CASE & OPGAVER

FORELÆSNINGSGANG 2 - SYSTEMATISK DESIGN AF BRUGERGRÆNSEFLADER, F2020

CASE

Øvelsen omhandler de følgende 2 UIs:

- Asos webshop: https://www.asos.com
- Mofibos lyd- og e-bogsunivers: https://mofibo.com/dk

I kan både arbejde individuelt og i grupper (grupper anbefales - gerne 2 og 2). Hvis I vil have feedback på denne valgfrie øvelse, så skal besvarelserne uploades til LearnIT inden fredag, den 7. februar kl. 23.59.

OPGAVE 1

Vælg 3 sider i hvert UI og lav en usability analyse ud fra Jakob Nielsens 10 regler (se bilag). Skriv jeres vurdering til hvert punkt.
Vælg 3 sider i hvert UI og lav Steve Krugs trunk test (se bilag). Skriv jeres vurdering til hvert punkt.

OPGAVE 2

Definer hvilken navigationstruktur og brug/typer af grupperinger UI'et bruger (taksonomi).

OPGAVE 3 - BONUSOPGAVE

Lav en card sorting-øvelse med de 10 sider, som i vurderer er vigtigst i hvert UI, og sammenlign det med den nuværende struktur.
http://www.usabilityfirst.com/usability-methods/card-sorting/

BILAG

FORELÆSNINGSGANG 2 - SYSTEMATISK DESIGN AF BRUGERGRÆNSEFLADER, F2020

JAKOB NIELSEN

https://www.nngroup.com/articles/ten-usability-heuristics/

1. Visibility of system status
The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

2. Match between system and the real world
The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

3. User control and freedom
Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

4. Consistency and standards
Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

5. Error prevention
Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

6. Recognition rather than recall

Minimize 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. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

7. 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. Allow users to tailor frequent actions.

8. 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.

9. Help users recognize, diagnose, and recover from errors
Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

10. 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. Any such information should be easy to

STEVE KRUG

What site is this? (is there a site ID?)
What page am I on? (is there a page name?)
What are the main sections? (are there section names?)
What navigation options do I have here? (is there local navigation?)
Where am I overall? (are there "you are here”-indicators?)