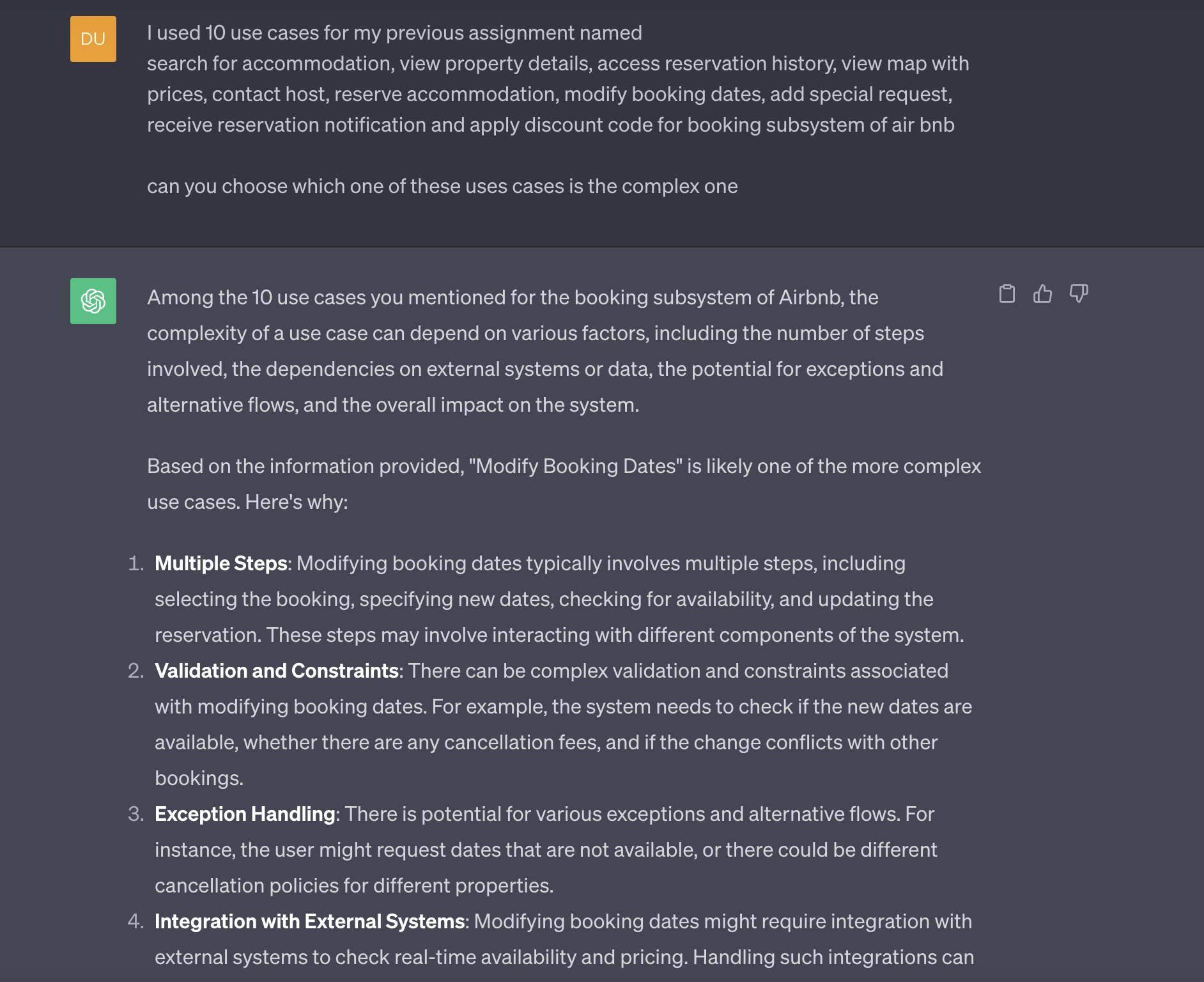
**Plain English description of “Modify Booking Dates”.**

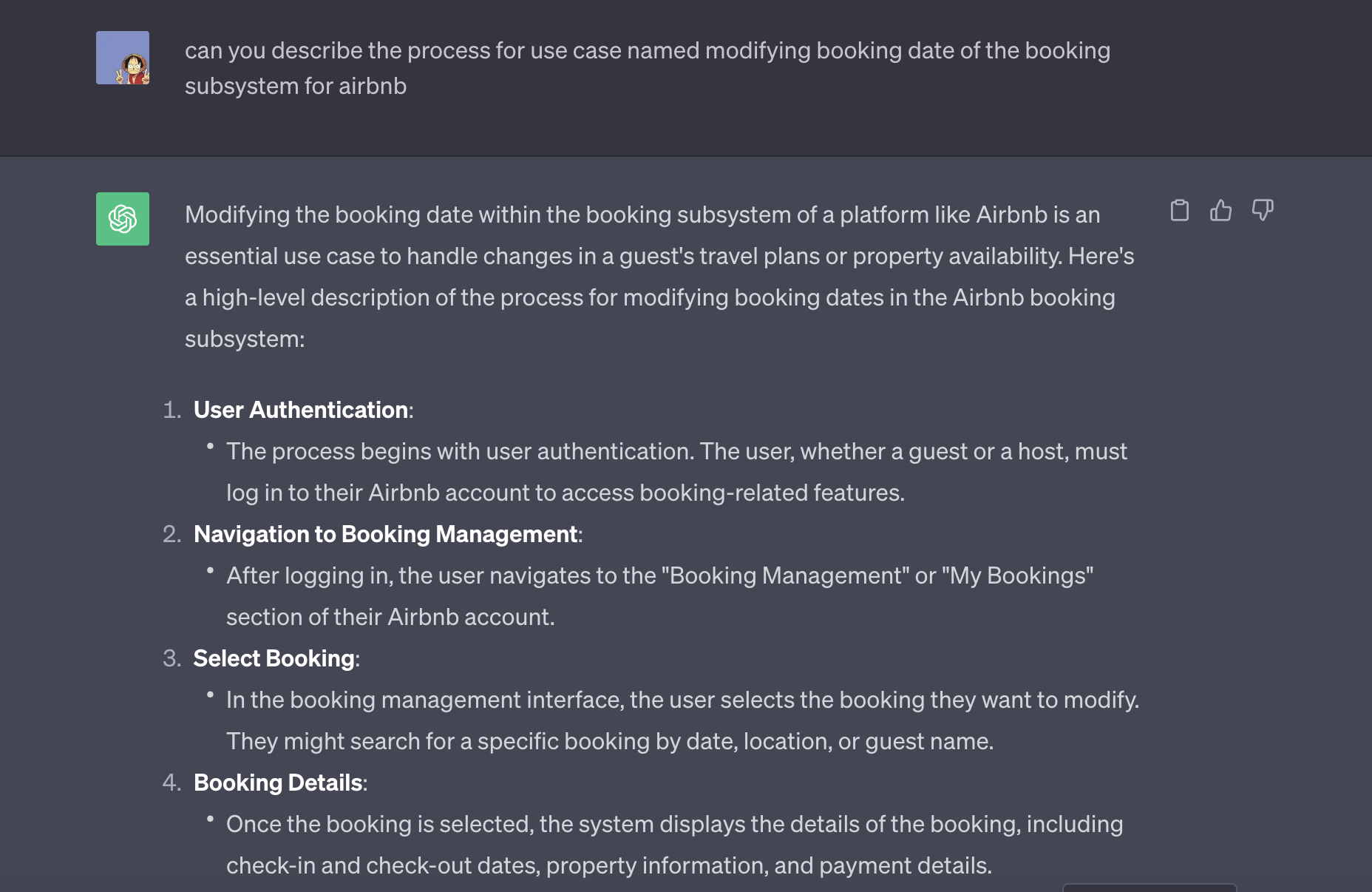
Among the ten use cases provided for the booking subsystem, I found modifying booking dates to be most complex amongst the others provided.

**Process for the mentioned use cases:**

The process of modifying booking dates within Airbnb's booking subsystem begins with user authentication, followed by navigation to the booking management section. Once the desired booking is selected, users initiate the modification, triggering an availability check and price adjustment. After reviewing and confirming the changes, payment details are updated, and confirmation notifications are sent to both users and hosts. An updated booking itinerary is generated, and host notifications are issued if applicable. The system logs the modifications, and cancellation policies are enforced when necessary. Following the stay, users and hosts can provide feedback and ratings. This process ensures flexibility and transparency while maintaining compliance with Airbnb's policies and host requirements.

1. **Drafts and Queries from the AI tool**





1. **Critiques from the AI tool**

The provided paragraph offers a clear and concise explanation of the process for modifying booking dates on Airbnb in plain English. It effectively breaks down the steps involved and uses straightforward language that is easy for a broad audience to understand. However, there are a few minor suggestions for improvement:

1. Clarification on "Confirm Changes": It might be helpful to briefly explain how users confirm the changes, whether it's through a button or a specific action, to provide a more detailed picture.
2. Cancellation Rules: While it mentions following Airbnb's rules for cancellations, it could be beneficial to briefly mention that cancellation policies can vary between bookings and hosts.
3. Feedback and Ratings: Adding a sentence on the significance of feedback and ratings, such as how they contribute to the Airbnb community, could enhance the paragraph's completeness.
4. Communication with the Host: It's a good practice to communicate with the host when you're considering modifying booking dates. While it's not always necessary, discussing the changes with the host can help ensure a smoother process, especially if there are any specific considerations or agreements regarding the modification.
5. Timely Modification: Try to request date modifications as early as possible. Hosts often appreciate guests who provide ample notice for changes, as this gives them more time to adjust their availability and bookings, accordingly, potentially avoiding conflicts with other guests.
6. **Final Solution:**

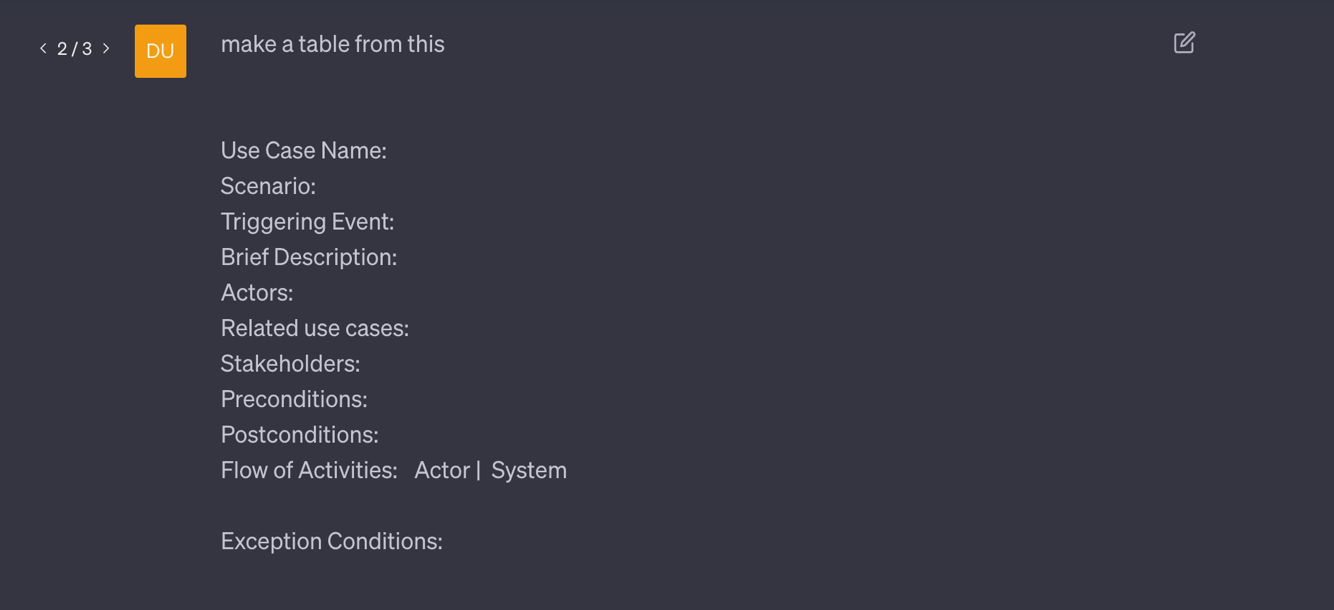
In the Airbnb booking subsystem, the process of modifying booking dates begins with user authentication and navigation to the 'Booking Management' section. Users select the specific booking they wish to modify and initiate the changes, triggering an availability check and price recalculation to ensure the property's availability and adjusted costs. Upon confirming the changes, payment details are updated, and both users and hosts receive confirmation notifications. The system logs the modifications for reference, and in cases of cancellation, Airbnb's specific cancellation policies are enforced. After the stay, users and hosts can provide feedback and ratings, fostering trust within the Airbnb community. This user-centric process ensures flexibility for travellers while adhering to Airbnb's policies and respecting host preferences.

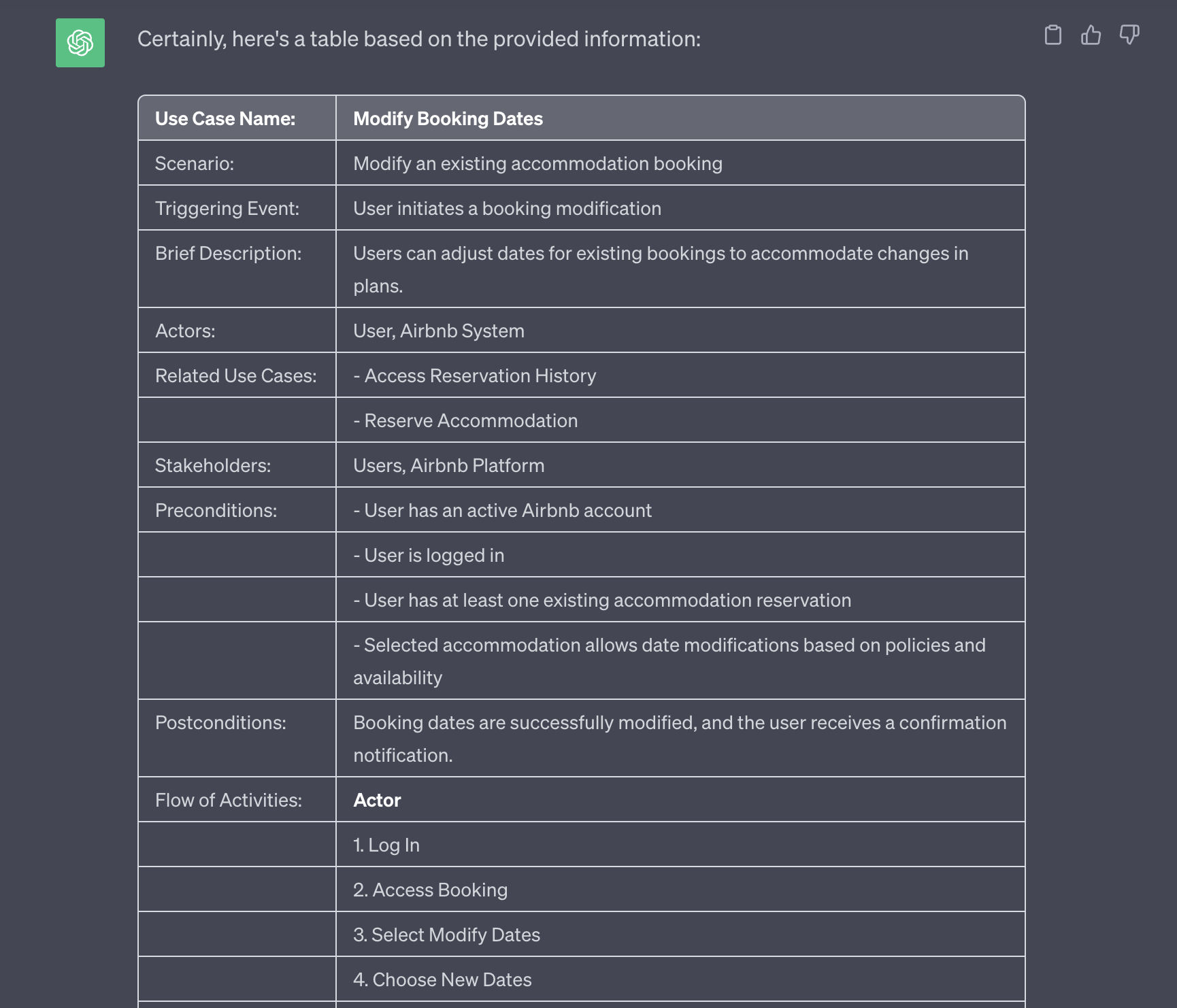
**Extended Use Case Description**

The extended use case description for the Modifying Booking Date is presented below:

|  |  |  |
| --- | --- | --- |
| **Use Case Name:** | Modify Booking Dates | |
| **Scenario:** | Modify an existing accommodation booking | |
| **Triggering Event:** | User initiates a booking modification | |
| **Brief Description:** | Users can adjust dates for existing bookings to accommodate changes in plans. | |
| **Actors:** | User, Airbnb System | |
| **Related use cases:** | Access Reservation History  Reserve Accommodation | |
| **Stakeholders:** | Users, Airbnb Platform | |
| **Preconditions:** | 1. User has an active Airbnb account. 2. User is logged in. 3. User has at least one existing accommodation reservation 4. Selected accommodation allows date modifications based on policies and availability | |
| **Postconditions:** | 1. Booking dates are successfully modified, and the user receives a confirmation notification. 2. The updated booking details, including the modified dates, are reflected accurately in the user's reservation history. | |
| **Flow of Activities:** | **Actor** | **System** |
| 1. User logs into their Airbnb account. 2. User accesses their existing bookings. 3. User selects the "Modify Dates" option. 4. User specifies new check-in and check-out dates.   .   1. User reviews and accepts pricing adjustments. 2. User confirms the modification. 3. User makes additional payment if necessary. 4. User receives a confirmation notification. | 1. Performs a real-time availability check. 2. Notifies user if new dates are unavailable. 3. Prompts user for payment if required. 4. Sends confirmation notification with updated details. |
| **Exception Conditions:** | - If new dates are unavailable, notify the user and provide options to select different dates or cancel the modification.  - If payment for the price adjustment fails, inform the user and guide them on resolving the payment issue. | |

1. Drafts and Queries from the AI Tool:





1. Critiques from the AI Tool

**Completeness:** The provided use case description is quite comprehensive. It covers various aspects of the process, from preconditions to postconditions, and includes detailed steps in the flow of activities.

**Clarity:** The use case is written in a clear and concise manner. Each step is logically organized and easy to follow, ensuring that both technical and non-technical stakeholders can understand it.

**Use of Actors:** The key actors, including the User and Airbnb System, are properly identified. Additionally, related use cases (Access Reservation History and Reserve Accommodation) are mentioned, which helps establish context and dependencies.

**Preconditions and Postconditions:** Preconditions and postconditions are well-defined. They provide a clear understanding of what must be in place before the use case can start and what outcomes to expect once it's completed. However, explicitly specifying the user's existing booking(s) could enhance clarity.

**Flow of Activities:** The flow of activities is structured logically, with each step labelled and described. It's helpful that the system's actions (I, II, III, IV) are outlined alongside the user's actions, making it easier to follow the interaction between the user and the system.

1. Final Solution

In the revised extended use case description for the "Modify Booking Dates" use case within Airbnb's booking subsystem, the document maintains a clear and concise structure. It addresses the critiques and suggestions provided, such as explicitly mentioning real-time availability checks for date validation and enhancing exception handling by providing clear actions in case of date unavailability or payment issues. The user experience is improved by offering options and guidance in exceptional scenarios. This final version of the use case description ensures comprehensive coverage of the process, promoting a more user-friendly and robust experience for Airbnb users seeking to modify their booking dates.

Activity Diagram

Figure:

**System Sequence Diagram (SSD)**

**Test Plan:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Guideline** | **Short Description** | **Test Data** | **Expected Output** |
| 1 | RIGHT-BICEP | Test reservation modification initiation. | User initiates modification. | The system accepts the modification request. |
| 2 | RIGHT-BICEP | Test new date selection. | User specifies new check-in and check-out dates. | The system records the new dates. |
| 3 | RIGHT-BICEP | Test availability check for new dates. | User selects new dates that are available. | The system proceeds with the modification. |
| 4 | RIGHT-BICEP | Test price adjustment calculation. | User confirms the modification, and the system calculates a price adjustment. | The system calculates the price adjustment accurately. |
| 5 | RIGHT-BICEP | Test payment process. | User confirms the modification, and an additional payment is required. | The system prompts the user for payment, and the payment process is successful. |
| 6 | Boundary - CORRECT | Test no price adjustment. | User confirms the modification, and there is no price adjustment. | No price adjustment is applied. |
| 7 | Boundary - CORRECT | Test notification to the user. | User receives a confirmation notification after successful modification. | User receives the confirmation notification. |
| 8 | Boundary - CORRECT | Test notification to the host. | Host receives a notification about the modification. | Host receives the notification about the modification. |
| 9 | Boundary - CORRECT | Test system log of modifications. | The system logs the modification for reference. | The modification is logged appropriately. |
| 10 | Boundary - CORRECT | Test cancellation policy enforcement. | In case of cancellation, Airbnb's specific cancellation policies are enforced. | The system enforces the cancellation policy correctly. |
| 11 | Boundary - CORRECT | Test feedback and ratings after the stay. | Users and hosts can provide feedback and ratings after the stay. | Users and hosts can provide feedback and ratings as expected. |
| 12 | Boundary - CORRECT | Test flexibility and policy compliance. | The process ensures flexibility for travellers while adhering to Airbnb's policies. | The process respects both flexibility and policy compliance. |

Queries and Drafts from AI Tool:

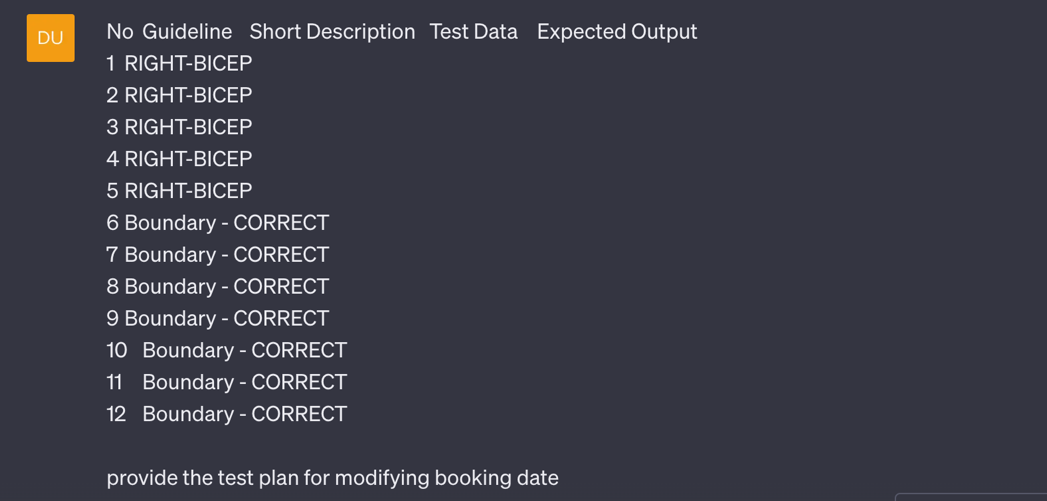


Figure:

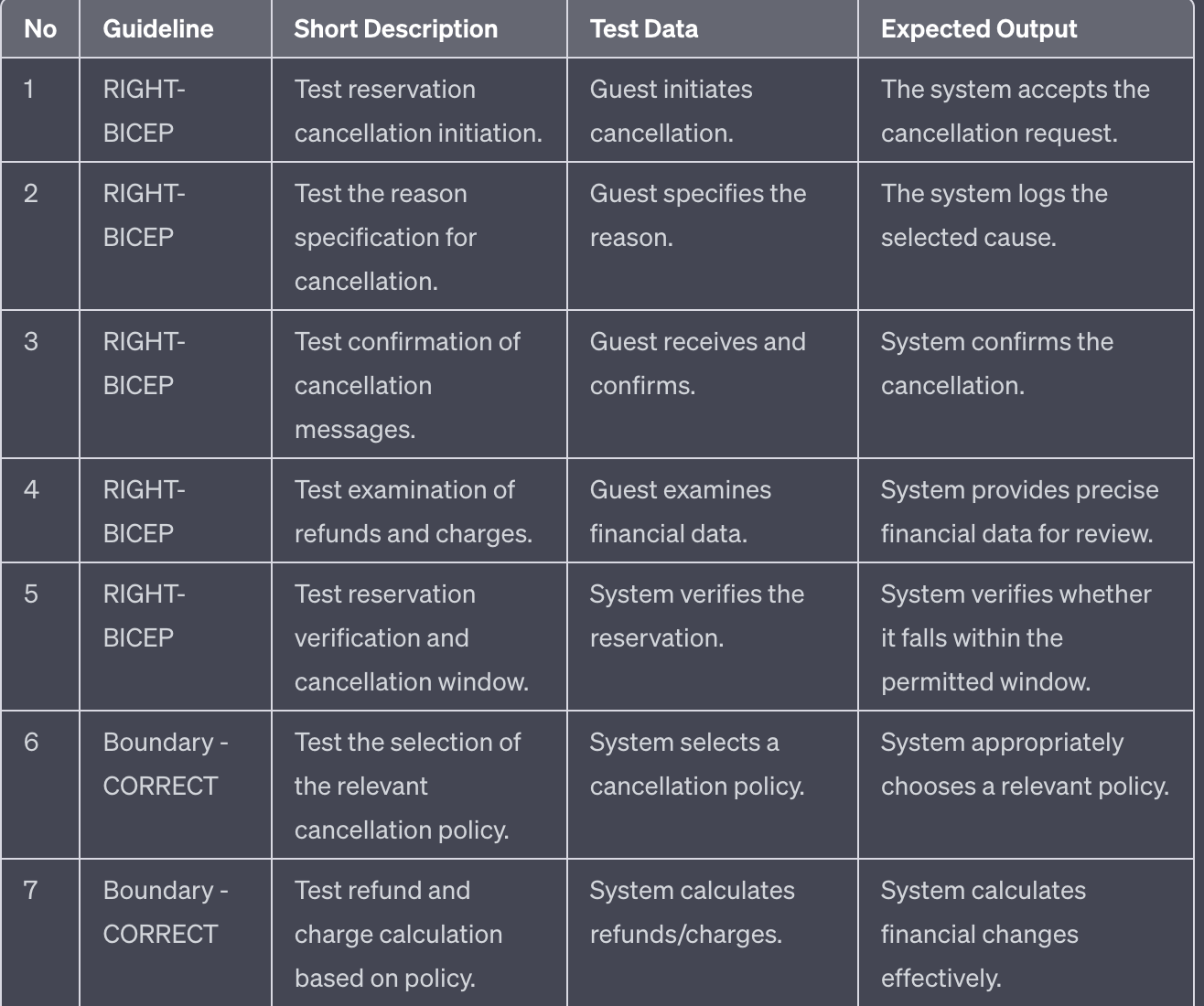


Figure:

**Critiques from the AI Tool:**

1. **Expected Output Details:** The "Expected Output" column should be more specific to make the expected results clearer. For instance, instead of saying "The system accepts the modification request," it could specify what the system should respond with upon accepting the modification request, such as "The system responds with a 'Modification request accepted' message."
2. **Coverage of Negative Scenarios:** The test plan primarily focuses on optimistic scenarios, and it would be beneficial to include negative cases or error handling tests. Negative scenarios, such as when the new dates selected are not available or when payment processing fails, should be considered, and documented. This ensures robust testing and helps uncover potential issues.
3. **Test Preconditions:** The test preconditions, which are prerequisites that must be met before a test case can be conducted, are not specifically mentioned in the plan. It's crucial to state that there must be an existing reservation to initiate the modification in test case 1 (modification initiation). This clarification is necessary to ensure that the tests are conducted under the appropriate conditions.
4. **Boundary Testing:** Although boundary testing is mentioned in the design, no precise boundary values or test circumstances are given. Defining precise boundary values, especially when dealing with financial calculations and policy, would enhance the test plan's completeness. Boundary testing is essential to assess how the system behaves at the limits of its functionality.
5. **Dispute Resolution:** Although dispute resolution is mentioned in the plan, it is not clear how it would be evaluated. The plan would be more complete if it included hypothetical situations that represent conflicts and their resolutions. Adding scenarios related to dispute resolution and their expected outcomes would provide a more comprehensive evaluation of this aspect of the system.

Final Solution:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Guideline** | **Short Description** | **Test Data** | **Expected Output** |
| 1 | RIGHT-BICEP | Test reservation modification initiation. | User initiates modification. | The system responds with a 'Modification request accepted' message. |
| 2 | RIGHT-BICEP | Test new date selection. | User specifies new check-in and check-out dates. | The system records the new dates accurately. |
| 3 | RIGHT-BICEP | Test availability check for new dates. | User selects new dates that are available. | The system proceeds with the modification and displays a 'Modification successful' message. |
| 4 | RIGHT-BICEP | Test price adjustment calculation. | User confirms the modification, and the system calculates a price adjustment. | The system calculates the price adjustment accurately and displays the adjusted cost to the user. |
| 5 | RIGHT-BICEP | Test payment process. | User confirms the modification, and an additional payment is required. | The system prompts the user for payment, and upon successful payment, it displays a 'Payment successful' message. |
| 6 | Boundary - CORRECT | Test no price adjustment. | User confirms the modification, and there is no price adjustment. | No additional cost is displayed, the system responds with a 'Modification successful' message. |
| 7 | Boundary - CORRECT | Test notification to the user. | User receives a confirmation notification after successful modification. | User receives a notification with a 'Modification confirmed' message. |
| 8 | Boundary - CORRECT | Test notification to the host. | Host receives a notification about the modification. | Host receives a notification with details about the modification. |
| 9 | Boundary - CORRECT | Test system log of modifications. | The system logs the modification for reference. | The modification is accurately logged with a timestamp. |
| 10 | Boundary - CORRECT | Test cancellation policy enforcement. | In case of cancellation, Airbnb's specific cancellation policies are enforced. | The system enforces the cancellation policy correctly and provides a 'Cancellation policy applied' message when applicable. |
| 11 | Boundary - CORRECT | Test feedback and ratings after the stay. | Users and hosts can provide feedback and ratings after the stay. | Users and hosts can submit feedback and ratings, and the system confirms receipt. |
| 12 | Boundary - CORRECT | Test flexibility and policy compliance. | The process ensures flexibility for travellers while adhering to Airbnb's policies. | The system allows flexible modifications and complies with Airbnb's policies. |

**Screen prototype**

The .fig file for the booking subsystem has been included in the submission folder, below is the screenshot of it.

