

## What is User Experience?

A discipline focused on designing the end-to-end experience of a product. To design an experience means to plan and act upon a certain set of actions, which should result in a planned change in the behaviour of a target group when interacting with a product.

UX work should always be derived from users problems and aim at finding a pleasurable, seductive, inspiring solution. The results of that work should always be measurable through metrics describing user behaviour.

When you're designing an experience, you are in fact planning a change in the behaviour of your target group. You've found out the users problem and you're trying to destroy the burden using design methods.

User experience lies at the crossroads of art and science and requires both extremely acute analytical thinking and creativity.

- > What is our product(s)? (Service Desk, Management Suite, etc)
- > What do we offer, and how is the product used?
- > Who is our target audience? (End Users, Managers, Analysts, etc)
- > What is our competitive advantage (What makes our product stand out from our competitors)?
- > What do we want our users to do? (action items)
- > Does it meet the UX Hierarchy of needs? (Purpose, Function, Understandable, Ease of use, Enjoyable)
  - Functionality - Interface works and does not break?
  - Information - Interface has correct, readable and up-to-date content?
  - Aesthetics - Interface looks unique, friendly and professional?
  - Usability - Interface allows users to complete tasks quickly and accurately?
- > What are our Basic Scenarios (provide a picture (Narratives, Storyboard, etc) of the current/intended user experience)?
- > Have we simulated the designs through the main features (Prototyping include complex interactions, new functionality and changes in workflow, technology or design)?
- > What is the potential problems of your future users?

## Usability Heuristics

- > Visibility of system status: Keep users informed about what is going on, through appropriate feedback within reasonable time.
- > Match between system and the real world: Speak the users' language, with words, phrases and concepts familiar to the user. Making information appear in a natural and logical order.
- > User control and freedom: Support undo and redo. Users often choose functions by mistake and will need a clear "Exit".
- > Consistency and standards: Follow conventions. Users shouldn't wonder if different words, situations, or actions mean the same.
- > Error prevention: Even better than good error messages is a careful design which prevents a problem from occurring. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.
- > Recognition rather than recall: Minimise the user's memory. Make objects, actions, and options visible. Instructions for use of the system should be visible or easily retrievable whenever appropriate.
- > Flexibility and efficiency of use: Accelerators (unseen by the novice user) may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users.
- > Aesthetic and minimalist design: Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.
- > Help users 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: Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation.

## User Testing

- > You are not the user (STOP guessing. YOU have to check everything!)
- > Validate your designs/prototypes/ideas with 5-10 Users at least.
- > Use a UX Agile Process

## Basic Principles

- > People Don't read. They scan.
- > Present fewer choices to the User.
- > Stay out of the users way (Don't tell the user what to do next, allow the user to test the application/process/functionality themselves in a real environment).
- > Great experience is about control (the user having the control).
- > Be simple and clear. (Make it easy for the user to understand what they are doing or what you want from the user).
- > "Call-to-action's" should be clear.
- > Engagement should be fun. Should be intuitive, responsive and clear to understand.
- > Great experiences are Simple.
- > Be consistent (In everything).
- > Start from the basics. Use hierarchy (Overview to detail).
- > Users come first.
- > Users don't like to lose control or lost.
- > Does your design address your target group?
- > Does it resolve their real problem?
- > Is your solution an accurate response to your target group's needs?
- > Back up your design assumptions with knowledge to minimise the risk of failure.

YouTube, Airbnb, Flipboard, Square, Pinterest, Etsy, Path, AboutMe, Slideshare - All these well designed, successful products were co-founded by designers.

## Questions for your users

- > Can the user navigate naturally through the prototype/designs?
- > Does the user understand how to start and complete tasks?
- > Does the user understand the terminology used in the prototype/designs?
- > Can the user discover features?
- > Does the user interpret features in the way intended?
- > Does the prototype/designs give users the content they need on a page/screen/form?
- > What does the user like about the prototype/designs? What are the user's main points?
- > What is the user's overall impression of the layout?