

Hotels cancellation flow using Agoda APIs:

- Please note that I've handled assumptions as per my knowledge. Feel free to get back to me in case of any discussions or modifications :)

Problem statement: To show the **cancellation policy** using **agoda APIs** and to note down the **responses** used in the **userflow**.

Why should we create this solution?

- Users should **know about the cancellation policy without any issues** and handle their travel planning accordingly.
- To give an **estimate of the refund** in case of a cancellation.
- Having a **cancellation policy** online **without the intervention of hotel management** in few clicks is great user pain point.

Solution:

- I've tried to follow up a **userflow that is least resistant** in nature to the user. Most of the travel websites/applications follow this path.
- I've only designed keeping in mind about the **mobile applications**.
- Since we are focusing on the **cancellation flow**, the main data we need to deal with is the **cancellation related data and booking details**. Agoda provides data in two formats: **JSON and XML**. Let's **focus on using the JSON data** as of now. It can easily be extended to XML data as well.
- But how does Agoda do this? They **provide us with APIs** to fetch this data. We request data through these APIs with **request parameters** and they give us back the **response** back with the **data**. We can then use this data to display it on the **application interface**.
- In this solution, I'll only **comment on the response parameters** of the APIs **and not the request parameters and also error handling** as this solution will become more complicated. Let's keep this simple :)

- However, if you feel the need to deal with the request parameters/error feel free to reach out to me!
- We can use the following APIS:

JSON BookingDetail API: This API gives us the booking details of the user.

The following response parameters are utilised in the userflow shown below:

1. **id:** This response parameter can be extracted from the "bookings" element -> "id" attribute and served on the booking information UI screen as the id of the booking.
2. **propertyName:** This response parameter can be extracted from the "bookings" element -> "property"-> "propertyName" attribute and served on the booking information UI screen as the property name of the booking.
3. **city:** This response parameter can be extracted from the "bookings" element -> "property"-> "city" attribute and served on the booking information UI screen as the property name of the booking.
4. **checkIn:** This response parameter can be extracted from the "bookings" element -> "checkIn" attribute and served on the booking information UI screen as the check in of the booking.
5. **roomType:** This response parameter can be extracted from the "bookings" element -> "room" -> "roomType" attribute and served on the booking information UI screen as the check in of the booking.
6. **roomsBooked:** This response parameter can be extracted from the "bookings" element -> "room" -> "roomsBooked" attribute and served on the booking information UI screen as the check in of the booking.
7. **Occupancy:** This response parameter can be extracted from the "bookings" element -> "occupancy" attribute and served on the booking information UI screen as the id of the booking.
8. **specialRequest:** This response parameter can be extracted from the "bookings" element -> "specialRequest" attribute and served on the booking information UI screen as the id of the booking.

JSON Cancel API: This gives us the cancellation policy and deductions post cancellation.

The following response parameters are utilised in the userflow shown below:

1. **cancellationPolicy:** This response parameter can be extracted from the "cancellationPolicy" element -> "PolicyText" attribute and served on the cancel booking UI screen as the cancellation policy of the hotel.
2. **paymentRate:** This response parameter can be extracted from the "paymentRate" element -> "inclusive" attribute and served on the cancel booking UI screen as total payment paid to the hotel. Additionally the type of currency can be extracted from the API.
3. **refundRate:** This response parameter can be extracted from the "refundRate" element -> "inclusive" attribute and served on the cancel booking UI screen as refund given by the hotel. Additionally the type of currency can be extracted from the API.
4. **Deductions:** This is not directly extracted from the API, but can be calculated by $(\text{paymentRate} - \text{refundRate})$.

JSON confirm cancel API: When we press the confirm cancellation this API is called to confirm the cancellation. This does not return any parameters to be handled.

JSON requestList API: We can use this API to request bookings done based on date range. However, I've not included this API in the userflow for a simpler solution.

You could **read more about these APIs** and their **responses below:**

<https://partners.agoda.com/DeveloperPortal/APIDoc/PostBookingJsonAPI#cancel>

Note: Most of the APIs use request parameter as the booking ID!

Here's a userflow based on the APIs discussed above. I will also attach the high quality JPEG just in case this userflow isn't visible in the document.

