

[Kano.com] Design Critique

Team: The PowerPuff Girls

Ha Young Kim (hayounk@g.clemson.edu)

Date: April 8, 2020

Description of the document

This document is to evaluate the wireframe by Usability Aspects using Nielsen's heuristics. I've looked on a interfaces and features of a booking service named Kano by using a UAR templet. There would be total twelve aspects, including three good aspects and nine problems. For each aspects, there would be evidence and explanation of each and will provide possible solution and/or trade-offs.

Executive Summary

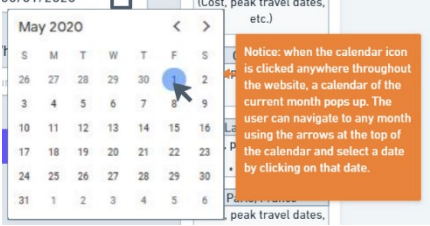
The wireframe was about a booking service named Kano that provides every point that people can use for their trips – lodging, rental car, flight, etc. The service was very well-made so the most problems are very cosmetic. But there are some problems that would make user feel tired to use the service.

The major problem that is very frequent is, absence of sorting options. Because the service mainly provides searching and the results of search includes amount of price and average customer's rates, there should be a sorting option for user's convenient. Also, the system has weak side with preventing errors. There is a section to send a confirmation email, but it doesn't double check for the spelling of confirmation email.

Introduction

Kano.com is a total booking service for trip. It provides Lodging search, Rental car search, Flight search, Bundle search and even provides global search. At the home page, users can do global search by entering departure location, destination location, estimated travel time. For specific search, user can simply change to the tabs they want to find. The search option would change for each category. For example, on Rental cars search tab, there would be an option to choose picking location, vehicle type, but on the Lodging search tab, there would be an option to choose destination location, number of guests, etc. User can use 'advanced search' to narrow their results. Also, by using a side bar on results page, users can filter their results and modify price range and check to see ratings and flexible dates. When customer adds a product, they can view their trip's summary and add special amenities/features before purchase. Users would enter the payment/billing information and confirmation email to complete purchase. Kano.com is very simple website to use and convenient since it has consolidated all categories on a site.

UARs

No. HYK-01	Problem/Good Aspect Good
Name: Calander date select	
Evidence : See the screenshot below.It gives a calendar to select the date for user. 	
Interface aspect : A calendar makes users to select their dates easier and intuitive.	
Heuristic : Match between system and real world, Consistency and standards.	
Explanation : “Match between system and real world” : This heuristic is adhered because the UI design is using a concept of calendar, which is very familiar to the user. “Consistency and standards” : Just like computer’s calender, it has small arrows to change months. So users can easily recognize what arrows mean.	
Severity or Benefit: Rating: NA – good aspect 1 – cosmetic problem (does not matter too much) 2 – minor problem (would be nice to solve, but not a high priority) 3 – major problem (a problem that should definitely be solved with high priority) 4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence): <i>Frequency: high</i> <i>Impact: low</i> <i>Persistence: medium</i> <i>How I weighted the factors:</i> Since this system is for booking for travel, this calendar would be used frequently. It’s a very familiar system to use. Any user can estimate how to use it. Choosing a specific date is the very	

first step to use this system. So, the need to selecting the date is less common.

Possible solution and/or trade-offs:

A potential downside of this feature is that there could be someone who didn't make exact date for travel, just looking this site to find out for prices, which could make hard to compare prices on each date since the user have to choose each date one by one.

Relationships:

None(for now)

No. HYK-02	Problem/Good Aspect Problem
Name: Missing scroll option! -needs one more	
Evidence : See the screenshot below. It only has one marker that can be moved to change price range. But actually user can change both minimum price and maximum price. <div data-bbox="215 443 646 692" data-label="Image"> </div>	
Interface aspect : Using a scroll to modify a price range is a good idea, but it provides only one changeable factor. Since user is able to change two values(minimum and maximum), the system should be also provide two scroll markers.	
Heuristic : User control and freedom	
Explanation : “User control and freedom” : This heuristic is violated because the system only one scrollable marker. Form the textbox, we can expect that both minimum price and maximum price can be changed by user. But on the scroll, there is only one marker – which looks like for maximum price. Using a scroll would make user to change the amount of price range easily, but if user cannot change minimum price by the scroll also, then it’s useless to have a scroll bar to modify the price range. Also, it’s hard to estimate the amount of price with just using scroll.	
Severity or Benefit: Rating: NA – good aspect 1 – cosmetic problem (does not matter too much) 2 – minor problem (would be nice to solve, but not a high priority) 3 – major problem (a problem that should definitely be solved with high priority) 4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence): <i>Frequency: low</i> <i>Impact: high</i> <i>Persistence: medium</i>	

How I weighted the factors:

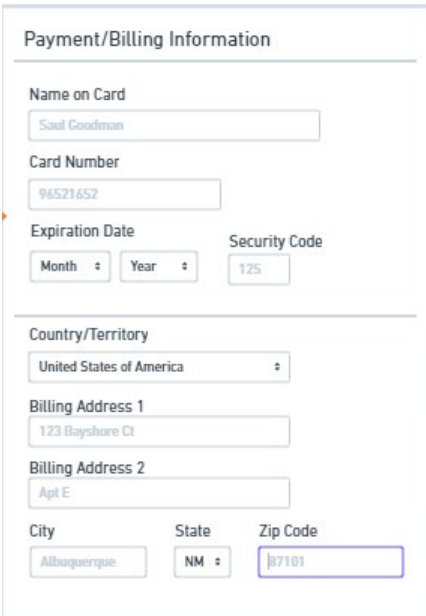
Adjusting the amount of price range is very important for users. Since users would have their own budget range, they won't use this UI that frequently, but if they have to modify their initial price range, it would be quite troublesome to change.

Possible solution and/or trade-offs:

The interface could make another marker for minimum price of scroll, but even though adding another marker user would be hard to recognize which amount their markers are located. To solve this, the scroll bar should show the amount of price briefly by using small scale.

Relationships:

None.

No. HYK-03	Problem/Good Aspect Problem
Name: Auto-fill / Absence of save card information option	
<p>Evidence :</p> <p>See the screenshot below. This is for payment/billing information, and this requires a lot of informations to be typed. It's a lot that user might hard to remember all informations.</p>  <p>Interface aspect:</p> <p>It's very simple but requires users to memorize a lot of informations.</p> <p>Heuristic: Recognition rather than recall, Flexibility and efficiency of use</p>	
<p>Explanation:</p> <p>“Recognition rather than recall” : This heuristic is violated because in this wireframe it is hard to find an option about saving information for further use.</p> <p>“Flexibility and efficiency of use” : This heuristic is violated because in some sites, they provides some previews of user’s information and give options to select when user tries to type the information on this screen.</p>	
<p>Severity or Benefit:</p> <p>Rating: <choose one of the following:></p> <p>NA – good aspect</p> <p>1 – cosmetic problem (does not matter too much)</p> <p>2 – minor problem (would be nice to solve, but not a high priority)</p> <p>3 – major problem (a problem that should definitely be solved with high priority)</p> <p>4 – usability catastrophe (a problem that renders the Web site / app useless)</p>	

Justification (Frequency, Impact, Persistence):

Frequency: low

Impact: high

Persistence: high

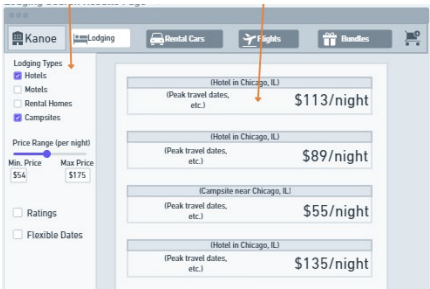
How I weighted the factors:

The payment/billing information is essential to use this system. If user wants to book a trip, they must provide this information to the system. So in user's booking scenario, user would use this interface only once, but it's impact would be very high for the future booking. Also, if the system guarantees to remember user's payment information, it would be very helpful for users. In many platforms that has shopping functions, it provides auto-filled payment option.

Possible solution and/or trade-offs:

The interface is simple but has no options to remember payment information(which is a lot). To solve this problem, the option can be included like a simple checkbox for asking users to 'remember this information for future travel' and also 'use this payment for future travel(s)'. Or simply ask the user to allow using a payment information when the user join-in the site.

Relationships:

No. HYK-04	Problem/Good Aspect Problem
Name : absence of sorting option	
Evidence : <p>See the screenshot below. The system shows lists of lodging that user is looking for and provides many options, but doesn't have a sorting option. Even though users can set a price range, but users should allowed to sort the results by price(low to high or high to low).</p> 	
Interface aspect : <p>Sidebar provides good options for users to select specific category to search, but doesn't have sorting options.</p>	
Heuristic: Consistency and standards	
Explanation : <p>“Consistency and standards” : This heuristic is violated because most travel booking services provides sorting options like sorted by price (low to high, high to low), featured, average customer ratings. The service must provide at least price, average customer ratings sorting because it shows price and ratings on a search results.</p>	
Severity or Benefit: <p>Rating: <choose one of the following:></p> <p>NA – good aspect</p> <p>1 – cosmetic problem (does not matter too much)</p> <p>2 – minor problem (would be nice to solve, but not a high priority)</p> <p>3 – major problem (a problem that should definitely be solved with high priority)</p> <p>4 – usability catastrophe (a problem that renders the Web site / app useless)</p>	
Justification (Frequency, Impact, Persistence): <p><i>Frequency: high</i></p> <p><i>Impact: medium</i></p> <p><i>Persistence: high</i></p> <p><i>How I weighted the factors:</i></p>	

Because this system mainly provide searching results and users would pick one of them, it is essential function for users. Customers who uses price comparing sites wants to use reasonable travel budgets, so they wants to find something more cheaper or has really good ratings. It doesn't mean that user cannot use this system because of lack of sorting function. But it would make users tired to find exact results they want to find and make users willing this function. Since the sorting function is very basic function in other booking/searching platforms, it's a major problem.

Possible solution and/or trade-offs:

The interface does not provide an sorting option, which makes customers feel tired and complicated to find a result they want. To solve this problem, a sorting option should be provided in the sidebar. The option should have at least two selections – sorting by price, sorting by average customer's rate.

Relationships:

No. HYK-05	Problem/Good Aspect Good
Name : Easy to determine upgrade amounts	
Evidence : See the screenshot below. It simply shows user amount of money they have to pay more to upgrade their options. <div data-bbox="217 443 647 907" data-label="Image"> <p>The screenshot shows a flight booking interface for a flight to Chicago, IL. It features two sections: 'Choose Class' and 'Cancellation Insurance'. In the 'Choose Class' section, 'Economy' is selected, 'Business' is \$20.00 more, and 'First' is \$50.00 more. In the 'Cancellation Insurance' section, 'Yes, I would like insurance' is selected for \$15.00, and 'No, I don't want to insure my flight' is an option.</p> </div>	
Interface aspect : Shows briefly how much would cost more to upgrade customer's options.	
Heuristic : Consistency and standards, Aesthetics and minimalist design	
Explanation : “Consistency and standards” : This heuristic adheres because many of other travel booking sites shows up the amount of money customers needs when they want to change their options. “Aesthetics and minimalist design” : This heuristic adheres because it is very simple to see and by giving brief information in a different color, customers can easily recognize it means amount of budget change between original cost and changed cost.	
Severity or Benefit: Rating: <choose one of the following:> NA – good aspect 1 – cosmetic problem (does not matter too much) 2 – minor problem (would be nice to solve, but not a high priority) 3 – major problem (a problem that should definitely be solved with high priority) 4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence): <i>Frequency: medium</i>	

Impact: low

Persistence: high

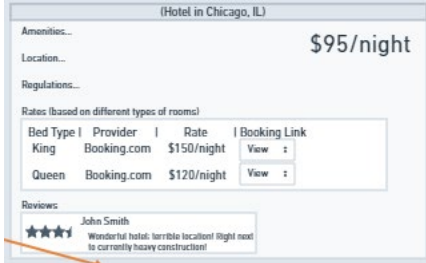
How I weighted the factors:

Every user would meet this interface at least once while they use this site. Not every customers would change their selections because they would have considered a lot of amenities/features before making their decisions.

Possible solution and/or trade-offs:

In the first box, it's order is +\$0, +\$20, +\$50. But in the second box, +\$15 is coming first than +\$0, it could make users confused. The second box's option order should be changed with each other's place.

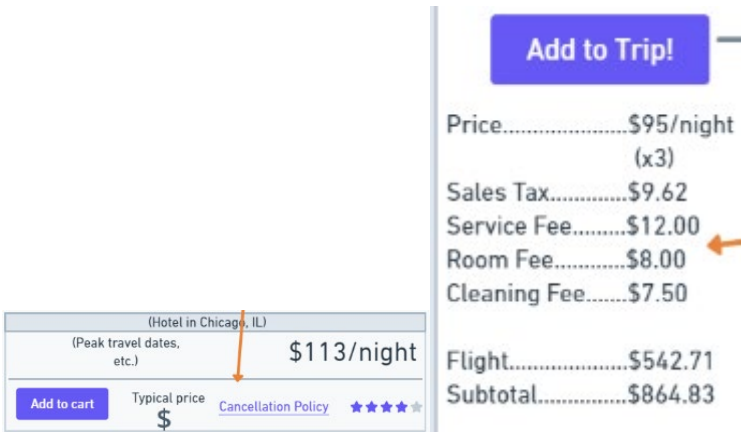
Relationships:

No. HYK-06	Problem/Good Aspect Good
Name: Nice summary	
Evidence : <p>See the screenshot below. It simply shows user about their selection's summary.</p> 	
Interface aspect: <p>Simple summary screen.</p>	
Heuristic: Error prevention, Consistency and standards	
Explanation: <p>“Error prevention” : This heuristic adheres because by taking a look in this interface, users can check their decisions before check out.</p> <p>“Consistency and Standards” : This heuristic adheres because many of sites that has shopping function has these interface commonly.</p>	
Severity or Benefit: <p>Rating: <choose one of the following:></p> <p>NA – good aspect</p> <p>1 – cosmetic problem (does not matter too much)</p> <p>2 – minor problem (would be nice to solve, but not a high priority)</p> <p>3 – major problem (a problem that should definitely be solved with high priority)</p> <p>4 – usability catastrophe (a problem that renders the Web site / app useless)</p> <p>Justification (Frequency, Impact, Persistence):</p> <p><i>Frequency: medium</i></p> <p><i>Impact: high</i></p> <p><i>Persistence: low</i></p> <p><i>How I weighted the factors:</i></p> <p>It's a summary page – users might meet this page before their check out. By using this interface users can check their decision again and change if they had make a mistake.</p>	

Possible solution and/or trade-offs:

Since the screenshot is showing a summary about Hotel reservation, it shows a price per night but doesn't show how many nights customers have choosed. This could make customers hard to check it correctly

Relationships:

No. HYK-07	Problem/Good Aspect Problem
Name: Using different name?	
<p>Evidence:</p> <p>See the screenshots below. It shows that the service uses two different button name for same function. This can confuse customers if they are using same functions.</p>  <p>Interface aspect:</p> <p>A button to add for purchase</p> <p>Heuristic: Consistency and standards</p>	
<p>Explanation:</p> <p>“Consistency and standards” : This heuristic violates because it is using different name on a same function button. It can make users wonder it’s doing different functions. The system should be consistent.</p>	
<p>Severity or Benefit:</p> <p>Rating: <choose one of the following:></p> <p>NA – good aspect</p> <p>1 – cosmetic problem (does not matter too much)</p> <p>2 – minor problem (would be nice to solve, but not a high priority)</p> <p>3 – major problem (a problem that should definitely be solved with high priority)</p> <p>4 – usability catastrophe (a problem that renders the Web site / app useless)</p> <p>Justification (Frequency, Impact, Persistence):</p> <p><i>Frequency: medium</i></p> <p><i>Impact: medium</i></p> <p><i>Persistence: low</i></p> <p><i>How I weighted the factors:</i></p>	

This interface doesn't need to be shown in every process, but to book a travel, users have to click on it. This button is not a major function, so the users cannot even notice the system is using different names on a same function button.

Possible solution and/or trade-offs:

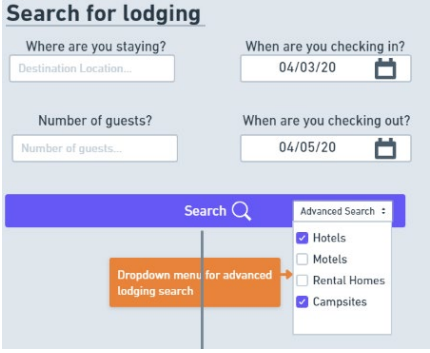
The button's names should be unified to be consistent.

Relationships:

No. HYK-08	Problem/Good Aspect Problem
Name: Useless ‘Advanced Search’	
Evidence: See the screenshot below. To search for rental vehicles, they don’t have any options for ‘Advanced search’. So that button is useless in this section. <div data-bbox="218 443 646 698" data-label="Image"> </div>	
Interface aspect: The option for advanced search looks like duplicated with original search button.	
Heuristic: User control and freedom	
Explanation: “User control and freedom” : This heuristic violates because it’s hard to figure out what’s ‘Advanced Search’ button is for. It doesn’t show up any options.	
Severity or Benefit: Rating: <choose one of the following:> NA – good aspect 1 – cosmetic problem (does not matter too much) 2 – minor problem (would be nice to solve, but not a high priority) 3 – major problem (a problem that should definitely be solved with high priority) 4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence): <i>Frequency: low</i> <i>Impact: low</i> <i>Persistence: low</i> <i>How I weighted the factors:</i> Since not all the users would use a rental vehicle option, it’s impact and frequency would be low.	
Possible solution and/or trade-offs: The useless ‘Advanced Search’ button makes users confused. To solve this problem, a advanced search	

button should show up separately or deleted.

Relationships:

No. HYK-09	Problem/Good Aspect Problem
Name: Button in a Button	
Evidence:	
See the screenshot below. It provides advanced search but it looks a little bit complicated because the advanced search option is inside of search button.	
	
Heuristic: Consistency and standards, Aesthetic and minimalist design	
Explanation:	
“Consistency and standards” : This heuristic is violated because users could be confused and make mistakes when they meet a button is inside of the other button.	
“Aesthetic and minimalsit design” : This heuristic is violated because the button design isn’t minimalist design.	
Severity or Benefit:	
Rating: <choose one of the following:>	
NA – good aspect	
1 – cosmetic problem (does not matter too much)	
2 – minor problem (would be nice to solve, but not a high priority)	
3 – major problem (a problem that should definitely be solved with high priority)	
4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence):	
<i>Frequency:</i> medium	
<i>Impact:</i> high	
<i>Persistence:</i> low	
<i>How I weighted the factors:</i>	
This interface is for searching for lodging. There would some users who doesn’t use this page, but if there is a user using it, it would impact them because it’s easy to skip to set a advanced search	

option. But if user keep uses this system, they might be used to it.

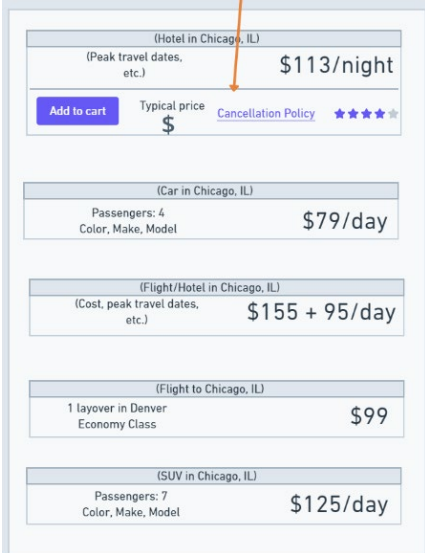
Possible solution and/or trade-offs:

The interface could make users skip or confused due to it's weird location. To solve this problem, the location of button 'Advaned Search' should change right upper side of 'Search' button.

Relationships:

No. HYK-10	Problem/Good Aspect Problem
Name: Need to make user double-check their typing	
Evidence: See the screenshot below. It should re-check if the confirmation email is right, but there is nothing to prevent an error. <div data-bbox="218 445 647 591" data-label="Image"> A screenshot of a web form titled 'Confirmation Email'. It features a single text input field with the email address 'bettercallsaul@gs.net' typed into it. There is a small orange star icon to the left of the input field. The entire form is enclosed in a light blue border. </div>	
Heuristic : Help users recover form errors, Recognition rather than recall.	
Explanation: “Help users recover form errors” : This heuristic is violated because as users have to type a confirmation email by themselves, there could be a mistake during typing. So the system has to double check it by asking a confirmation email again to the user, but it’s missing. “Recognition rather than recall” : This heuristic is violated because by using customer’s information (if they are not using a guest ID), the system would be able to auto-fill the confirmation email.	
Severity or Benefit: Rating: <choose one of the following:> NA – good aspect 1 – cosmetic problem (does not matter too much) 2 – minor problem (would be nice to solve, but not a high priority) 3 – major problem (a problem that should definitely be solved with high priority) 4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence): <i>Frequency: low</i> <i>Impact: high</i> <i>Persistence: high</i> <i>How I weighted the factors:</i> Since this interface is shown at the end of purchase page, it’s not that frequent. But having error prevention is very important and if it didn’t, users can have some troubles.	
Possible solution and/or trade-offs: The interface should include a another textbox to check if user has typed same confirmation email as they did on upper textbox. Or it could provide a auto-fill by using user’s information.	

Relationships:

No. HYK-11	Problem/Good Aspect Problem
Name: Too complicated to recognize each category	
Evidence: <p>See the screenshot below. In this screenshot, it's hard to recognize which price indicates categories.</p>  <p>The screenshot shows a list of travel options with the following details:</p> <ul style="list-style-type: none"> Hotel in Chicago, IL: (Peak travel dates, etc.) \$113/night. Includes an 'Add to cart' button, 'Typical price \$', 'Cancellation Policy', and a 4-star rating. Car in Chicago, IL: Passengers: 4, Color, Make, Model \$79/day Flight/Hotel in Chicago, IL: (Cost, peak travel dates, etc.) \$155 + 95/day Flight to Chicago, IL: 1 layover in Denver, Economy Class \$99 SUV in Chicago, IL: Passengers: 7, Color, Make, Model \$125/day 	
Heuristic: Aesthetic and minimalist design, Consistency and standards,	
Explanation: <p>“Aesthetic and minimalist design” : This heuristic violates because it’s hard to find out relative categories at a glance. Each different categories are using same layouts.</p> <p>“Consistency and standards” : This heuristic violates because users can wonder whether the price belongs to the category they were looking for.</p>	
Severity or Benefit: <p>Rating: <choose one of the following:></p> <p>NA – good aspect</p> <p>1 – cosmetic problem (does not matter too much)</p> <p>2 – minor problem (would be nice to solve, but not a high priority)</p> <p>3 – major problem (a problem that should definitely be solved with high priority)</p> <p>4 – usability catastrophe (a problem that renders the Web site / app useless)</p> <p>Justification (Frequency, Impact, Persistence):</p> <p><i>Frequency: low</i></p>	

*Impact: **high***

*Persistence: **high***

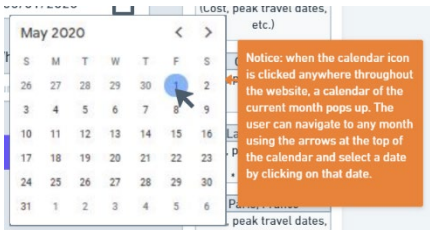
How I weighted the factors:

Since the platform is mainly about searching, but only some user would meet this interface due to its categorized tabs. However, the user who faces this interface, it would be frustrated because it's all messed up without distinction of each categories. This might make users overwhelmed and feel tired to search.

Possible solution and/or trade-offs:

The interface makes user confused to find out exact informationt they want and tired. To solve this problem, search result should be divided with each category using different layouts/blocks.

Relationships:

No. HYK-12	Problem/Good Aspect Problem
Name: Hard to choose other months	
Evidence : See the screenshot below. Using a calendar is a good way, but it's too small to check the future dates. It should provide at least next month of current month. 	
Heuristic: Consistency and standards	
Explanation: “Consistency and standards” : This heuristic is violated because it makes hard to select future dates. Because of that, many travel booking sites provides 3-month calendar for selection.	
Severity or Benefit: Rating: <choose one of the following:> NA – good aspect 1 – cosmetic problem (does not matter too much) 2 – minor problem (would be nice to solve, but not a high priority) 3 – major problem (a problem that should definitely be solved with high priority) 4 – usability catastrophe (a problem that renders the Web site / app useless)	
Justification (Frequency, Impact, Persistence): <i>Frequency: medium</i> <i>Impact: low</i> <i>Persistence: low</i> <i>How I weighted the factors:</i> Every users might use this date select calendar, but users can still use this service with one-month calendar	
Possible solution and/or trade-offs: If a user wants to view further dates, it would take a lot efforts to view it. To solve this problem, like other travel booking sites provides, the site can provide 3-month calendar instead of 1-month calendar.	
Relationships: HYK-01	

Final conclusion

Summarize of the problems :

This wireframe is very well-designed with several minor problems. But it does have some major problems, hard to prevent an error when user mistypes confirmation email address, lack of sorting option and lack of auto-filled payment. Since the service mainly provides searching items for user, sorting option is essential; it can easily find on other competing sites. Also, it's a good idea to ask for confirmation email address to send a summary of their purchase, but this should be double-checked for error prevention because user can do some mistakes within. The last major problem is about auto-filled payment information. Many of peoples don't memorize all of their payment information. Since every user who wants to make a purchase via this service should type the payment information, providing a autofill would be great for users. It's also provided from other compatible sites.

Summarize of the recommendations for redesign :

As I mentioned before, this is a well-designed wireframe. Though, for user convenience, in a results page, if it shows more than one category it should be divided by a different box to recognize it easier which category this item is. The rate of the item is also one of a major criterion, it should be shown in the same location of the price. Adding these functions, it would be competitive enough to similar services.

Reflection

How many problems did you find?

I've found total nine problems in this wireframe, but most of the problems are very cosmetic. Missing scroll option! -needs one more, auto-fill / absence of save card information option, absence of sorting option, using different name, useless 'Advanced Search', button in a button, need to make user double-check their typing, too complicated to recognize each category, hard to choose other months.

Do you think you found all problems, or were there more beyond the ones you reported?

I've tried to find all problems from the wireframe, but more problems can appear according to one's point of view.

How much time did you spend on the evaluation and on finalizing the UARs?

I spend about 6.5 hours to evaluation(2.5) and finalizing(4) the UARs. It took a lot to find problems at first.

Do you think Nielsen's Heuristics are a good tool for finding usability problems? Why (not)?

I personally think Nielsen's Heuristics are a very good tool for finding usability problems. It is well categorized, and it has established standards for essential functions.

Do you think that UARs are a good way to describe usability problems? Is there anything you would add to the UAR template to make it better?

UARs are a good way to describe usability problems. It offers exact evidence with the heuristics, explanations, possible solutions and/or trade-offs in a organized template – which includes every aspects that should be provided to developers to recognize the problems.

How would you improve the process if you had to do the same assignment again?

I've spent a lot of time to find a problem from the wireframe because I could recognize only confirmation email address problem and sorting option problems at first. After looking for more heuristic examples I found more cosmetic problems, but I had to rush my work due to the deadline. If I have to do the same assignment again, I would summarize heuristics with a photo of examples in a paper to help myself not to confuse with the heuristics.