UpGrad

Summary

Usability Testing Techniques

As a PM, you should be aware of different usability techniques which are prevalent in the industry. These techniques would help you assess the usability of a product. Usability is the ease with which the user uses your product. Conducting these techniques would help in understanding how your users are actually using the product, what they think and feel about the product, and the features which they are interested in.

Quick and Dirty Usability Testing

The quick and dirty method is used to do quick validation of product designs so that your team can move onto the next steps of product development. Typically, a session would last somewhere between 10 to 15 minutes and it's done at the early stage of product development. For conducting this technique, you don't have to recruit real users, rather you test the designs with the available resources. This technique is not meant for expert users.

An example of quick and dirty technique was covered in the 'Prototyping' module where Romil and Anshumani took us through the example of paper and pen prototype of a phone number based login screen. The objective was to figure out whether this screen would work or not. You can continuously iterate on the product design based on the rapid feedback you receive during the session.

Heuristic Evaluation

Heuristic evaluation is done by UX experts generally on working prototypes and is helpful for competitive benchmarking. Remember that in this case you are not doing the testing with users. UX experts will apply the heuristics given by Nielsen Norman on the product and catch UX violations.

Heuristic evaluation is carried out by UX experts by comparing the product interface (website/app/prototypes) with accepted usability principles given by Nielsen Norman. In the session, you saw the application of these principles on the Zomato website. These usability heuristic principles are as follows:

- Visibility of system status: The system should always keep the user informed about what's going on through appropriate feedback within reasonable time.
- Match between the system and the real world: The system should speak user's language rather than system-oriented terms.
- User control and freedom: User often chooses system functions by mistake and will need a clearly marked emergency exit to leave the unwanted state.
- Consistency and standards: The user should not have to wonder whether different words, situations or actions means the same thing. The same action should lead to similar results.

- Error prevention: Either eliminate error-prone conditions or check for them and present the user with a confirmation option before they commit to the action. Guide the user to prevent errors.
- Recognition rather than recall: Minimise the user's memory load by making objects, actions and options visible. The user should not have to remember information from one part of the dialogue to another.
- Flexibility and efficiency of use: Make the system efficient by using accelerators. Accelerators are complex functions which are achieved by a click or two.
- Aesthetics and minimal design. Dialogue should not contain information which is irrelevant or rarely needed. Visually, functions and information should be easy to find and identify. Also, a similar set of actions should have a visual resemblance and differentiability.
- Help the user recognise, diagnose and recover from errors: Error messages should be expressed in plain language; no codes precisely indicate the problem and constructively suggest a solution.
- Help and documentation: Provide help and documentation wherever necessary. Small tips and help text can help to enhance the efficiency of a system, guiding users through what needs to be done to achieve the desired output.

Next, we took Zomato (and a few other websites) as an example to explain the heuristics given by Nielsen Norman.

Heuristic	**Meaning of Heuristic**	**Example**
Visibility of system status	System should always keep the user informed about what's going on through appropriate feedback within reasonable time.	In Zomato, the system automatically fetches your location and it shows the status next to the search bar that your location has been detected. Having a prefilled location here makes food ordering simpler. In Flipkart, when you scroll over any link it changes the color to blue and shows you the status that it is clickable.
Match between the system and the real world	System should speak the user's language rather than system-oriented terms.	In Zomato, the delivery icon may not match the user's perception of the delivery icon. A better representation of delivery icon could be a vehicle which actually does delivery rather than showing the content of the food item.
User control and freedom	The user often chooses system functions by mistake and will need a clearly marked emergency exit to	When viewing images of a restaurant in Zomato, it takes the user to a gallery view and a cross button on the top right corner

	leave the unwanted state.	clearly marks the exit function.
Consistency and standards	The user should not have to wonder whether different words, situations or actions means the same thing. The same action should lead to similar results.	Zomato has maintained consistency in terms of content visible for each listing, i.e. name, location, cuisine, rating.
Error prevention	Either eliminate error-prone conditions or check for them and present user with a confirmation option before they commit to the action. Guide the user to prevent errors.	The search bar in Zomato shows you what needs to be searched hence eliminating the possibility of having an ambiguous search result.
Recognition rather than recall	Minimise the user's memory load by making objects, actions and options visible. The user should not have to remember information from one part of the dialogue to another.	In Zomato, images of pizzas very clearly indicate what the restaurant serves, making the choice of ordering easier.
Flexibility and efficiency of use	Make the system efficient by using accelerators. Accelerators are complex functions which are achieved by a click or two.	The 'Share' button on Zomato helps you share with friends instantly, hence enhancing its efficiency. Having prefilled addresses in payment options also enhances the efficiency of the ordering process.
Aesthetics and minimal design	Dialogue should not contain information which is irrelevant or rarely needed. Visually, functions and information should be easy to find and identify. Also similar set of actions should have a visual resemblance and differentiability.	In Zomato, 'Order Food' and 'Book a Table', being two primary actions, are given equal form as a button but are still differentiable by colour.
Help the user recognise, diagnose and recover from errors	Error messages should be expressed in plain language; no codes precisely indicate the problem and constructively suggest a solution.	If the user fills in wrong text or a text which the system cannot process, it shows the error instantly. In Zomato, if you book a table for 21 guests, it clearly shows the alert notification which states that the choices made by the user cannot be handled by the system and thus prevents error.

Help and documentation	Provide help and documentation wherever necessary. Small tips and help text can help to enhance the efficiency of system, guiding users through what needs to be done to achieve the desired output.	The text box, giving the details on how to book a table at a restaurant on Zomato, solves the problem for most of the users. The 'How it works' help text helps the user understand how the table booking function works.

One-on-One Interview

One-on-one interviews help in figuring out what the users think and feel about your product or your competitor's product. It is also a great technique to get direct user feedback on the product.

This can be conducted at different stages of product development. The interview duration is between 15 to 60 minutes depending on the type of questions you have and what it is that you are trying to validate.

In the lecture, we conducted a usability interview of the Uber app. The following insights were gathered from the interview:

1. The user uses Uber everyday for shorter commutes
2. A mismatch between ETA and actual time of arrival is a problem
3. The user has to initiate the call to explain location
4. Drivers are not able to reach obscure locations
5. Paying via Paytm is convenient for user
6. Regular Paytm user
7. Upon cancellation of a booking, immediate user feedback is not taken and money is deducted from the user's account
8. Refund process online and convenient
9. Discoverability of promotional campaigns might be a problem
10. Discoverability of split fare feature might be a problem

Heatmap Testing

Heatmap is a form of visual analytics in which colours are used to represent relationship between data values. Visual analytics looks at the transitions within a particular page whereas traditional analytics focuses on transition between pages.

Heatmap testing will help you figure out:

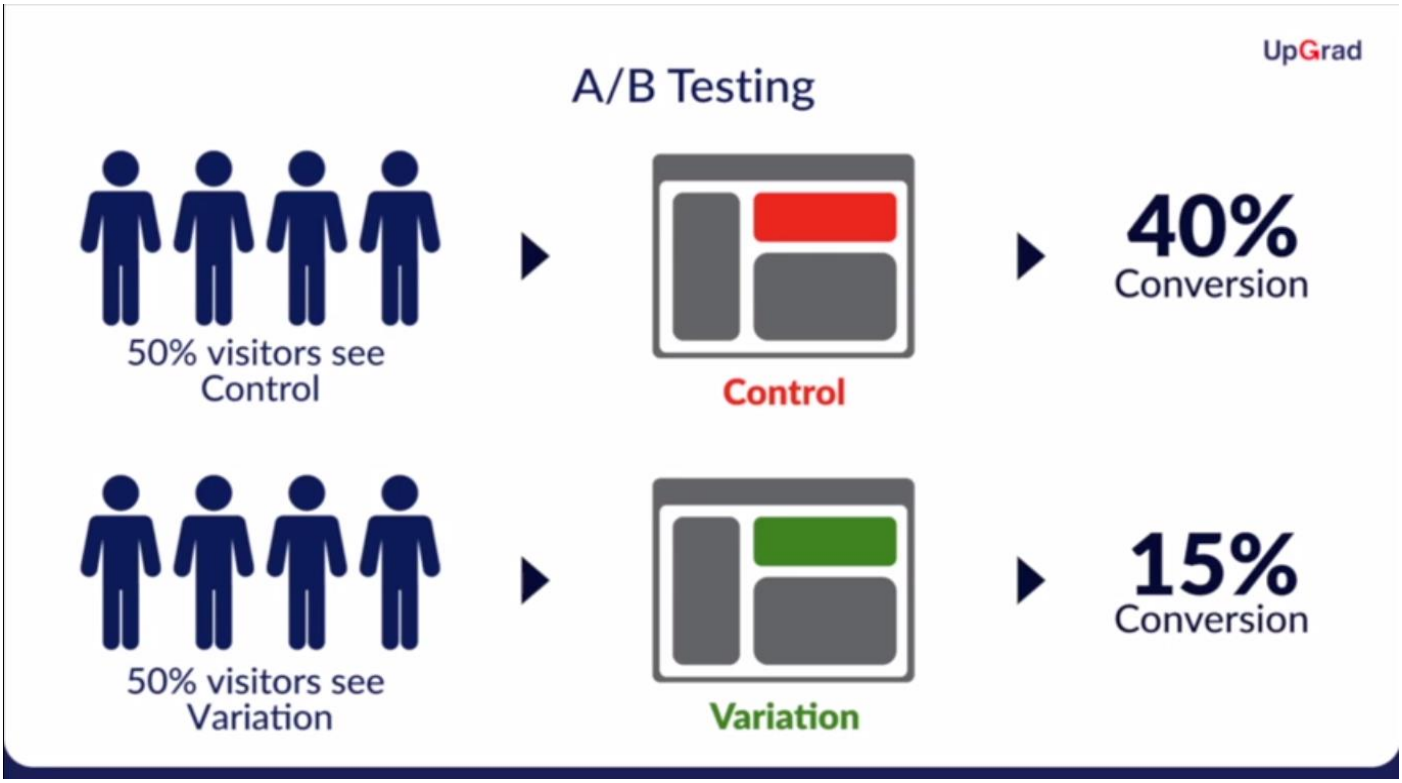
- Where users have clicked on a particular page
- Where their eye is focusing on
- Which elements are they more interested in on the page
- How long it takes a user to complete the task; e.g. a sample task could be finding some content on a page
- Overall visual hierarchy of the product

For example, if you are in an ecommerce company, heatmap testing would let you know whether the user is actually spending time looking at the price simply by either tracking their eye movement or tracking cursor movement. And this would tell you whether the kind of font size or the kind of placement of the price point is correct or not. Heatmap also helps in identifying the placement of filters which could be horizontal, as in the cases of Swiggy and BookMyShow, or they could be vertical, as in the cases of Practo and Zomato.

Heatmap testing is a data-driven technique and is typically carried out at a middle to later stage of product development. It can be performed using tools such as Crazyegg, Clicktale, and Luckyorange.
We also looked at how BookMyShow used heatmap testing to make changes to its website and app.
First, they changed the date dropdown to show today's date as they realised that most of their users were just picking today's date. They also changed the placement of global navigation bar based on the data received from heatmap testing.

A/B Testing

A/B testing helps you compare two different versions of a product against each other to determine which one performs better. The original version is known as the 'control' and the modified version is called the 'variant'. An A/B test can be conducted from right to the very early stage of a product to a very advanced late stage product. A/B testing can be performed using tools such as Unbounce, Optimizely, and Google Analytics.



A general framework for conducting A/B testing is as follows:

- Identify metrics
- Create different variations
- Run A/B testing
- Measure the metrics
- Analyse the collected data

BookMyShow used A/B testing to show changing of show times on the seat layout so that a user has the flexibility of picking a previous show time and the next show time on the same screen, without having to go back. Another A/B test was done on the order summary screen of the BookMyShow app. Earlier, the app used to show the offer screen to the user and he/she had to hit skip offers, which most of the users were doing. Through analysis of A/B testing data, BookMyshow changed the positioning of 'Avail offers' option.

Usability Testing of BookMyShow

Usability test is conducted on BookMyShow website using Lookback recording tool, which captures the user's expressions.

In this usability test, the user was trying to book movie tickets on BookMyShow website. Following insights were gathered from the usability test:

- Coupons were misunderstood as advertisements
- The user was not looking for offers
- The user knew about m-tickets. So, discoverability of m-ticket is not a problem.

You should be able to:

- Understand different usability techniques used in the industry
- Identify the reasons for conducting a particular usability technique
- Conduct a usability testing technique