Interaction Design Supervision 2

Neelu Saraswatibhatla (srns2) - CST IA

1.
    a. Heuristic evaluation is used to find usability problems in a user interface design. A set of evaluators check the UI against ten heuristics and identify violations of these heuristics, along with the severity of the violations. They compile these as a set of recommendations to the designer.

       The severity is rated on a scale of 1 - 4:
       1 - cosmetic problem
       2 - minor usability problem
       3 - major usability problem
       4 - usability catastrophe

       The 10 heuristics are:
        i. Visibility of system status
        ii. Match between system and real world
        iii. User control and freedom
        iv. Consistency and standards
        v. Error prevention
        vi. Recognition rather than recall
        vii. Flexibility and efficiency of use
        viii. Aesthetic and minimalist design
        ix. Help users recognise and recover from errors
        x. Help and documentation
    b.
        i. This is a good design, as only one of the heuristics (10) is violated.
            1. Visibility of system status - applies, as the current selected tab is clearly shown as the active one
            2. Match between system and real world - applies, as real world folders have tabs, much like this window
            3. User control and freedom - applies, as the user has maximum control given that this is an options window
            4. Consistency and standards - applies, as the window bar looks like every Windows XP window bar, and the tabs match the OS styling, etc.
            5. Error prevention - applies, as there are discrete choices for options which the user selects from rather than the user specifying one
            6. Recognition rather than recall - applies as most of the options have non-tech related meanings as well, such as directories and workspace, so users will recognise these terms
            7. Flexibility and efficiency of use - applies, as it is easy to modify and add options and integrate them into the design just by making these changes

8. Aesthetic and minimalist design - applies, as the only features are the tabs and the labels, so there isn't unnecessary clutter
9. Help users recognise and recover from errors - not applicable, as there can't be errors in a tab bar
10. Help and documentation - violated, as there is no explanation for what each options tab represents [Severity rating 2 - the options are quite self-explanatory and documentation isn't necessarily required]

ii. This is a good design but requires some changes, as three heuristics are violated (2, 6, and 10).

1. Visibility of system status - applies, as the current selected tab is clearly shown as the active one with a blue background
2. Match between system and real world - violated as most of the words (e.g. Source, Compile, etc.) don't mean much in this context outside of programming [Severity rating 2 - given that this appears to be a code editor of some sort, the user likely has a fair amount of knowledge so this likely isn't a huge problem]
3. User control and freedom - applies, as the user has control to select whichever option they require
4. Consistency and standards - applies, as once again, the window bar looks like every Windows XP window bar, and the tabs match the OS styling, etc.
5. Error prevention - applies, as again, there are discrete choices for options which the user selects from rather than the user specifying one
6. Recognition rather than recall - Violated, as similarly to heuristic 2, some of these words are quite specific [Severity rating 2 - same reason as in heuristic 2]
7. Flexibility and efficiency of use - applies, as again, it is easy to modify and add options and integrate them into the design just by making these changes
8. Aesthetic and minimalist design - applies, as the only features are the tabs and the labels, so there isn't unnecessary clutter
9. Help users recognise and recover from errors - not applicable, as there can't be errors a menubar
10. Help and documentation - violated, as there is no explanation for what each options tab represents [Severity rating 3 - given that these are quite specific and possible unrecognisable terms, documentation would be very useful]

2.

a.

i. Problem: Gathering sufficient data
Solution: Ask as many target stakeholders as possible

ii. Problem: Managers may have different requirements to employees who are actually using the system
Solution: Involve many different stakeholders (including these employees) in the requirements gathering process

iii. Problem: Many people may be gathering data so collaboration could be a problem
Solution: Use a platform such as Google Drive (or even Git for some purposes)

b.

i. Figure-ground relationship - The object in focus is different to the background - The foreground and background are clearly marked, as the foreground is a gradient of greys and the background is white with blue at the top
ii. Proximity - group by location - Each icon has its label close to it (right underneath it)
iii. Similarity - group by type - The menu options are all blue icons, while the menubar options are white icons
iv. Symmetry - equal margins, etc. - The menu is symmetric
v. Continuity - group by alignment - the menu items are aligned in the square
vi. Closure - perception of shapes that aren't completely there - Not used here

c.

i. P1 - Consistency and standards, Aesthetic and minimalist design
ii. P2 - Match between system and real world, Error prevention, Recognition rather than recall
iii. P4 - Help and documentation
iv. P5 - Consistency and standards
v. P7 - Consistency and standards, Help and documentation, Recognition rather than recall, Help users recognise and recover from errors