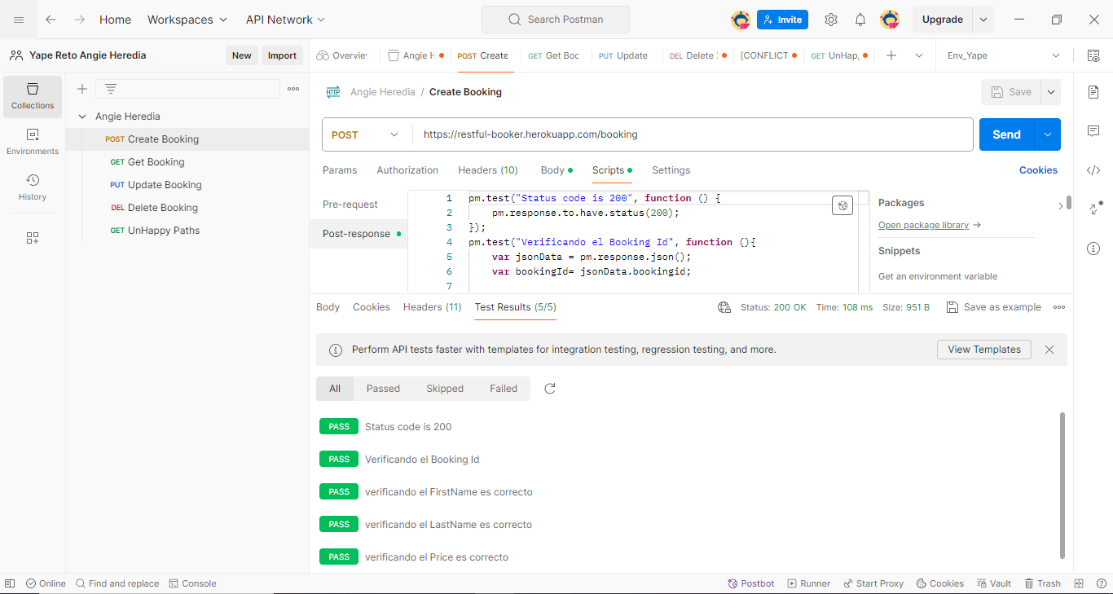
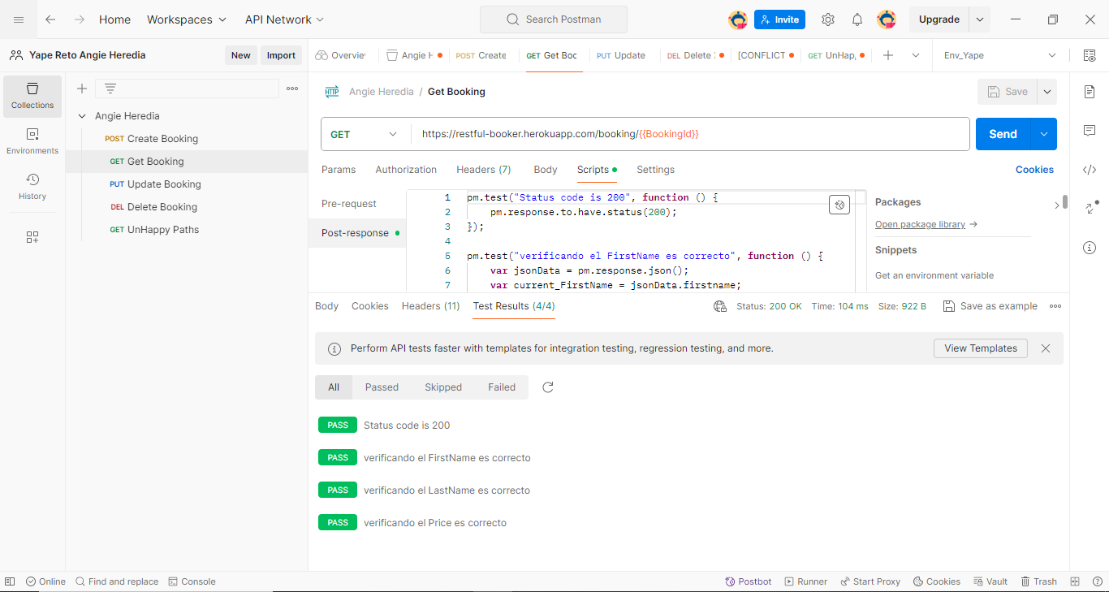
* + - **Dado** que el servidor está operativo.
    - **Cuando** se realiza una solicitud de verificación.
    - **Entonces** el servidor responde con un mensaje indicando que está operativo.

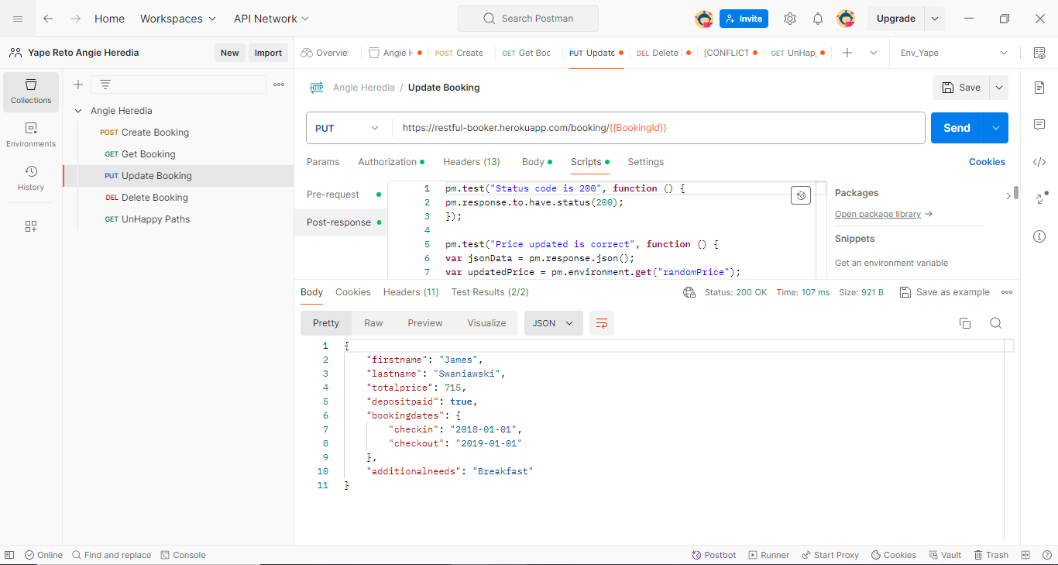
**POSTMAN**

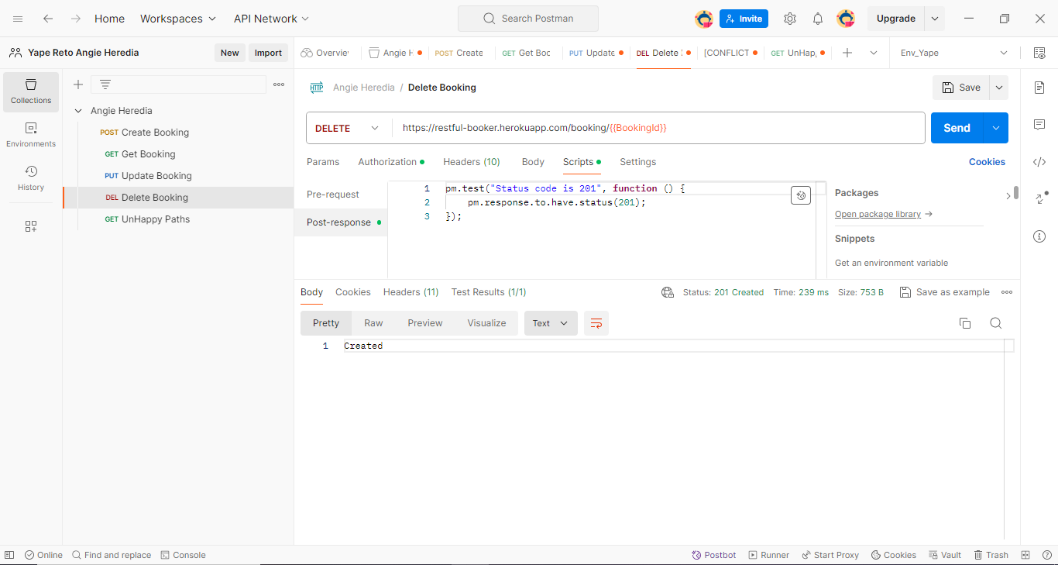
**AUTH**

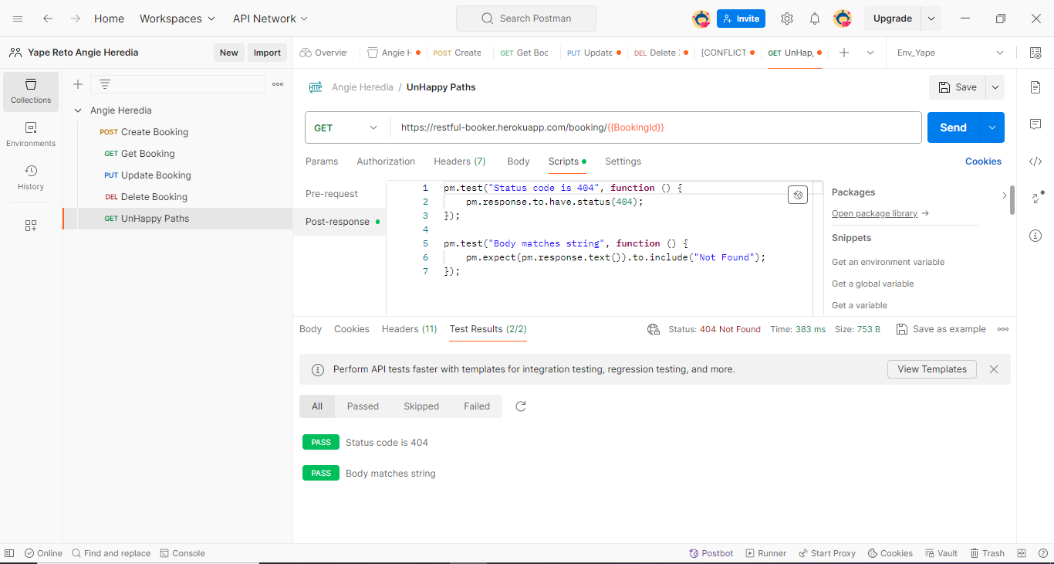


**BOOKING**









**ANEXO 2**

**Cobertura de Pruebas**

1. **Búsqueda de Alojamiento:**
   * Buscar alojamiento en la ciudad de Cusco.
   * Seleccionar fechas de estadía del 14/02 al 28/02.
   * Seleccionar 1 habitación, 2 adultos y 1 niño de 5 años.
2. **Selección y Reserva:**
   * Seleccionar el segundo ítem de los resultados.
   * Elegir la habitación y reservar la primera opción.
   * Llenar los datos de la persona que realiza la reserva.
   * Verificar los datos de la reserva y proceder al último paso.
   * Elegir forma de pago con tarjeta de crédito.

**GHERKIN**

Feature: Booking.com Mobile App

Scenario: Reserva de alojamiento

Given que estoy en la pantalla de búsqueda

When ingreso "Cusco" como destino

And selecciono fechas desde "14/02/2023" hasta "28/02/2023"

And selecciono 1 habitación, 2 adultos y 1 niño de 5 años

And presiono buscar

Then selecciono el segundo ítem de los resultados

And elijo la primera opción de habitación

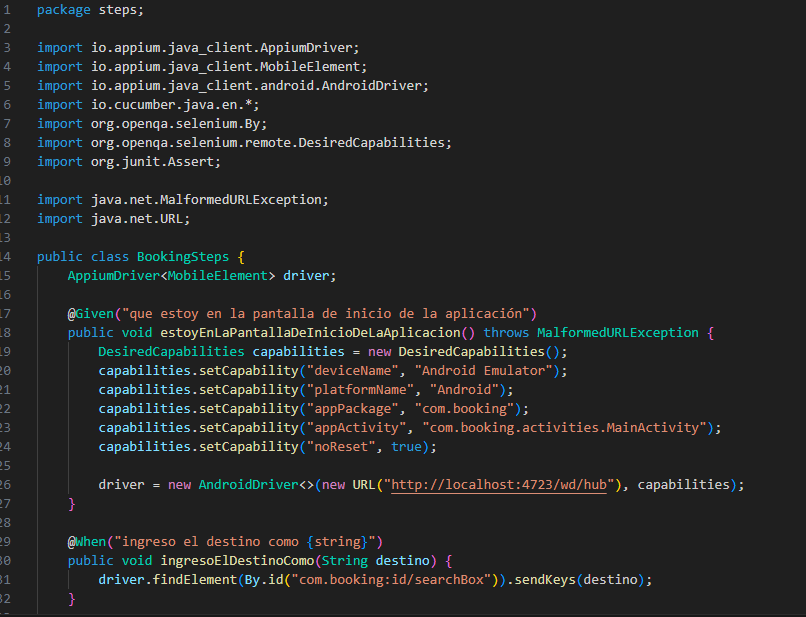
And lleno los datos de la persona que realiza la reserva

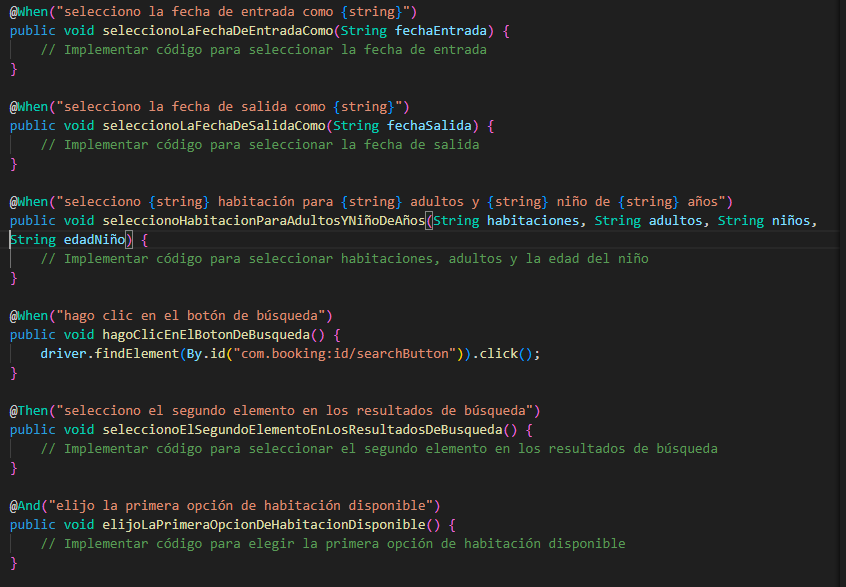
And verifico los datos de la reserva

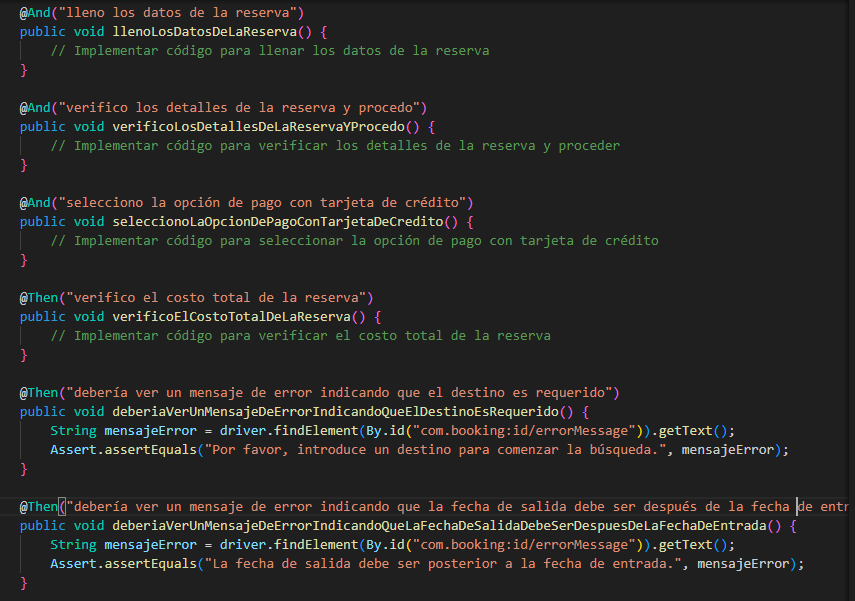
And selecciono la forma de pago tarjeta de crédito

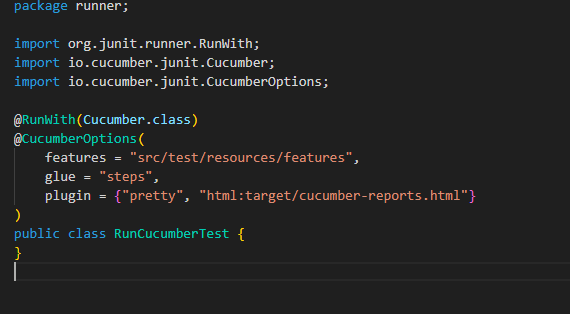
Then la reserva debe completarse correctamente

Happy path









Unhappy path

Feature: Reservar alojamiento en Booking.com

**Scenario: Reservar una habitación de hotel sin ingresar el destino (UnHappy Path)**

Given que estoy en la pantalla de inicio de la aplicación

When ingreso el destino como ""

And selecciono la fecha de entrada como "2023-02-14"

And selecciono la fecha de salida como "2023-02-28"

And selecciono "1" habitación para "2" adultos y "1" niño de "5" años

And hago clic en el botón de búsqueda

Then debería ver un mensaje de error indicando que el destino es requerido

**Scenario: Reservar una habitación de hotel con fechas de entrada y salida incorrectas (UnHappy Path)**

Given que estoy en la pantalla de inicio de la aplicación

When ingreso el destino como "Cusco"

And selecciono la fecha de entrada como "2023-02-28"

And selecciono la fecha de salida como "2023-02-14"

And selecciono "1" habitación para "2" adultos y "1" niño de "5" años

And hago clic en el botón de búsqueda

Then debería ver un mensaje de error indicando que la fecha de salida debe ser después de la fecha de entrada

**Scenario: Reservar una habitación de hotel sin seleccionar una habitación (UnHappy Path**)

Given que estoy en la pantalla de inicio de la aplicación

When ingreso el destino como "Cusco"

And selecciono la fecha de entrada como "2023-02-14"

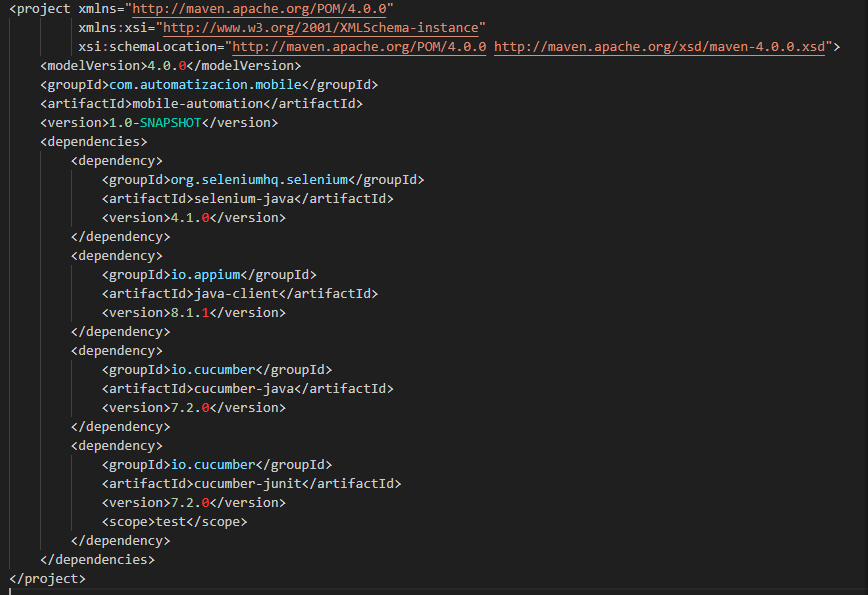
And selecciono la fecha de salida como "2023-02-28"

And selecciono "0" habitaciones para "2" adultos y "1" niño de "5" años

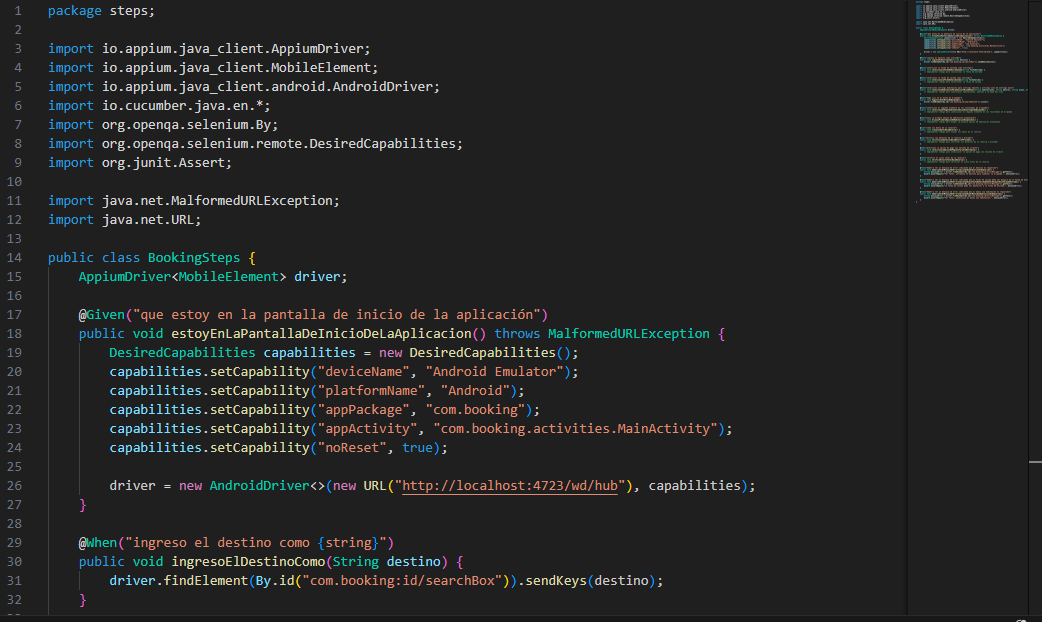
And hago clic en el botón de búsqueda

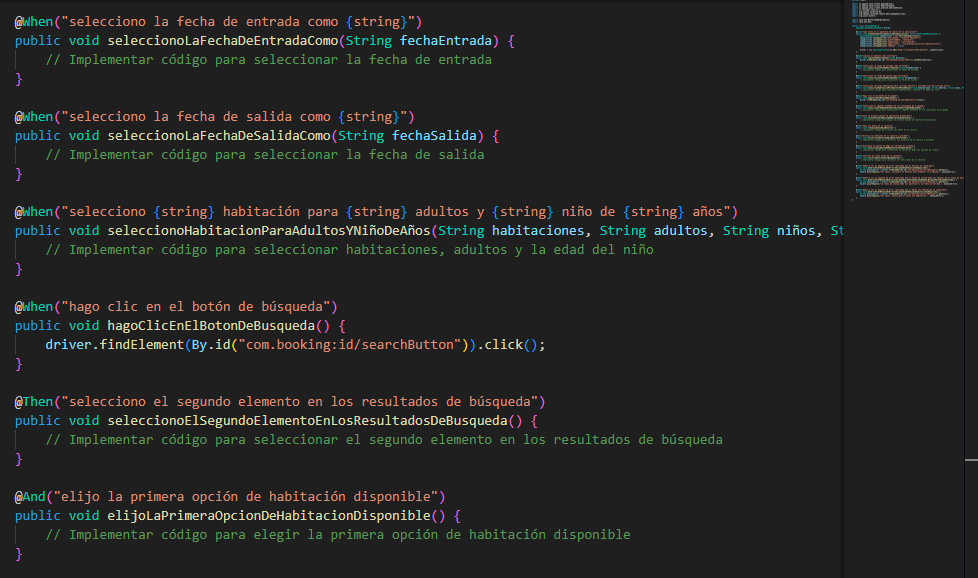
Then debería ver un mensaje de error indicando que al menos una habitación es requerida

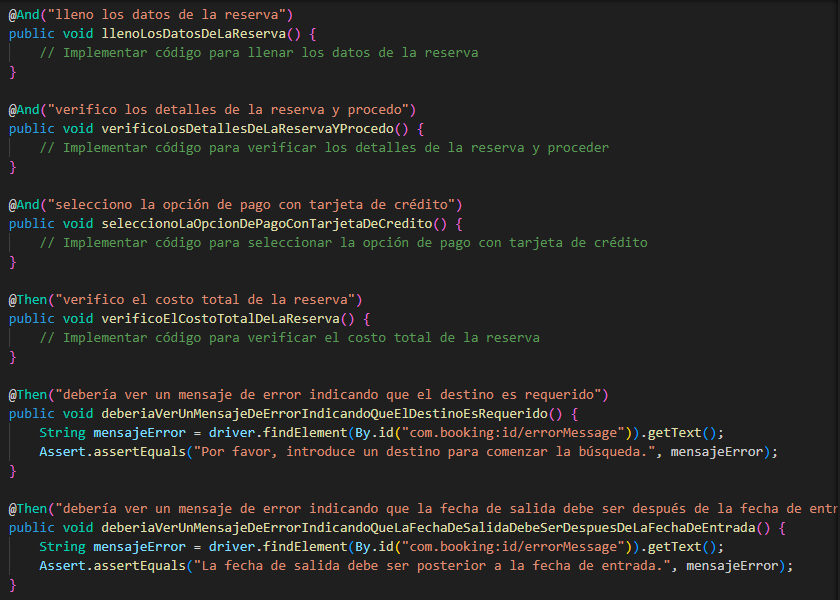
**Dependencias necesarias en el archivo** pom.xml **- Maven**:

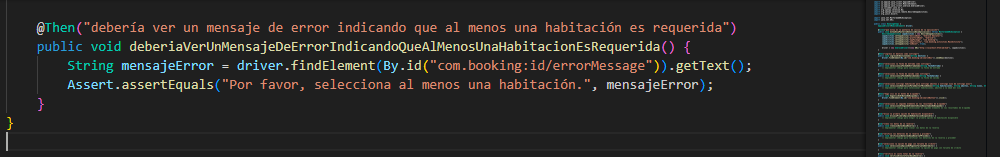


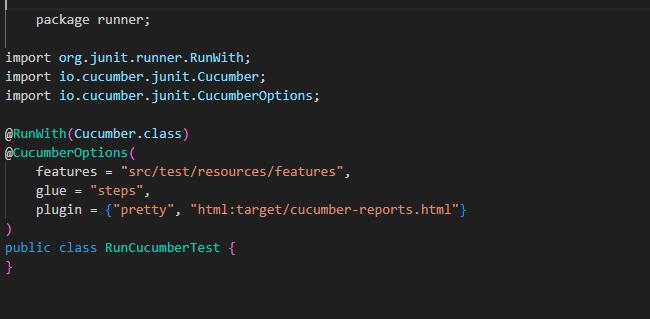
Archivo **BookingSteps.java** en la carpeta **src/test/java/steps**:









Archivo **RunCucumberTest.java** en la carpeta **src/test/java/runner**: