User interface prototypes

Davide Falessi

Introduction (1/2)

- The purpose of a UI prototype is to describe the **manner** in which the human user will interact with the software system.
- This helps gain an **early** reaction from users on the concepts proposed for the application.

Introduction (2/2)

- The prototype depict **every** window, dialog, menu, button, error message, etc. in the system.
- Appropriate annotations describe any aspects which are not obvious.
- The prototype should be **self-explanatory**.
  - The prototype is incomplete if the author has to explain how particular aspects will operate.

User Interface Design Criteria

- **Visibility of system status.** Users should always know where they are and what's going on.
- **Real world - system match.** The system should mirror the real world of the user as much as possible. Use language, concepts, etc. that are familiar to the user. Order the processes/screens in a way that is meaningful and logical to the user.
- **Control and freedom.** Don't "trap" the user. Support clearly marked exit, undo, and redo functions. Don't force them into a long linear sequence of operations with no escape.

User Interface Design Criteria

- **Recognition not recall.** Provide visual objects, actions, and options (e.g. cue cards) to assist the user for navigation and input activities. Don't expect they will memorize commands.
- **Flexibility and efficiency of use.** Accelerators (unseen by novice users) can speed up interaction for expert users. Allow users to customize frequent actions whenever possible.
- **Aesthetic and minimalist design.** Visibility of rarely needed information should be avoided. The more information that appears on the screen, the less visible each unit of information becomes.

User Interface Design Criteria

- **Online help and additional documentation.** Though a well designed system can be used without documentation and help, supplemental information may still be necessary. Keep this information tied to user tasks, support easy to use search functions, and don't make this section too large.
- **Effective error handling.** Assist users to recognize, diagnose, and recover from errors. Don't just tell them there's an error, suggest corrective action whenever possible.
- **Error prevention.** A design that prevents errors from occurring is better than a good error message.

Consistency and standards

- **Interface.** Is the "look and feel" of the interface consistent? Is every screen recognizable as your product? For example, is there a common background and/or color scheme for all screens? Common screen layouts?
- **Function/Appearance.** Do all object that appear the same, function the same? E.g., are Help, Exit, and Search objects in the same spot on all screens? Do they have the same design, color, etc.?
- **Text characteristics.** Are the text characteristics constant from screen to screen. E.g., is Arial 14 point bold italics always used only for chapter titles? Does blue underlined text always represent a hyperlink?

Consistency and standards

- **Semantic characteristics.** Are metaphors and icons used consistently throughout the interface? E.g., does a magnifying glass mean the same thing every place it is used? Does a green cat always mean Help?
- **Navigation.** Are navigation objects and steps consistent throughout the interface? Are screens linked consistently? E.g., Previous/Next buttons, etc.
- **Interaction tools.** Are interaction tools like mouse pointers, touch screens, joysticks, used consistently?

Consistency and standards

- **Conventions.** Are conventions familiar to the user employed consistently?
- **Screen configuration(s).** Are all related items on the screen grouped together visually in a format that makes sense?
- **Labels.** Are labels on buttons, menus, and titles used consistently?

Storyboard

- A storyboard is a sequence of illustrations displayed in sequence.

- Examples…

Wireframe

- It is a GUI prototype specifically addressing WEB applications.
- Wireframes are created for the purpose of arranging elements to best accomplish a particular purpose.
- The wireframe depicts the page layout or arrangement of the website's content, including interface elements and navigational systems, and how they work together

Wireframes vs. Storyboards

- The Wireframe is more about the system (how)
- The Storyboard is more about the user experience (what)
- The process can be iterative and it is possible to have a refined version of the storyboard that reports high graphical details after having a wireframe

Possible Tools

- **Figma.com**
- **NinjaMock**
- **Draw.io**
- Moqup
- Balsamiq
- FluidUI
- JustInMind
  Mockflow
- Framebox
- Pencil