Electronics and
Computer Science

University of
Southampton

UI Design

COMP6226: Software Modelling Tools and Techniques for Critical Systems

Dr A. Rezazadeh (Reza)
Email: ra3@ecs.soton.ac.uk or ar4k06@soton.ac.uk

November 24

Overview

- What is User Interface Design?
- Why is UI design important?
- Types of User Interface Design
- User Interface Design – Some Principles/Heuristics
- What’s the Difference Between UX and UI?
- ICONIX Process – UI Design as a starting point
- UI- Different Layers
- Sitemap
- Wireframing
- The Value of Wireframes

Objectives

- Understand the role and importance of UI Design
- Make sense of different kinds of UI
- List UI principals/heuristics and apply them
- Differentiate between UI and UX design
- Understand and use different approaches in UI design
- Use different tools such as Sitemap and Wireframe for UI design

What is User Interface Design?

- An interface is a medium that humans use to interact with a software, system, a computer, or a machine.
- User interface design refers to the set of design elements that maximise effective interaction and create pleasurable user experiences.
  - The best User Interface (UI) clearly articulates what is important.
- The end goal of any UI design is to make it user-centric so that the interactions are always transparent and efficient.
- UI is a major driving factor behind the success of different software systems.
  - Before modern UI design, interacting with computers required fluency with program or machine languages.

Why is UI design important?

- Good design can improve efficiency and productivity.
  - A well-designed UI facilitates effective interaction between the user and the program/app, through effective visuals, clean design and responsiveness.
  - It will boost customer retention and satisfaction
    - Users often judge a system by its interface rather than its functionality
  - Increases usability and accessibility
  - User interface is important to meet user expectations and support the effective functionality of your application.

Why is UI design important?

- UI should be designed to match the skills, experience and expectations of its anticipated users.
  - A poorly designed interface can cause a user to make catastrophic errors
- Poor UI design is the reason why so many software systems are never used
  - The most elegant, efficient, and high-quality architectural and detailed designs can be failed by a poor UI design.

Types of User Interface Design

- **Command-Line Interfaces CLI:** Requires a user to type in commands from a list of allowable commands
- **Graphical User Interface (GUI):** An interface that provides users with different visual elements to interact with
- **Voice-Controlled User Interface (VUI):** This works on voice or audio commands.
  - Smart assistants like Siri, Alexa, and Cortana are well-known examples of VUIs.
- **Gesture-Based Interfaces:** Bodily gestures, movements, and signals comprise the primary mode of interaction
  - As augmented reality (AR) and virtual reality (VR) grow in popularity, gesture-based interfaces become more important than ever. Gesture-based UI translate a user's motions in a 3D space into commands.

User Interface Design – Some Principles/Heuristics

1. Enhance user control: Navigating the interface should focus on making learning and adopting processes easier for the user.
2. Make actions reversible: Users should be able to backtrack their specific tasks without the fear of failure.
   - This boosts user confidence.
3. Navigation should always be easy: Easy navigation and option predictability ensure a clear and evident exploration of an interface.
4. Informative feedback is a must: Acknowledging any action in the form of colour change or sound prompts increases user interactivity.
   - The acknowledgment also varies with the complexity of user actions.

User Interface Design – Some Principles

5. System status visibility: For functions that are time-consuming, a corresponding graphical cue that signifies the ongoing progress has a major impact on user attention.
6. Remove extra elements: A screen should not have anything that does not add value to user tasks as that increases design noise.

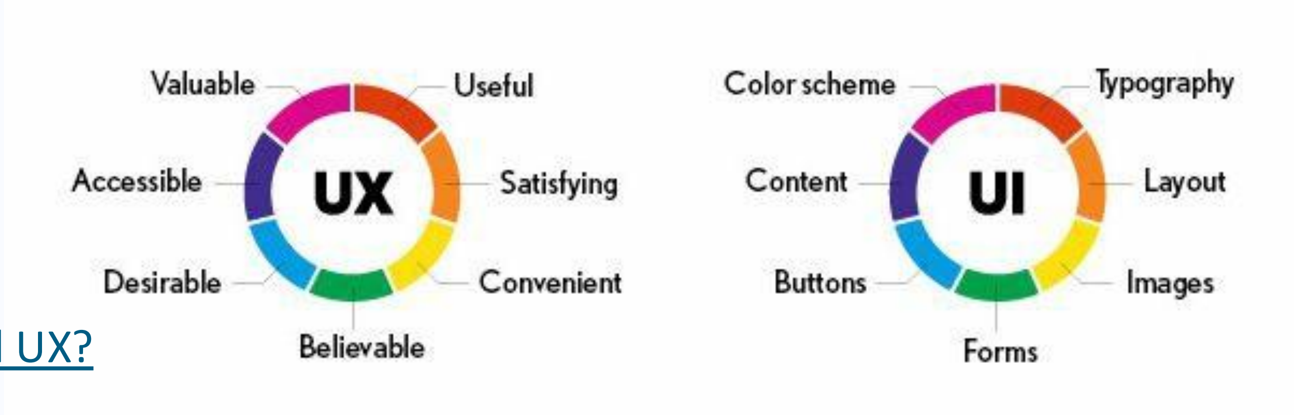
7. Ease of language: The language has to be sharp and crisp with minimal use of jargon for even the best UI designs to succeed.

User Interface Design – Some Principles

8. Use relatable metaphors: Metaphors strengthen the connection between the real world and the virtual interface.
   - For instance, words like 'trash' or 'bin' inevitably point toward data clean-up.
9. Reduce the number of actions to furnish specific tasks: The three-click rule says that a design should accomplish user goals within three clicks to avoid user fatigue.
10. Visual and functional consistency: A set of design managing guidelines will help you maintain consistency in your fonts, themes, and colour palettes.
    - Also, visual elements should perform the same function throughout the interface.

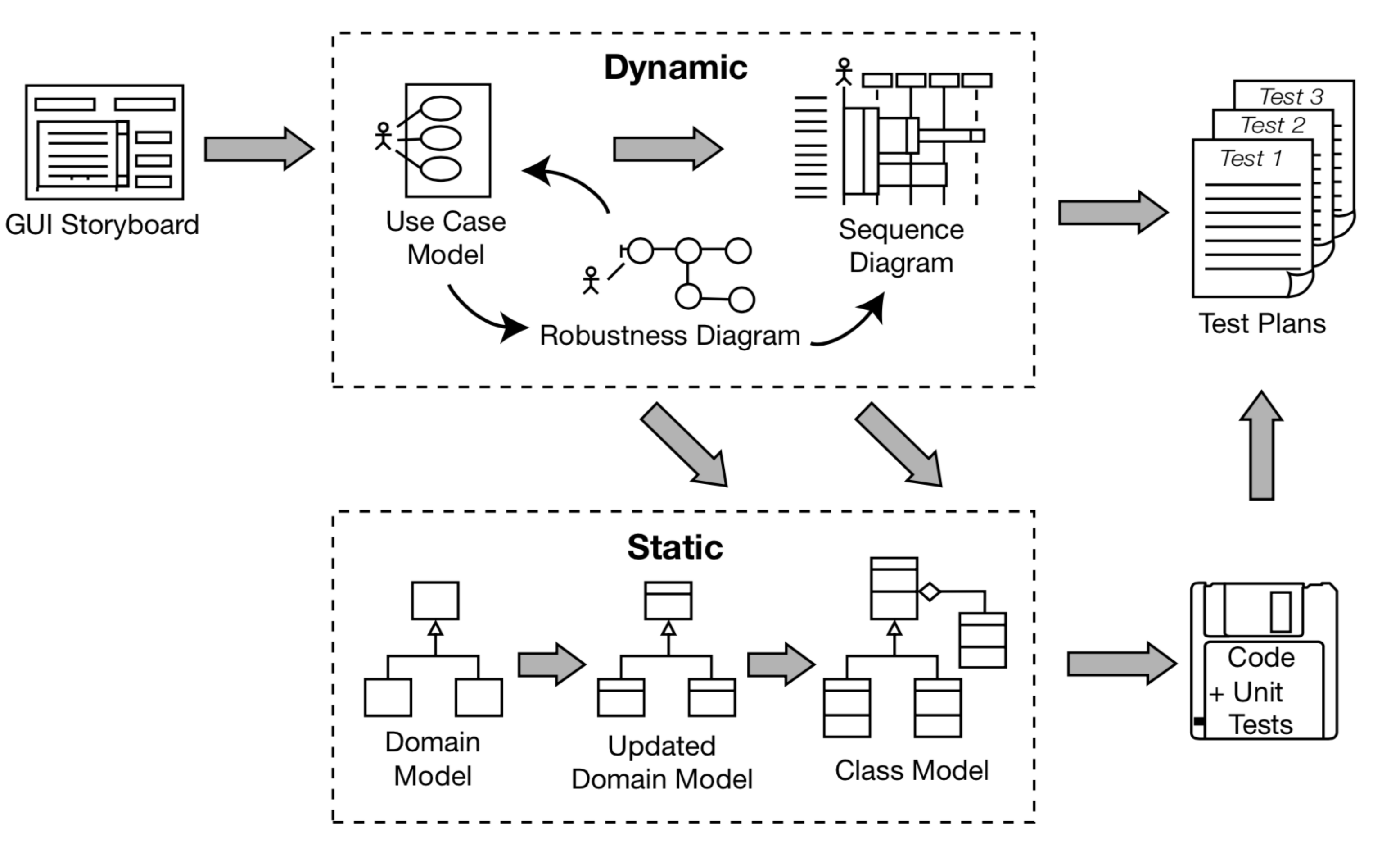
What’s the Difference Between UX and UI?

- User Experience (UX) and UI designs are deeply interconnected but have crucial differences as they are two distinct professions.
- UX designers work on the overall feel of an experience through user research, underline the specific pain points, and design a product keeping all these aspects in mind.
- User interface design is about translating this user experience into an interface with the help of visual elements, typography, images, and colors to connect to a specific audience.



What is the difference between UI and UX?

ICONIX Process – UI Design as a starting point



https://iconixprocess.wordpress.com/iconix-process/

UI Design Languages

- The Interaction Flow Modelling Language (IFML) is a standardised modelling language in the field of software engineering.
- IFML includes a set of graphic notations to create visual models of user interactions and front-end behaviour in software systems. It provides constructs for:
  - Content that is seen by users within the user interface.
  - Navigation paths between elements of the user interface.
  - User Events and Interactions.
  - Binding to the business logic.
  - Binding to persistence layers.

Essential IFML Modelling Concepts

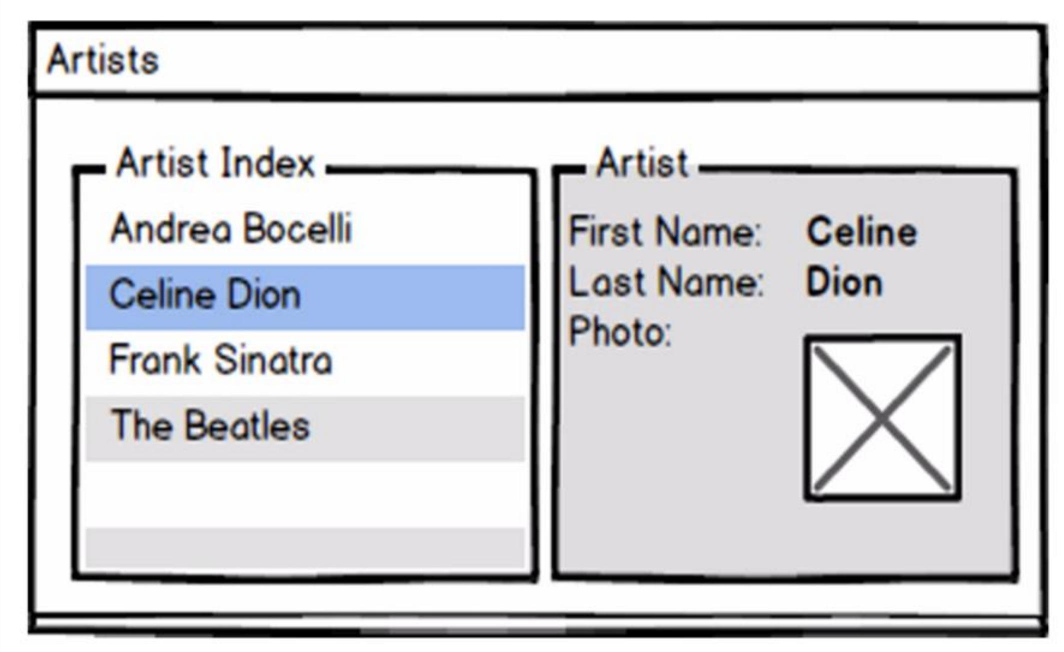
- **ViewContainer** – An element of the interface that comprises other containers and/or elements displaying content and supporting interaction.
  - *Examples:* Web Page, Window, Panel.
- **ViewComponent** – An element of the interface that displays content or accepts input.
  - *Examples:* An HTML list, A JavaScript image gallery, An input form.
- **Event** – Users interaction or system-generated occurrence that affects the state of the application.
  - *Examples:* Selecting an item from a list, Submitting a form, Clicking a menu item.
- **Action** – A piece of business logic triggered by an event.
  - *Examples:* A database update, The sending of an e-mail.

Essential IFML Modelling Concepts

- **Navigation Flow** – An input-output dependency. The source of the link has some output that is associated with the input of the target of the link.
  - *Examples:* Sending and receiving of parameters in the HTTP request.
- **Data Flow.** An input-output dependency between the source and the target of the flow that is not directly associated with the user interaction.
  - *Examples:* Sending the book ID from a component displaying the book details to another component showing the book reviews.
- **Parameter Binding Group.** Set of Parameter Bindings* associated with an Interaction Flow (being it navigation or data flow).
  - *Examples:* Connecting the book title in output from a component to the input parameter of another component.

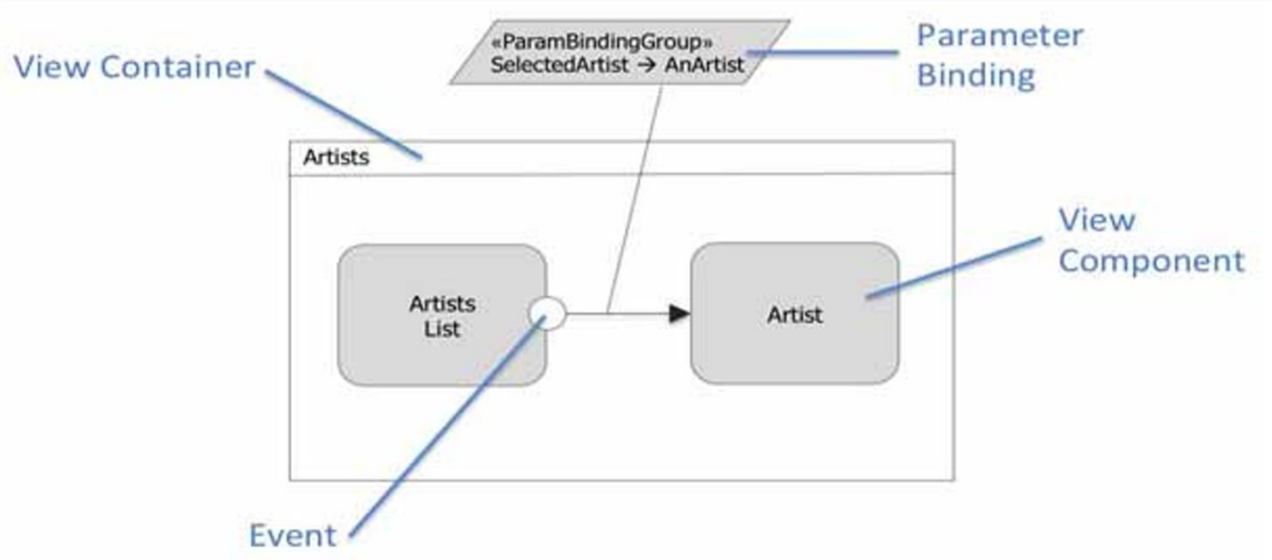
**Specification that an output parameter of a source is associated with an input parameter of a target.*

Example of UI (left) and corresponding IFML model (right)



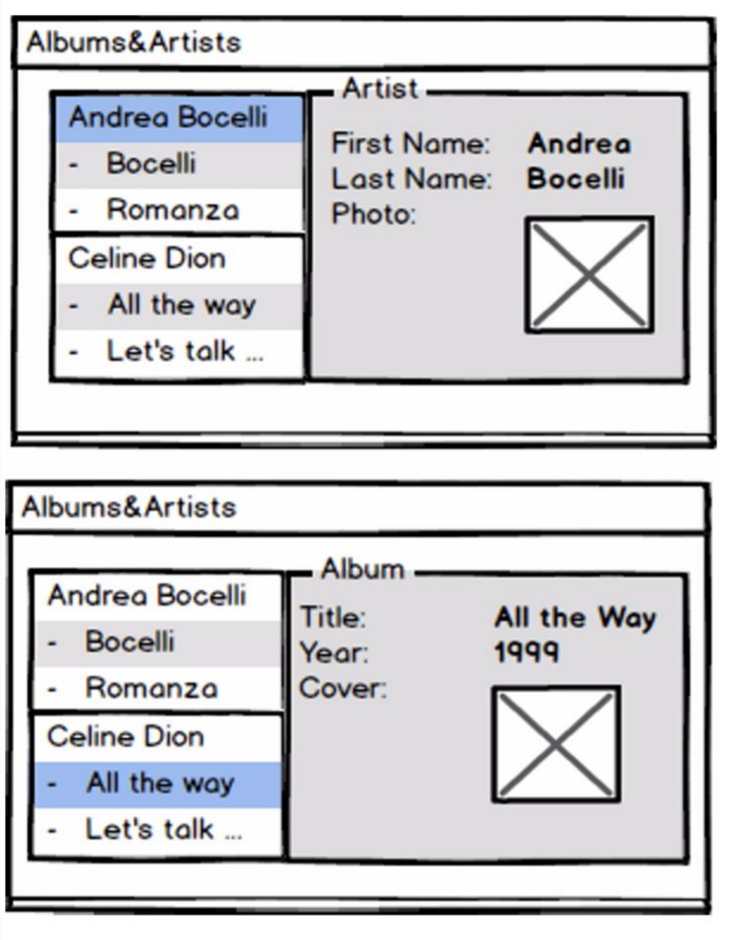
Andrea Bocelli
Celine Dion
Frank Sinatra
The Beatles

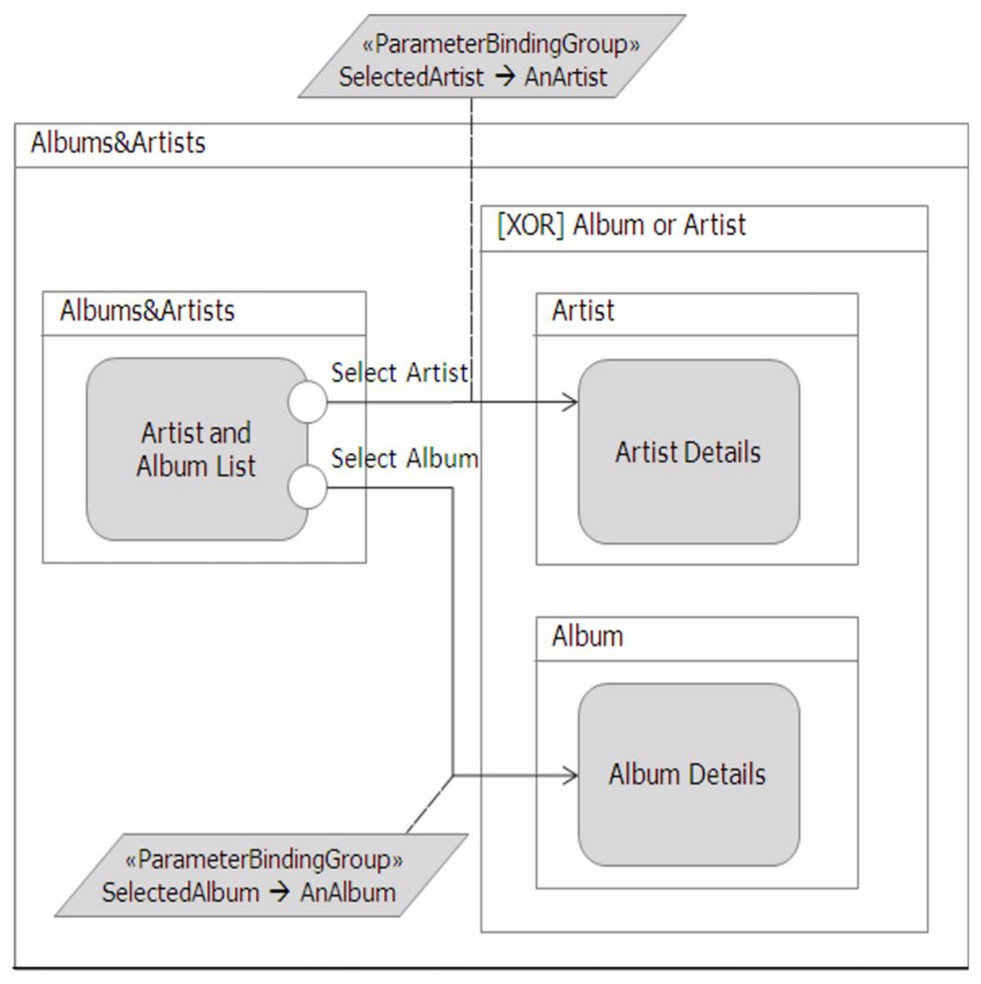
First Name: Celine
Last Name: Dion
Photo:



The user selects an item in the list and displays its details in the same view container.

IFML model –Another example





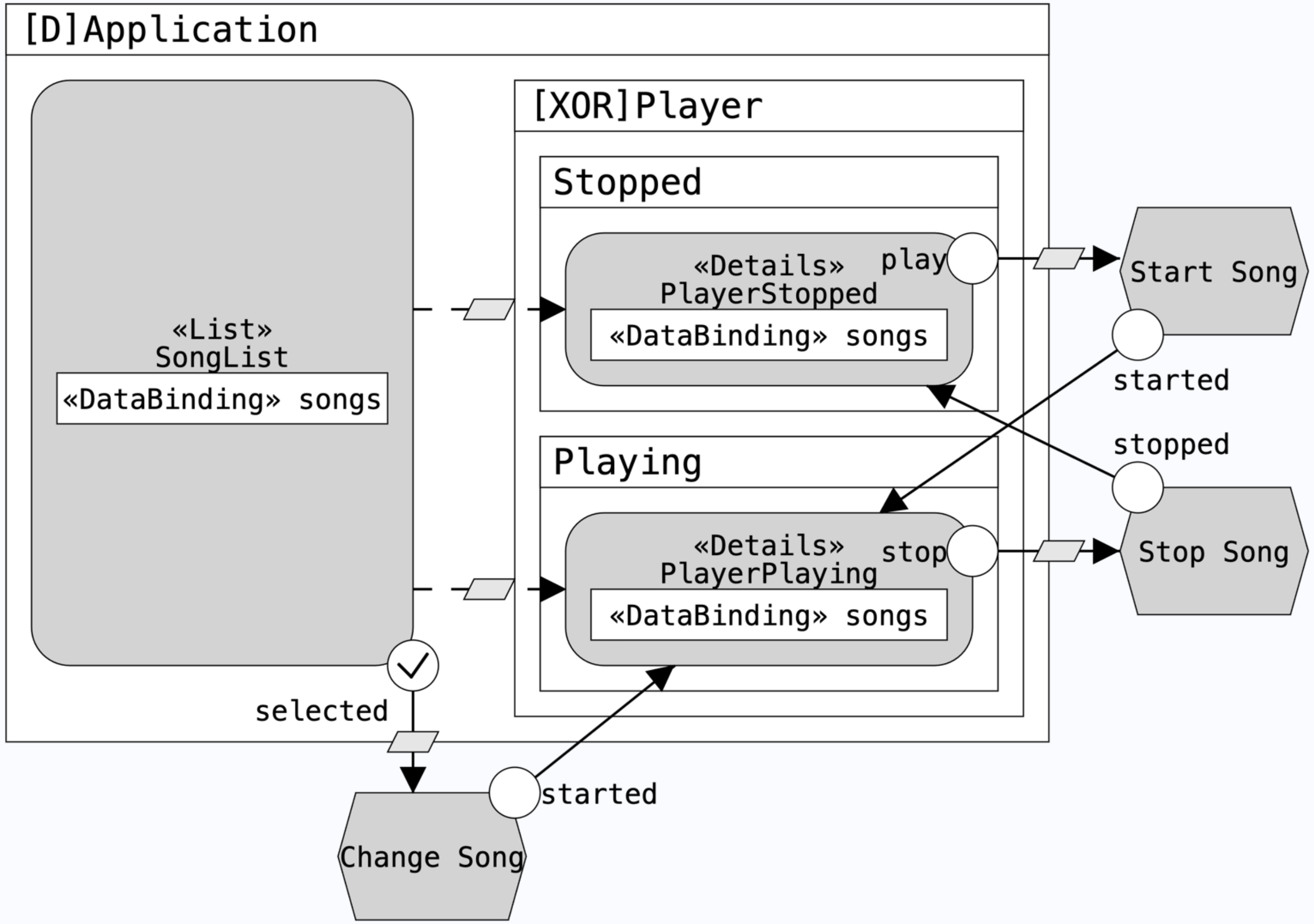
Example of user interface (left) and corresponding IFML model (right).

One top-level container comprises three view containers: one with a list of artists and of their albums, one with the details of an artist, and one with the details of an album.

The latter two view containers are mutually exclusive: only one at a time is displayed.

IFML – Tool Support

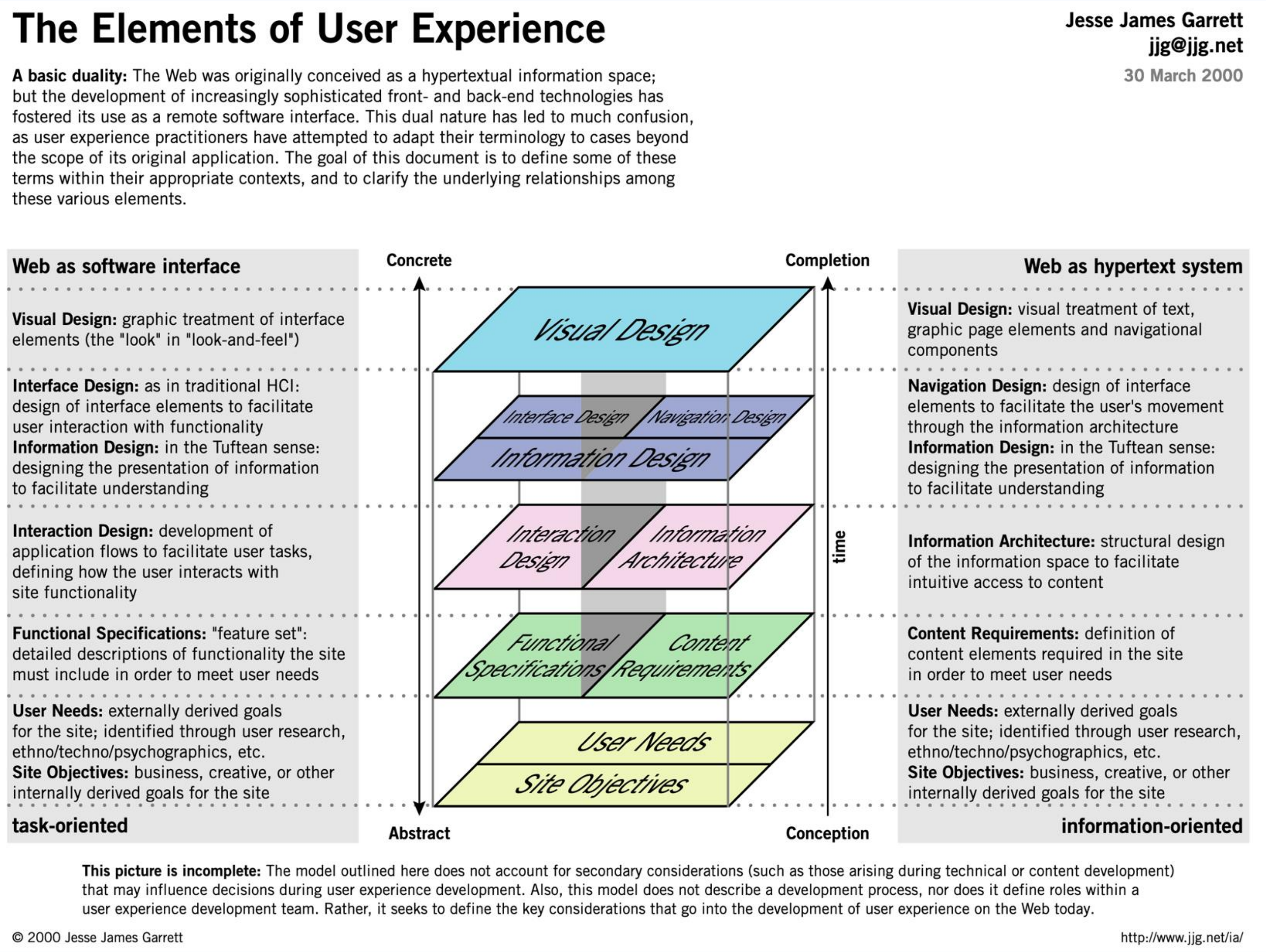
https://editor.ifmledit.org/



Other Tools for UI Design – Figma

- Figma is a free, online UI tool to create, collaborate, prototype, and handoff. (https://www.figma.com/ui-design-tool/)
- What is UI design?
- 5 UI design principles to improve your product
- What is the difference between UI and UX?
- Simplicity in design

UI/UX Design – The Elements of User Experience



The Elements of User Experience

Jesse James Garrett
jjg@jjg.net
30 March 2000

A basic duality: The Web was originally conceived as a hypertextual information space; but the development of increasingly sophisticated front- and back-end technologies has fostered its use as a remote software interface. This dual nature has led to much confusion, as user experience practitioners have attempted to adapt their terminology to cases beyond the scope of its original application. The goal of this document is to define some of these terms within their appropriate contexts, and to clarify the underlying relationships among these various elements.

Web as software interface

Visual Design: graphic treatment of interface elements (the "look" in "look-and-feel")

Interface Design: as in traditional HCI: design of interface elements to facilitate user interaction with functionality
Information Design: in the Tuftean sense: designing the presentation of information to facilitate understanding

Interaction Design: development of application flows to facilitate user tasks, defining how the user interacts with site functionality

Functional Specifications: "feature set": detailed descriptions of functionality the site must include in order to meet user needs

User Needs: externally derived goals for the site; identified through user research, ethno/techno/psychographics, etc.
Site Objectives: business, creative, or other internally derived goals for the site

task-oriented

Web as hypertext system

Visual Design: visual treatment of text, graphic page elements and navigational components

Navigation Design: design of interface elements to facilitate the user's movement through the information architecture
Information Design: in the Tuftean sense: designing the presentation of information to facilitate understanding

Information Architecture: structural design of the information space to facilitate intuitive access to content

Content Requirements: definition of content elements required in the site in order to meet user needs

User Needs: externally derived goals for the site; identified through user research, ethno/techno/psychographics, etc.
Site Objectives: business, creative, or other internally derived goals for the site

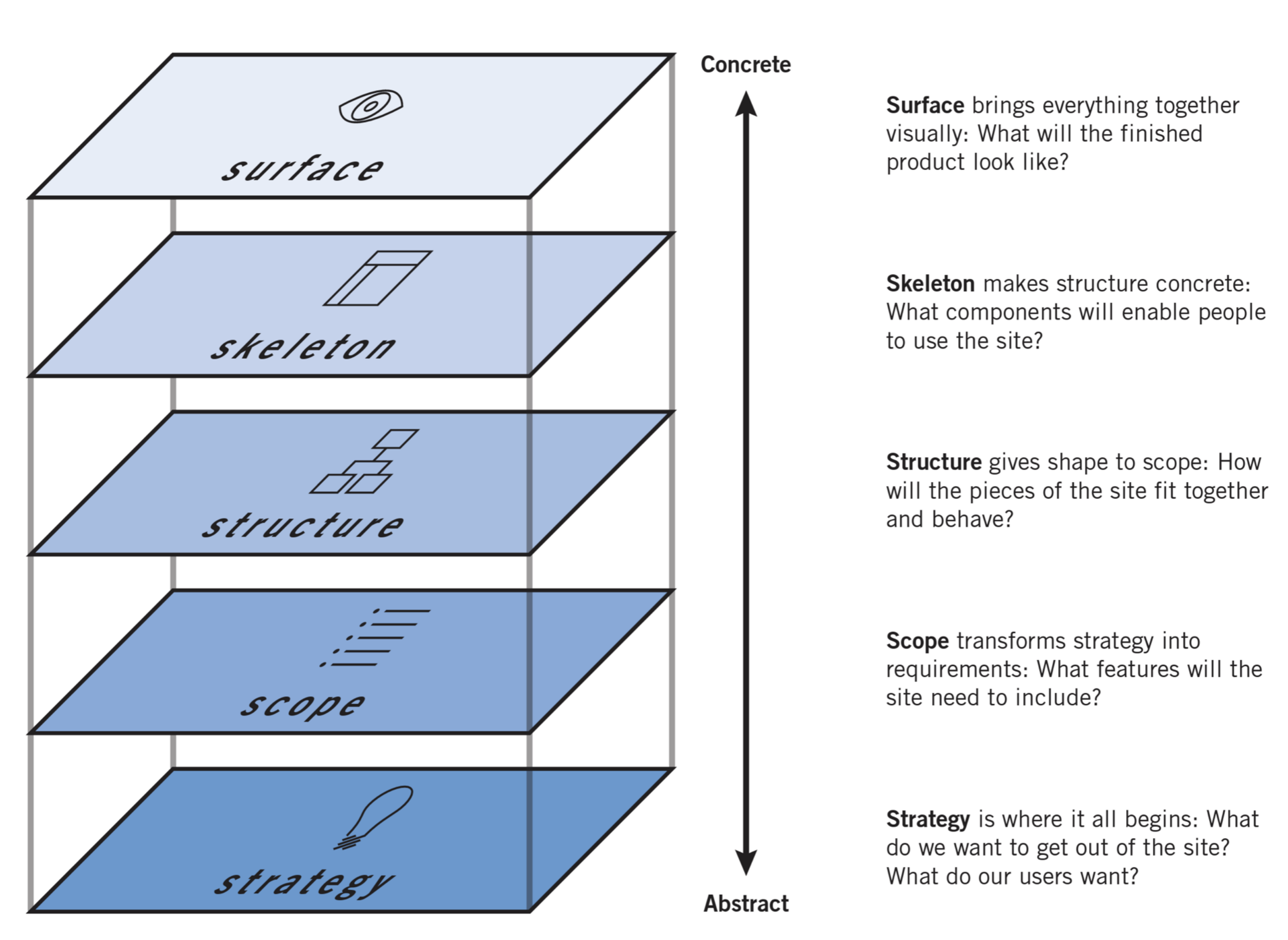
information-oriented

This picture is incomplete: The model outlined here does not account for secondary considerations (such as those arising during technical or content development) that may influence decisions during user experience development. Also, this model does not describe a development process, nor does it define roles within a user experience development team. Rather, it seeks to define the key considerations that go into the development of user experience on the Web today.

http://www.jjg.net/ia/

http://www.jjg.net/elements/pdf/elements.pdf

UI/UX Design – The Elements of User Experience



Surface brings everything together visually: What will the finished product look like?

Skeleton makes structure concrete: What components will enable people to use the site?

Structure gives shape to scope: How will the pieces of the site fit together and behave?

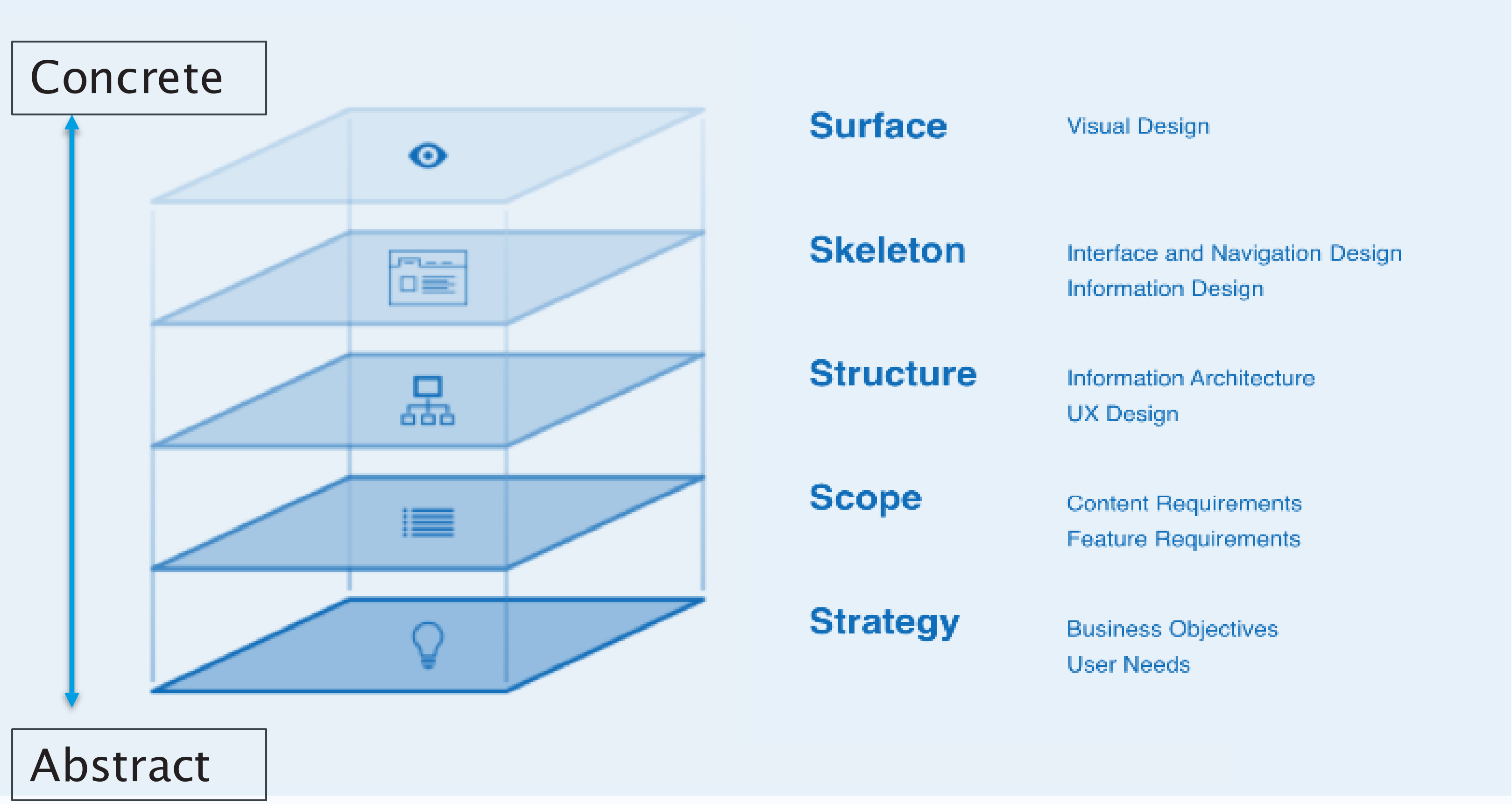
Scope transforms strategy into requirements: What features will the site need to include?

Strategy is where it all begins: What do we want to get out of the site? What do our users want?

http://www.jjg.net/elements/pdf/elements_simpleplanes.pdf

UI- Different Layers

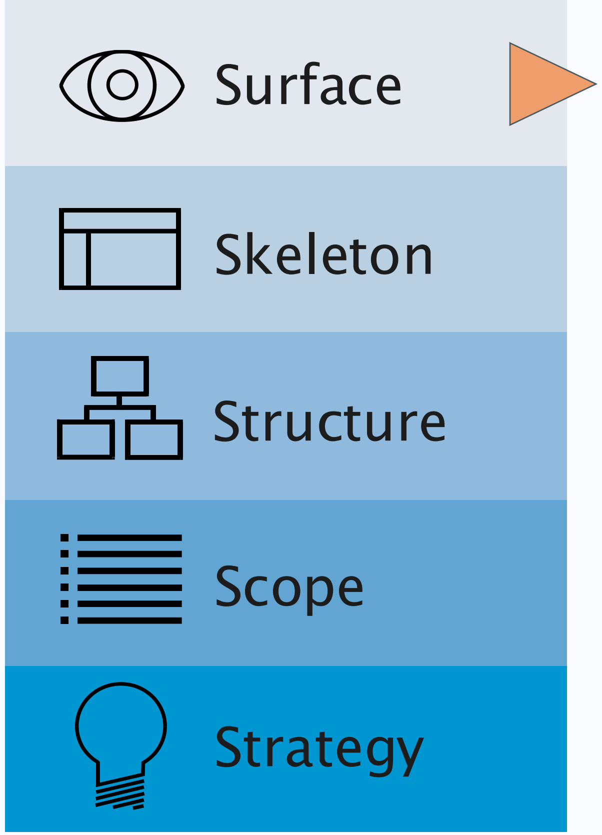
- User Interface is built from dependent layers

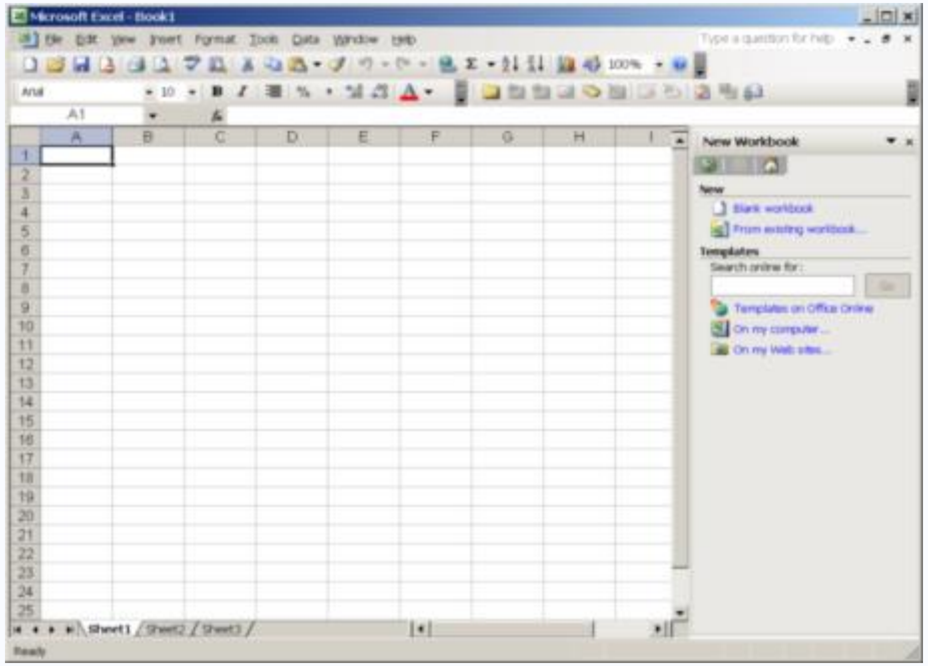


Jesse James Garrett’s **Elements of User Experience**

The Surface Layer

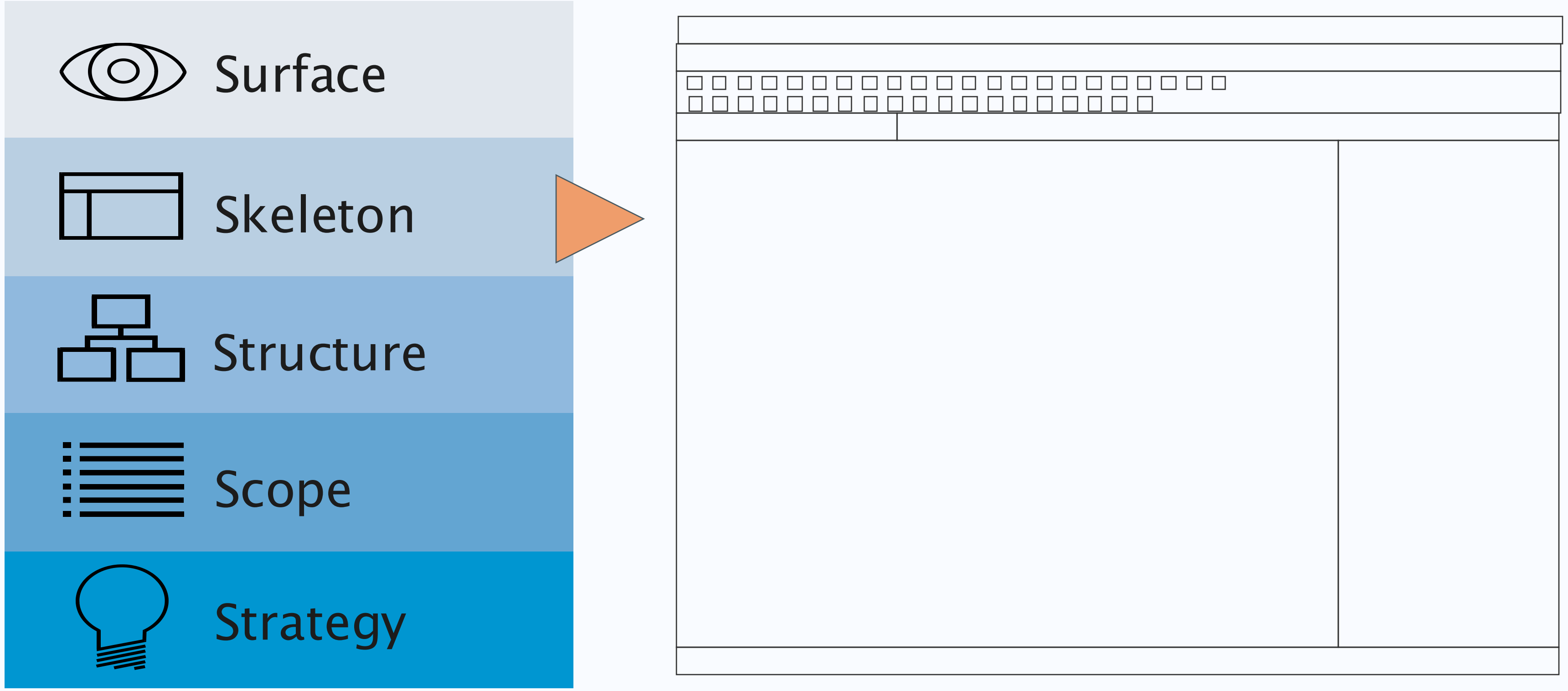
- The surface layer describes finished visual design aspects



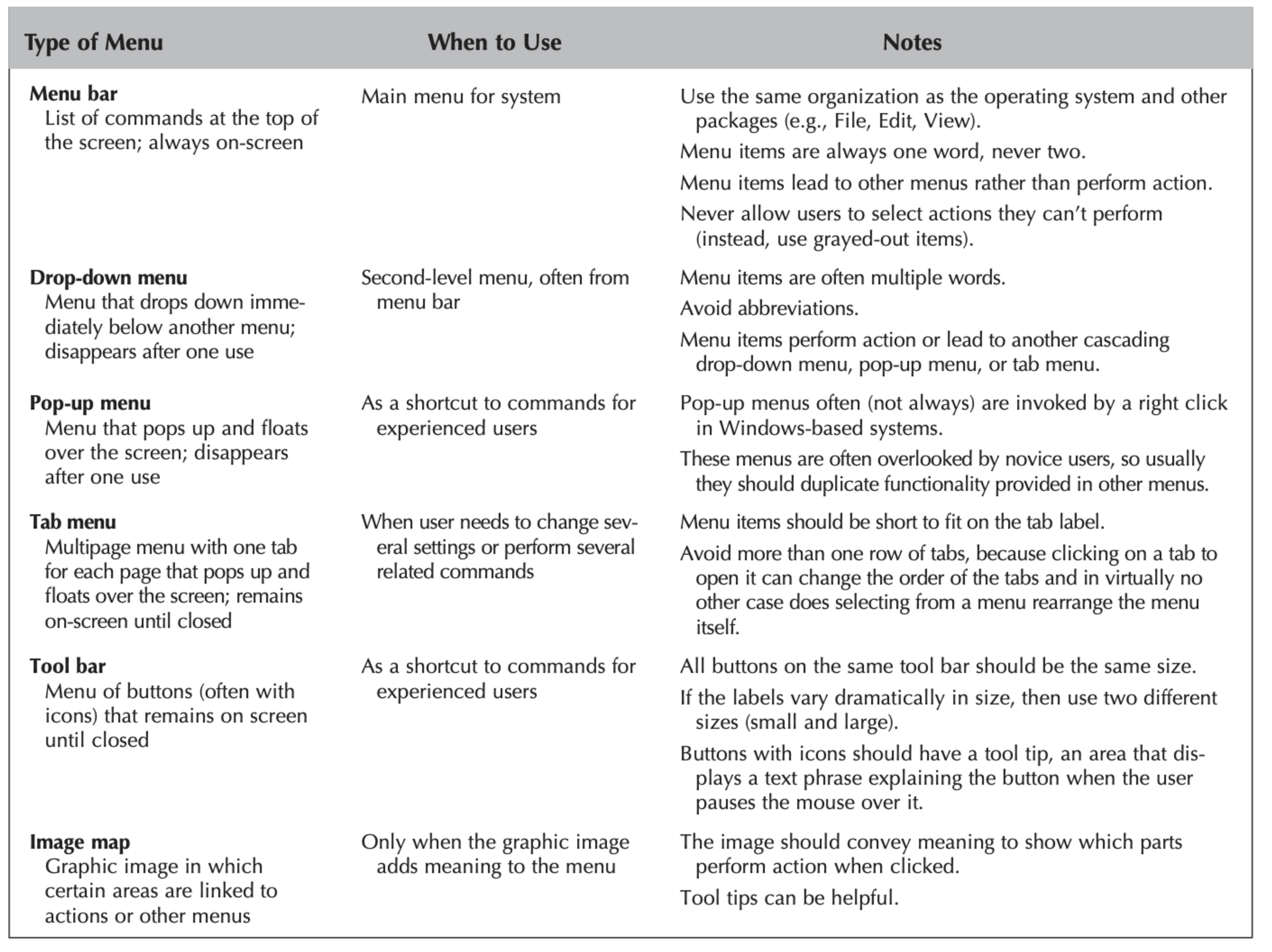


The Skeleton Layer

- The skeleton describes screen layout and functional compartments on the screen



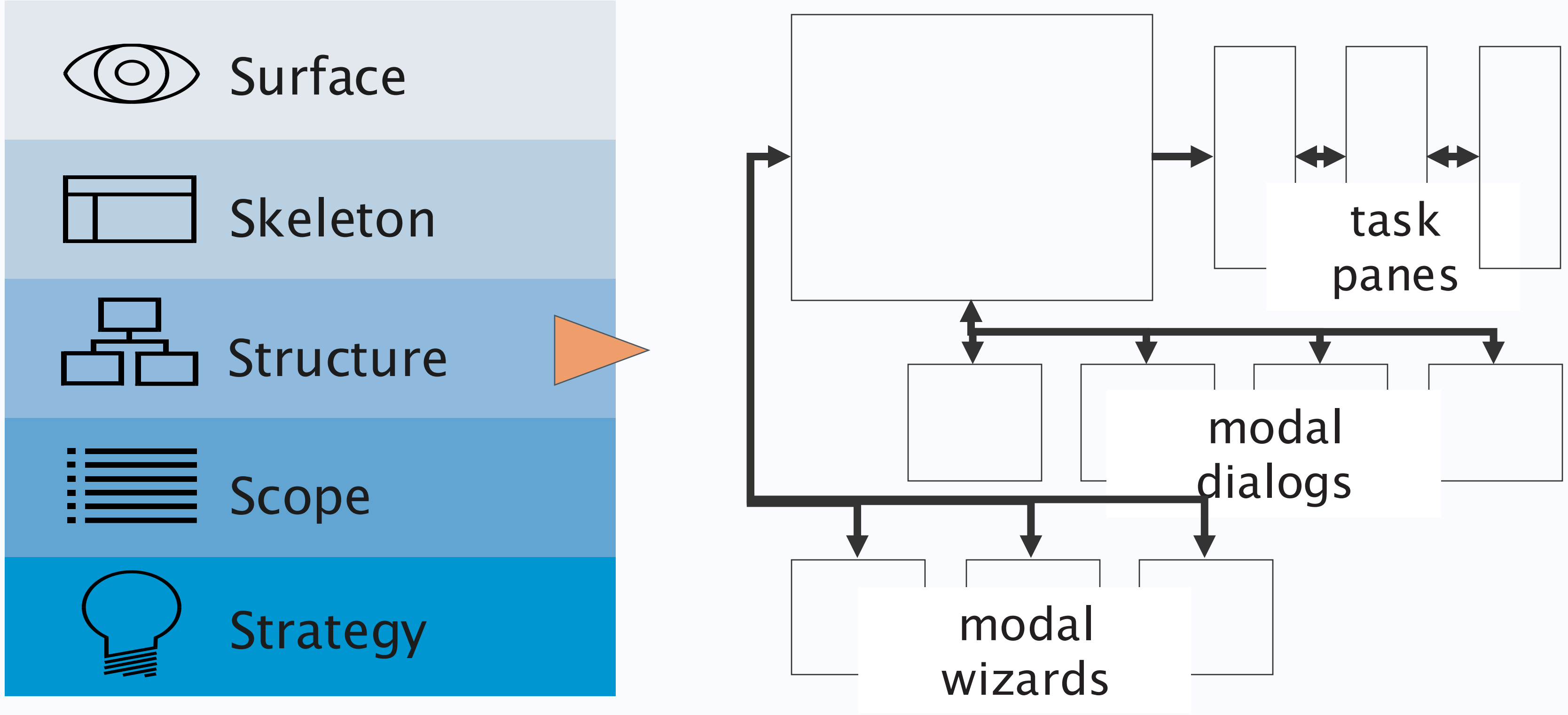
Navigation Controls



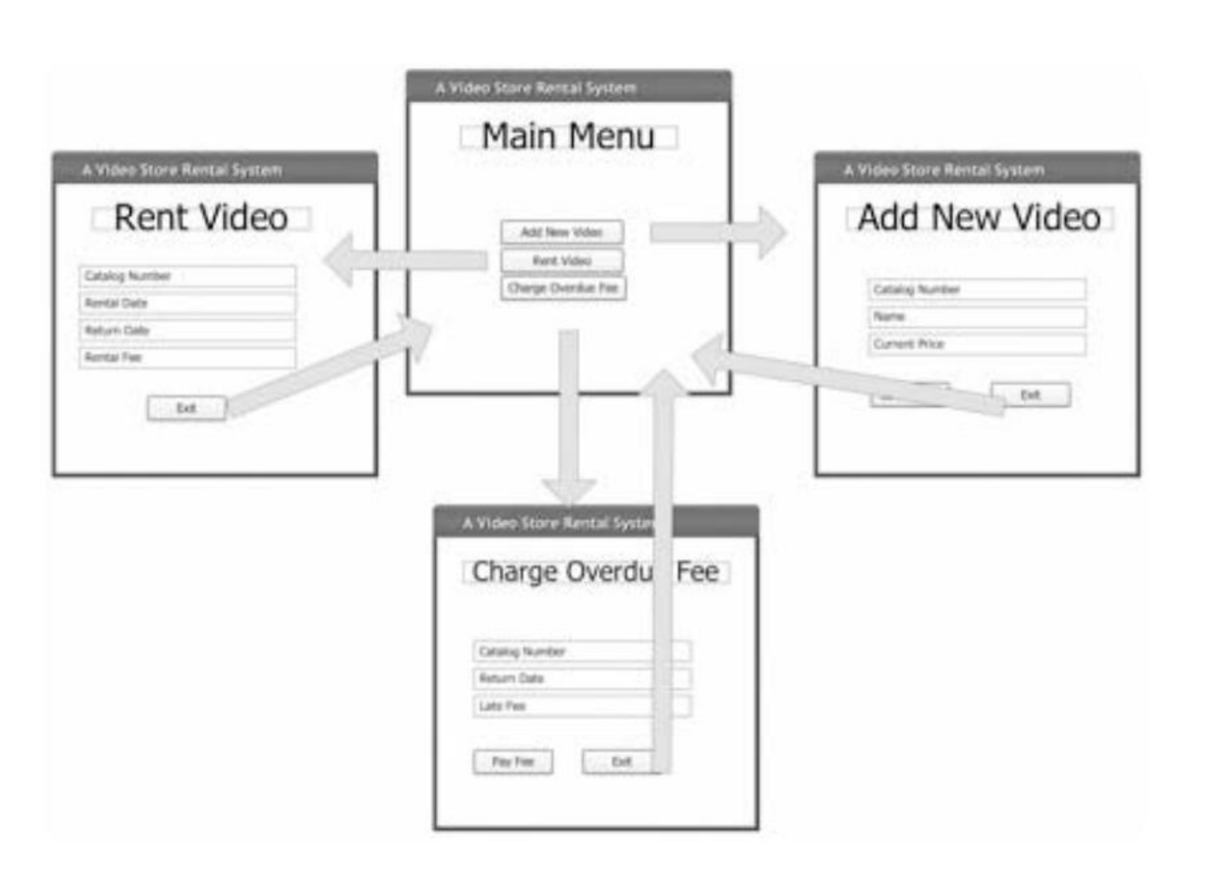
Type of Menu	When to Use	Notes
Menu bar List of commands at the top of the screen; always on-screen	Main menu for system	Use the same organization as the operating system and other packages (e.g., File, Edit, View). Menu items are always one word, never two. Menu items lead to other menus rather than perform action. Never allow users to select actions they can't perform (instead, use grayed-out items).
Drop-down menu Menu that drops down immediately below another menu; disappears after one use	Second-level menu, often from menu bar	Menu items are often multiple words. Avoid abbreviations. Menu items perform action or lead to another cascading drop-down menu, pop-up menu, or tab menu.
Pop-up menu Menu that pops up and floats over the screen; disappears after one use	As a shortcut to commands for experienced users	Pop-up menus often (not always) are invoked by a right click in Windows-based systems. These menus are often overlooked by novice users, so usually they should duplicate functionality provided in other menus.
Tab menu Multipage menu with one tab for each page that pops up and floats over the screen; remains on-screen until closed	When user needs to change several settings or perform several related commands	Menu items should be short to fit on the tab label. Avoid more than one row of tabs, because clicking on a tab to open it can change the order of the tabs and in virtually no other case does selecting from a menu rearrange the menu itself.
Tool bar Menu of buttons (often with icons) that remains on screen until closed	As a shortcut to commands for experienced users	All buttons on the same tool bar should be the same size. If the labels vary dramatically in size, then use two different sizes (small and large). Buttons with icons should have a tool tip, an area that displays a text phrase explaining the button when the user pauses the mouse over it.
Image map Graphic image in which certain areas are linked to actions or other menus	Only when the graphic image adds meaning to the menu	The image should convey meaning to show which parts perform action when clicked. Tool tips can be helpful.

Structure Layer

- Structure defines navigation from one section to another section in the user interface

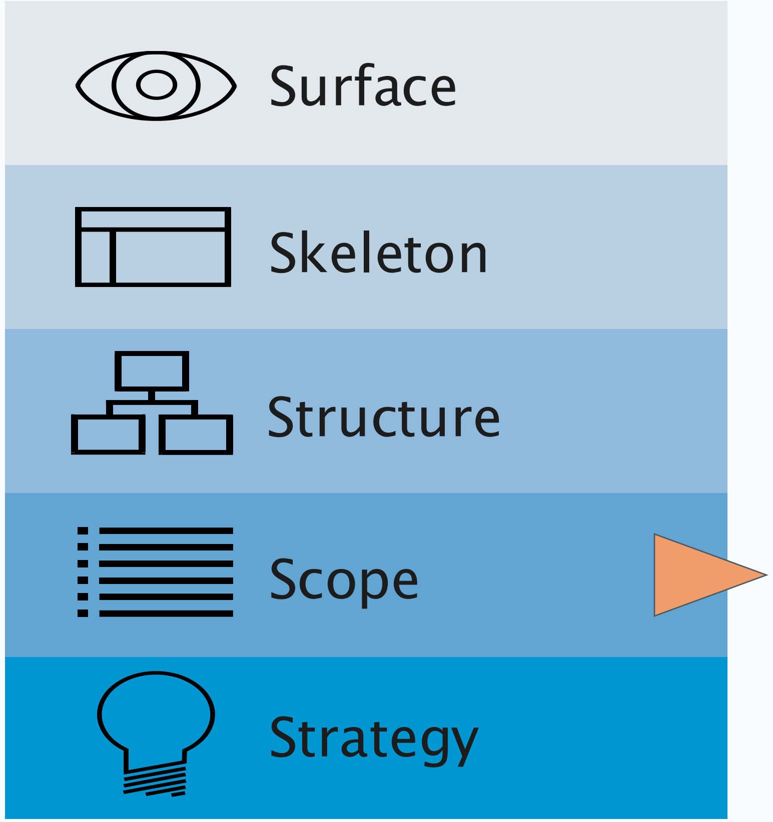


Windows Navigation and Layout Diagrams



The Scope Layer

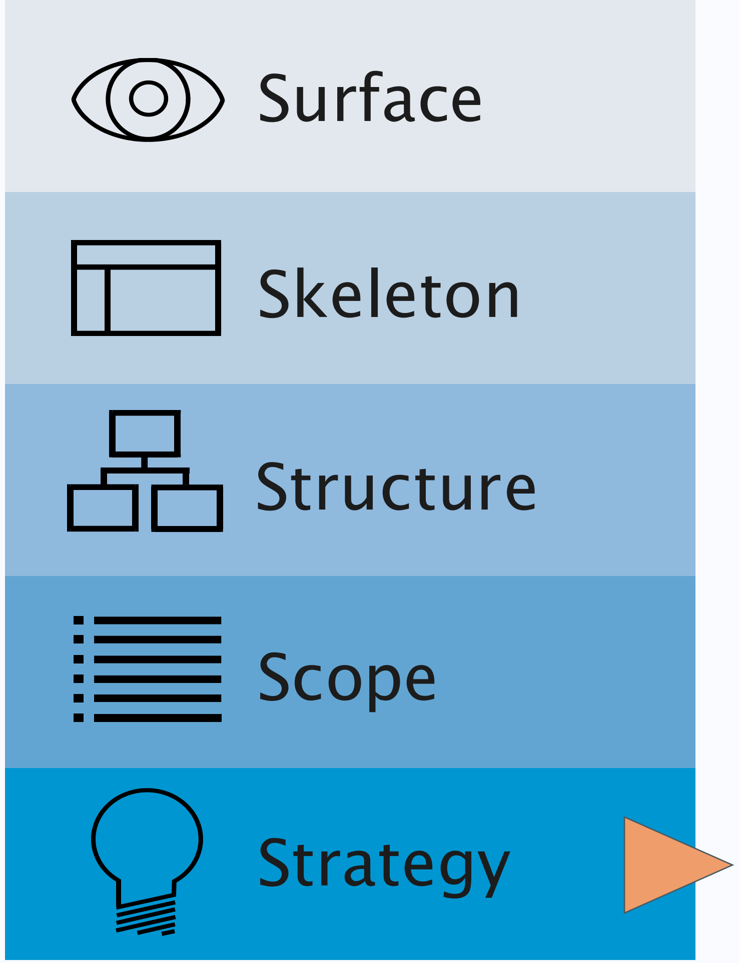
- The sections in the user interface are built to support user tasks.



user tasks:

- enter numbers
- enter text
- enter formulas
- format cells
- sort information
- filter information
- aggregate information
- graph data
- save data
- import data
- export data
- print
- …..

The Strategy layer



business goals:

- displace competitive products
- motivate sale of other integrated products
- establish file format as default information sharing format
- …

user Roles:

- accountant
- business planner
- housewife
- …

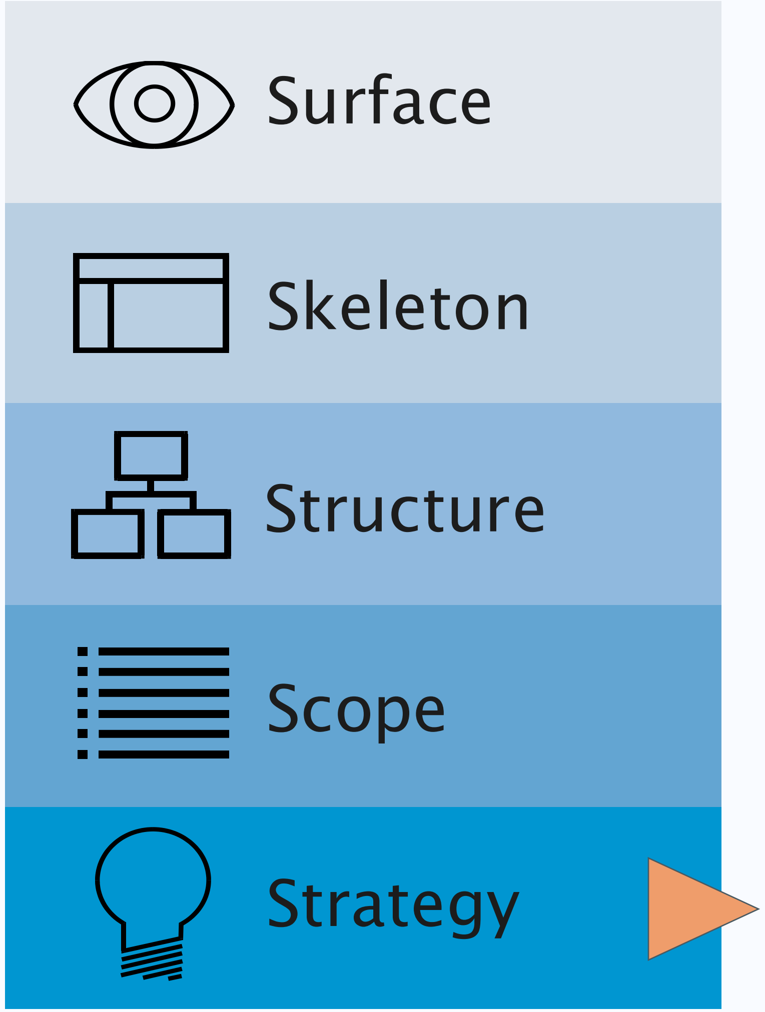
usage contexts:

- office desktop
- laptop on airplane
- Interactive display in car
- …

The Strategy layer

- Business goals determine user constituencies and relevant contexts supported by the system to form a strategy.
- High level goals can be refined to user & stakeholders' specific goals and the contexts that helps to achieve these goals.
- This strategy incorporates not only what the people running the site want to get out of it but also what the users want to get out of the site as well.
  - For example, in the case of a bookstore example, some of the strategic objectives are pretty obvious:
    - Users want to buy books, and we want to sell them.
  - Other objectives might not be so easy to articulate.

An Example – Home Analogy



goals:

- live comfortably
- eat well
- stay clean
- be healthy
- keep up with Jones’s
- …

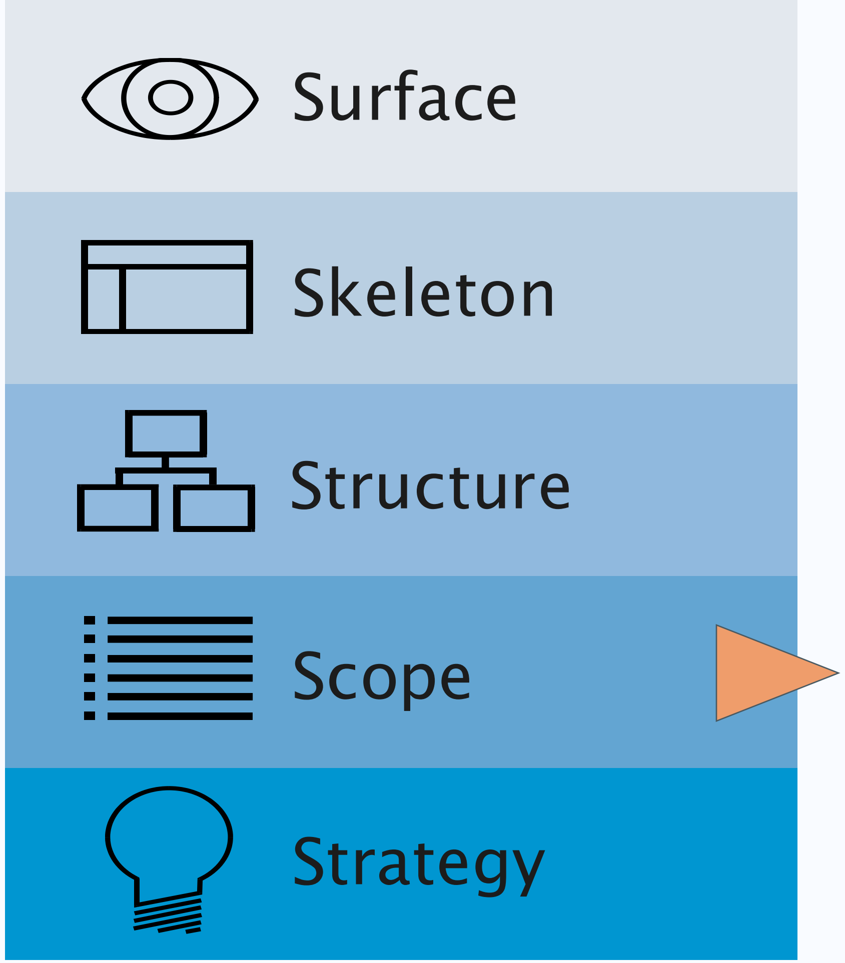
user constituencies:

- me
- spouse
- child
- …

usage contexts:

- suburban neighborhood
- near good schools
- near shopping
- …

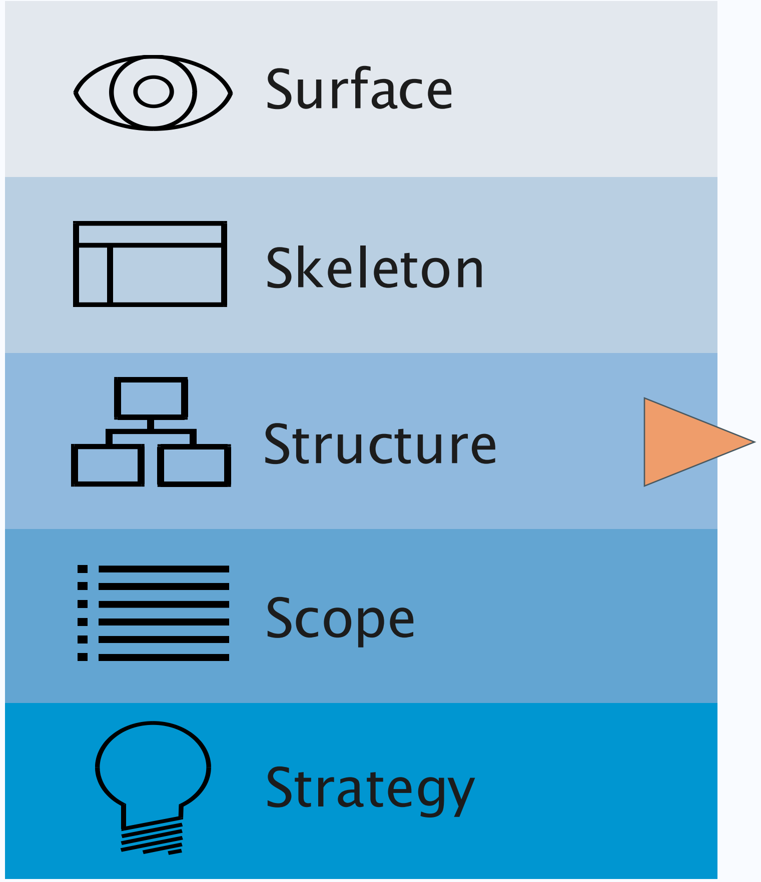
What might I do to reach my goals?



user tasks:

- store food
- prepare food
- eat food
- sleep
- bathe
- store changes of clothing
- stay out of rain
- entertain guests
- entertain self
- …

Arranging tasks by affinity allows me to think about contexts that best support tasks. Contexts in a home have common names we all know.



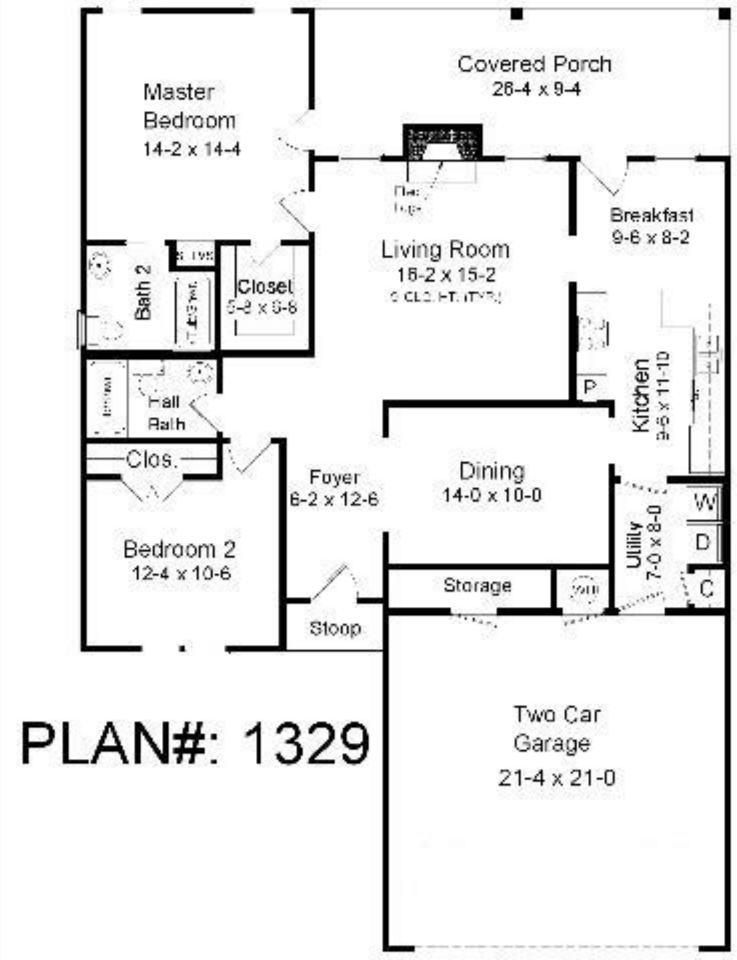
Surface

Skeleton

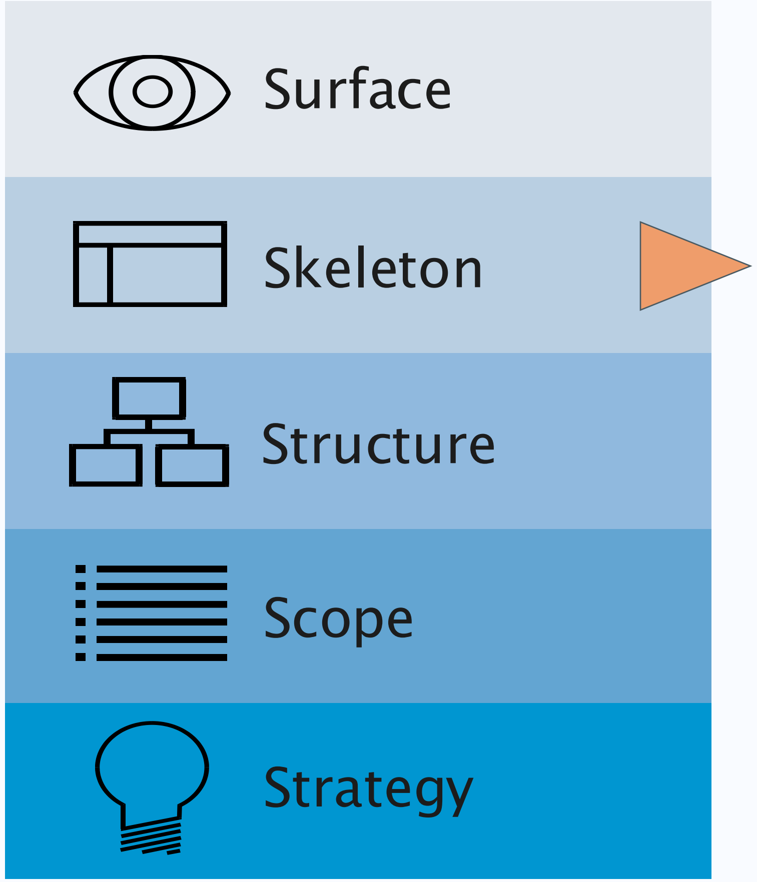
Structure

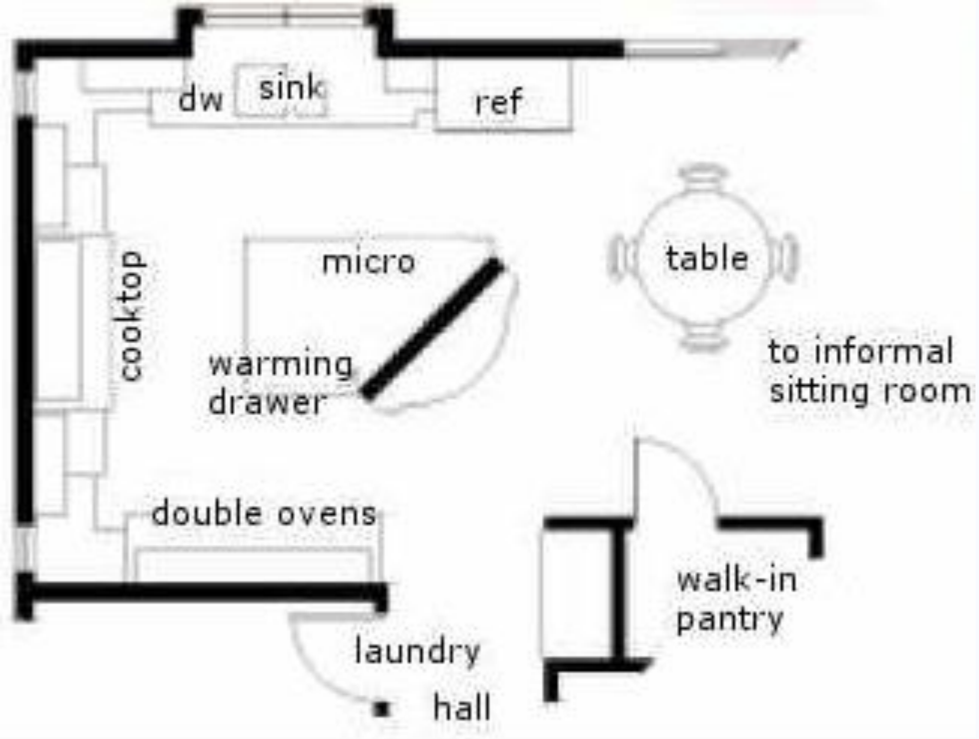
Scope

Strategy

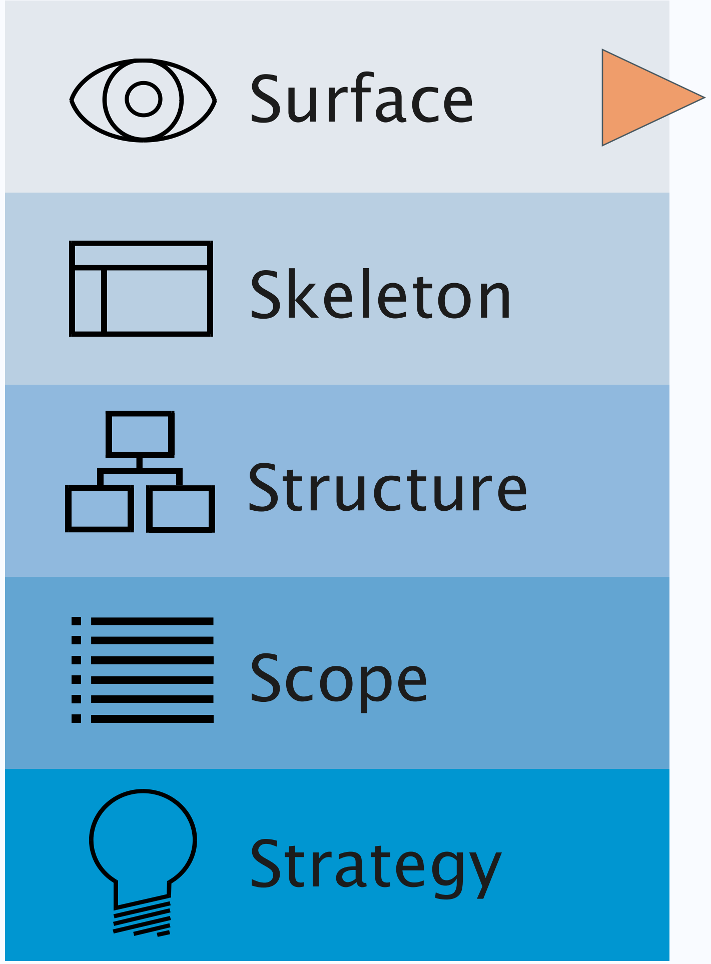


When designing a particular interaction context – a kitchen for instance – I optimise layout and tool choices to support tasks I'll do there.





I’m going to spend a lot of time here; I want my experience to be as pleasant as possible…



Surface

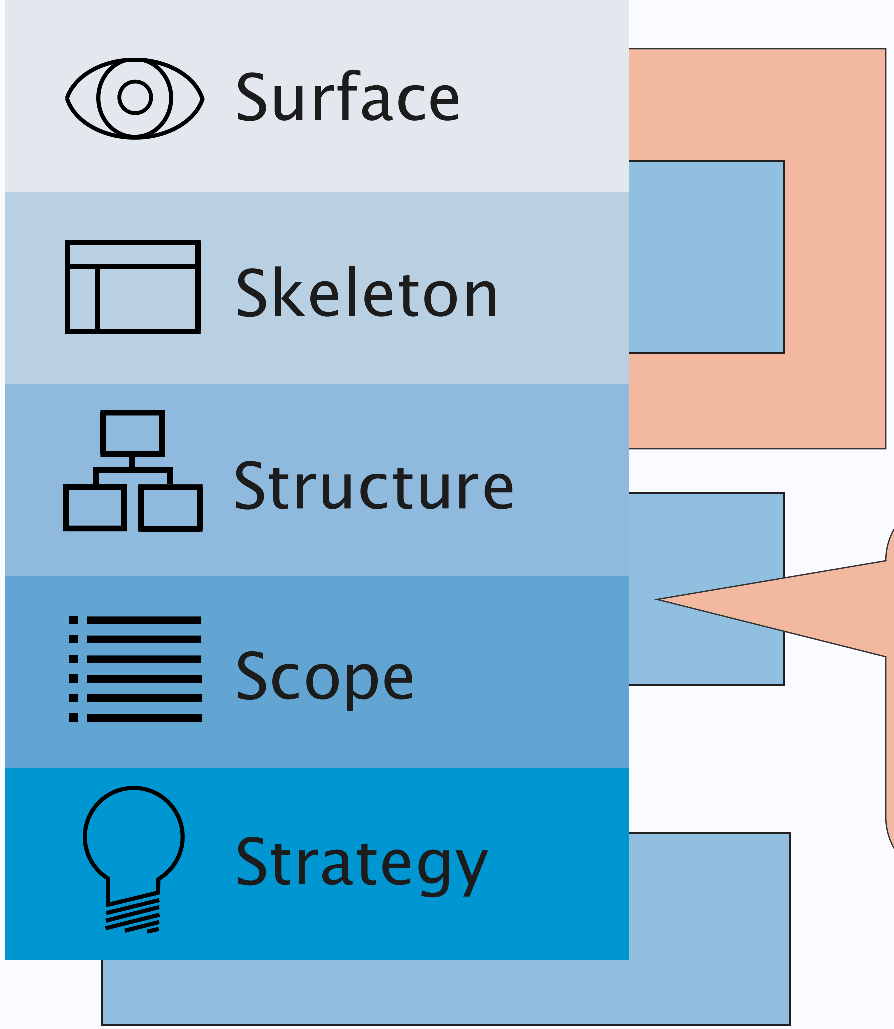
Skeleton

Structure

Scope

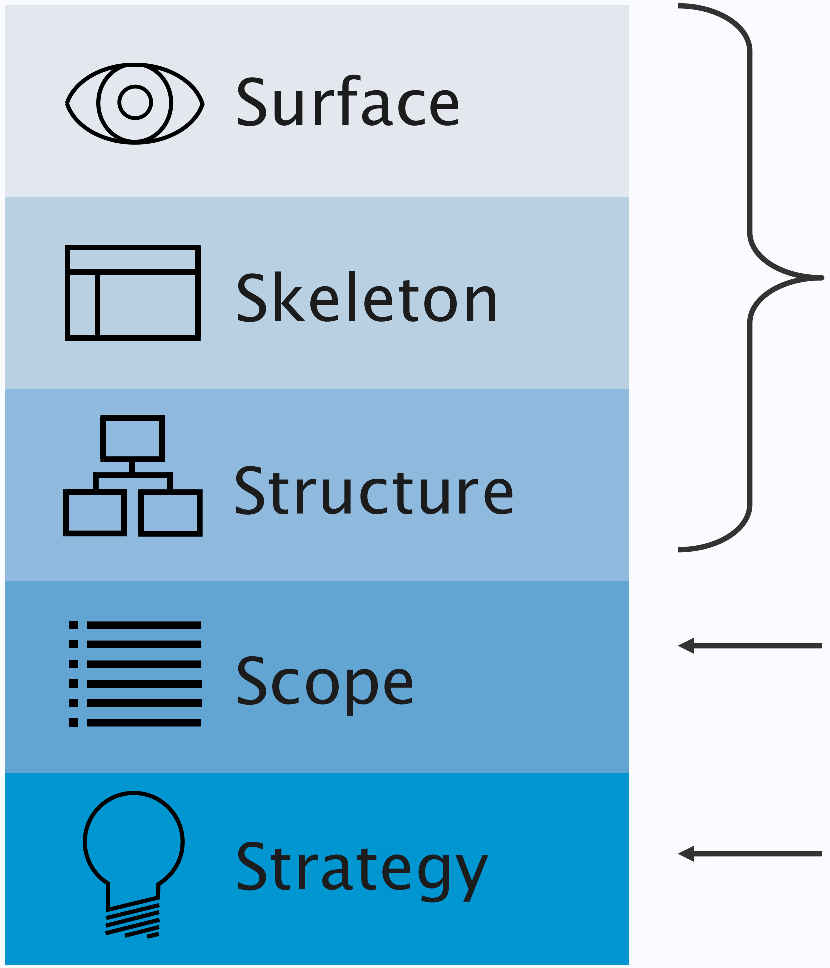
Strategy

Understanding the relationship between goals, tasks, & tools is critical.



one or more users engaged in many **tasks** in support of a common high-level **goal** is often referred to as **work practice**

Garrett’s model provides helpful guidance for tool builders



Tools

- Navigation Map
- Page Wireframes
- UI Design Guidelines

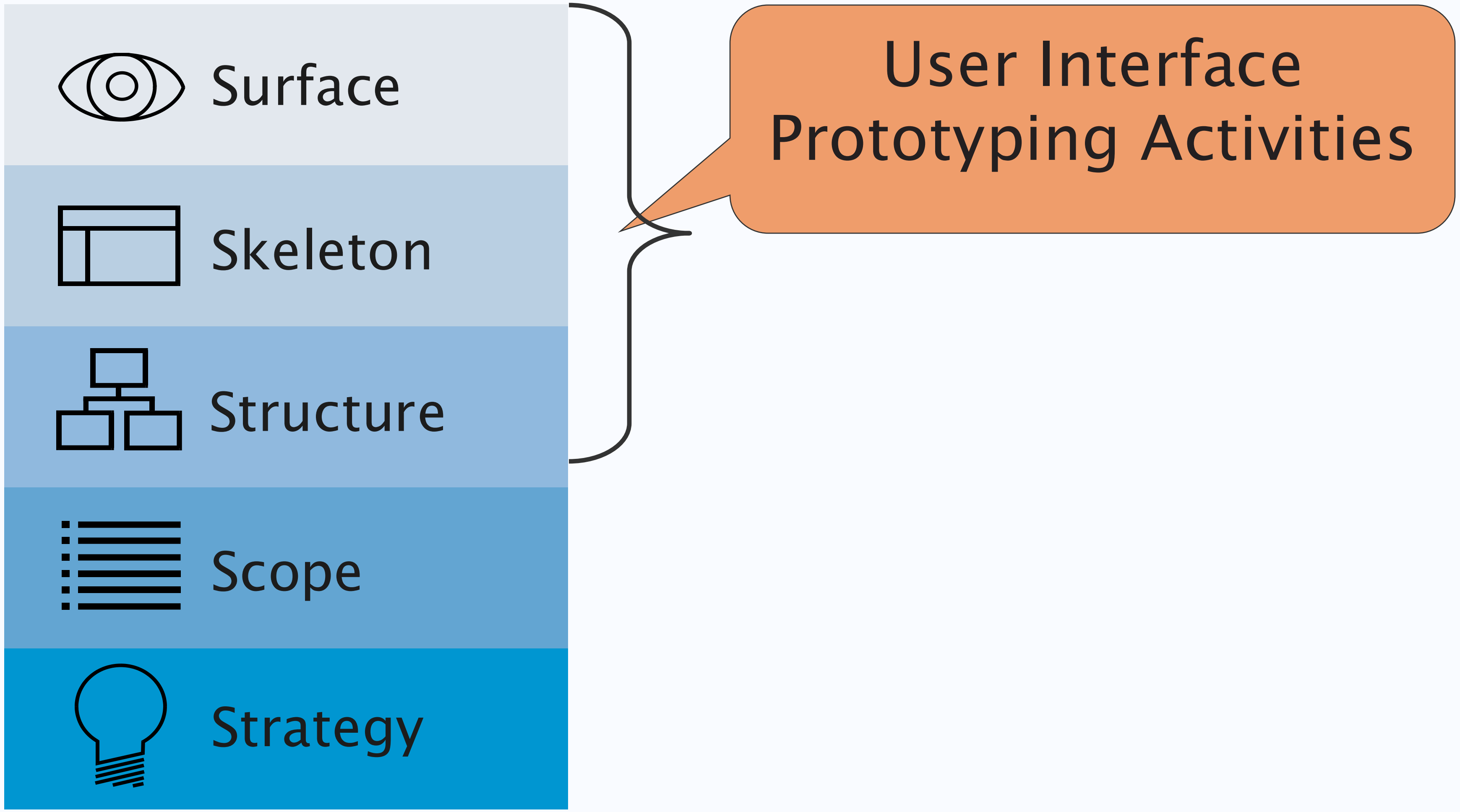
Tasks

- User Tasks & Activities, or Use Cases
- Technology Independent

Goals

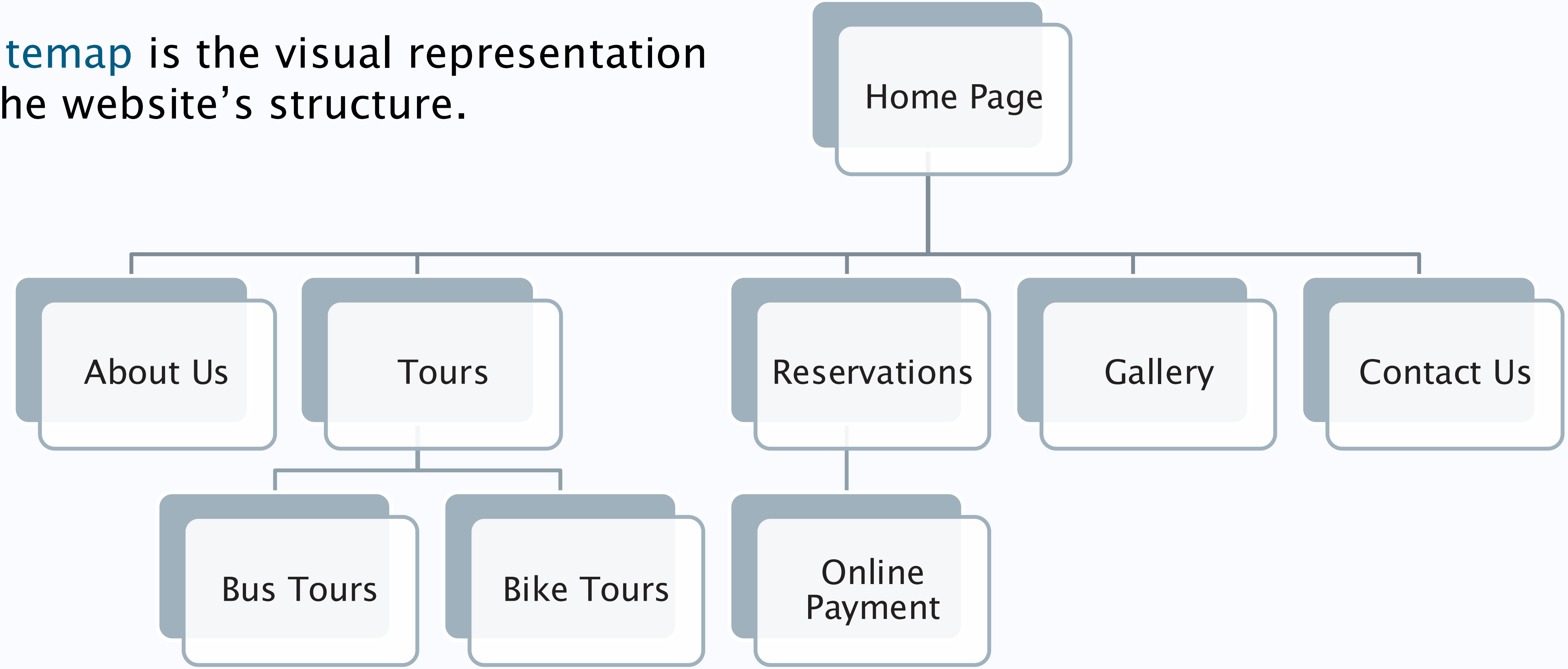
- Business Goals
- User Model with User Goals
- Architectural Goals Based On Context of Use

Today we’ll place our focus on tool-building: the structure, skeleton, & surface



Sitemap – An Example

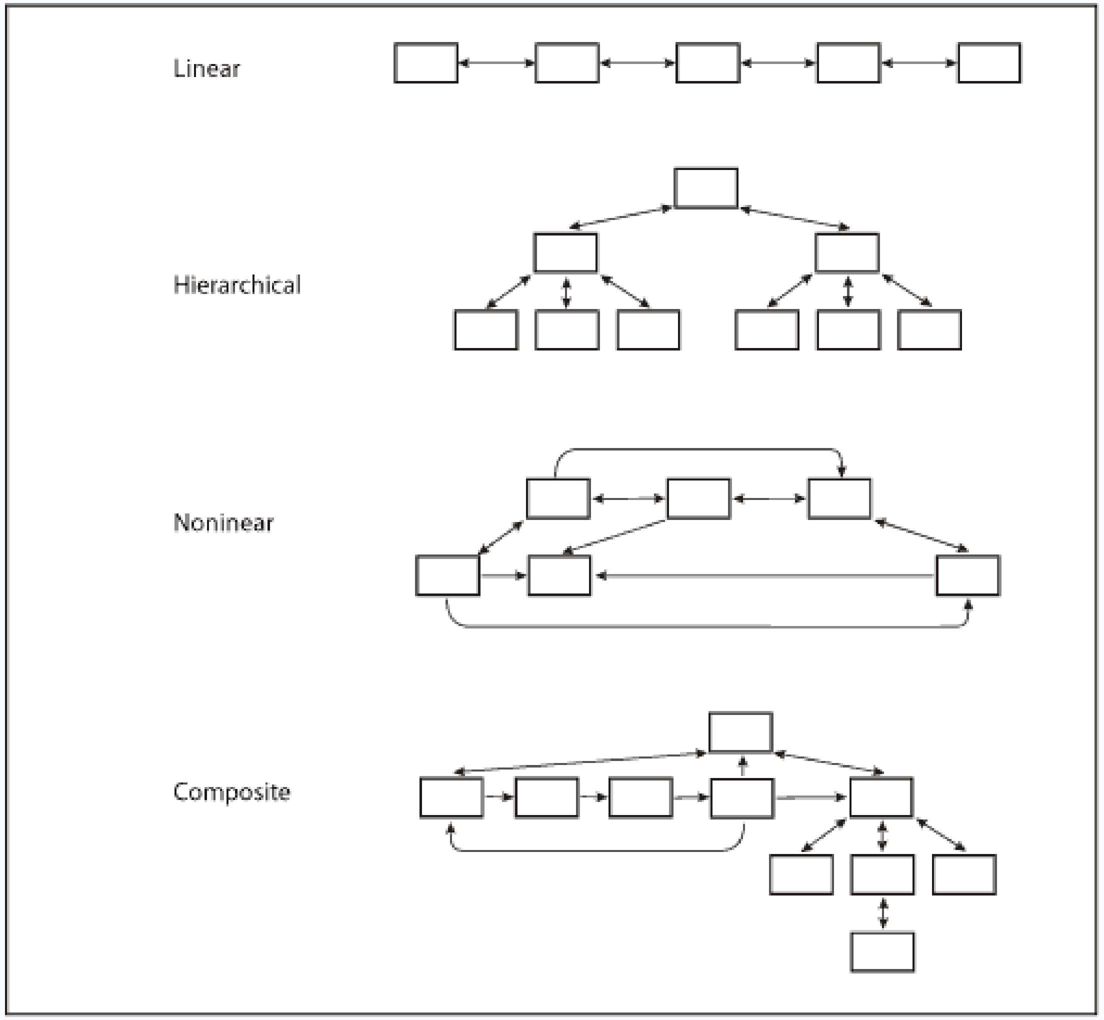
A sitemap is the visual representation of the website’s structure.



Types of Sitemap

- **Linear sitemaps**: users can navigate sequentially from one web page to another, and they can't jump to any page they want. Examples include online survey, IQ tests and slide shows.

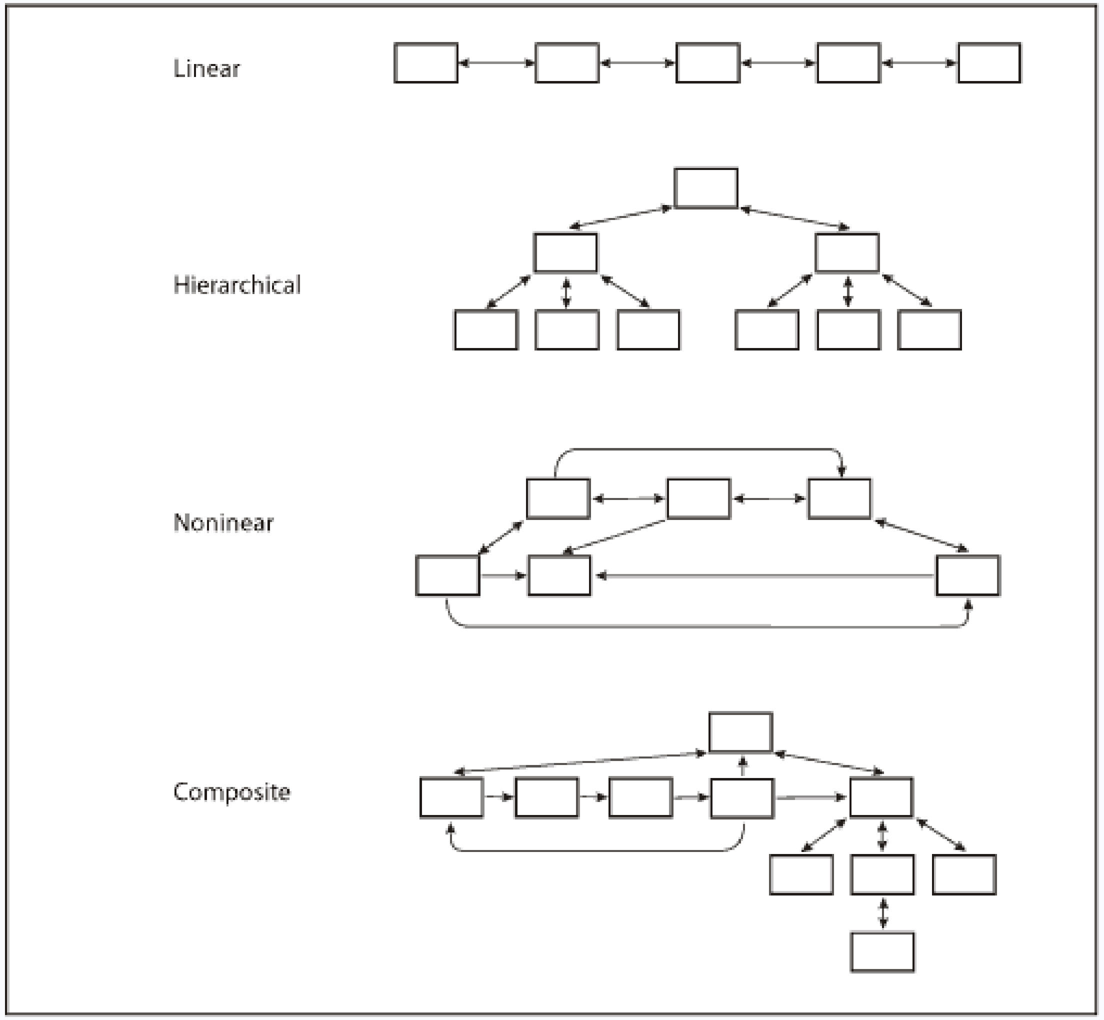
- **Hierarchical sitemaps**: users can navigate along the branches of a tree structure that is shaped by natural logic of the content. Bilingual websites are examples of this.



Types of Sitemap

- **Nonlinear sitemaps:** users navigate freely through the content of the website unbound by predetermined routes. Most majority of existing websites are nonlinear.

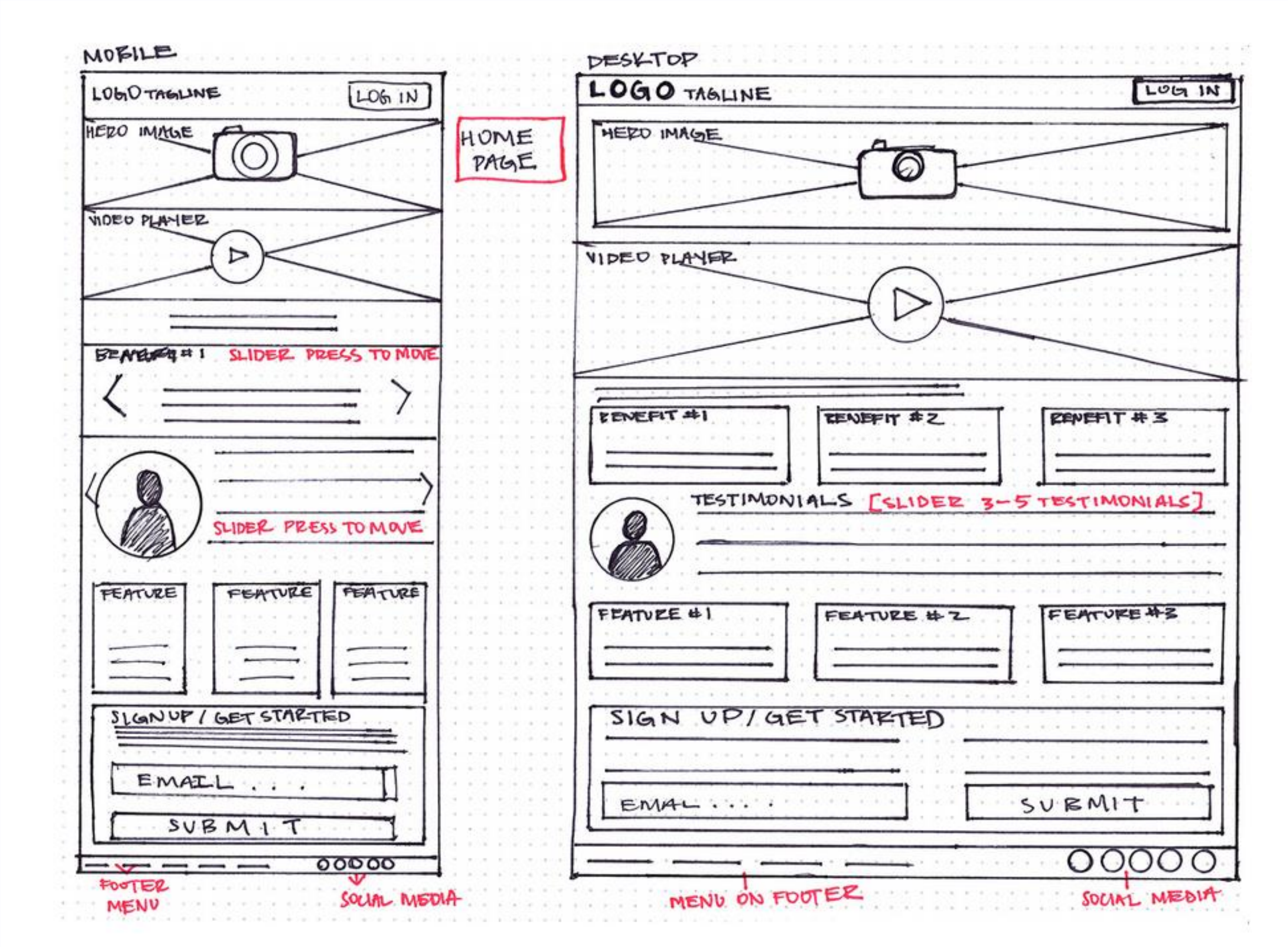
- **Composite sitemaps:** users may navigate freely (nonlinear) but are occasionally constrained to a linear process. Any website with shopping cart could be composite.



Wireframing

- A wireframe is a schematic or blueprint illustration of a software interface that specifically focuses on space allocation and prioritization of content, functionalities available, and intended behaviours.
- Wireframes typically do not include any styling, colour, or graphics.
- It primarily allows you to define the information hierarchy of your design.
- Making it easier for you to plan the layout according to how you want your user to process the information.
- Instead of designing a full-blown UI design, Wireframes allow you to test your interface, find omissions and make low-cost changes.

Wireframe – An Example



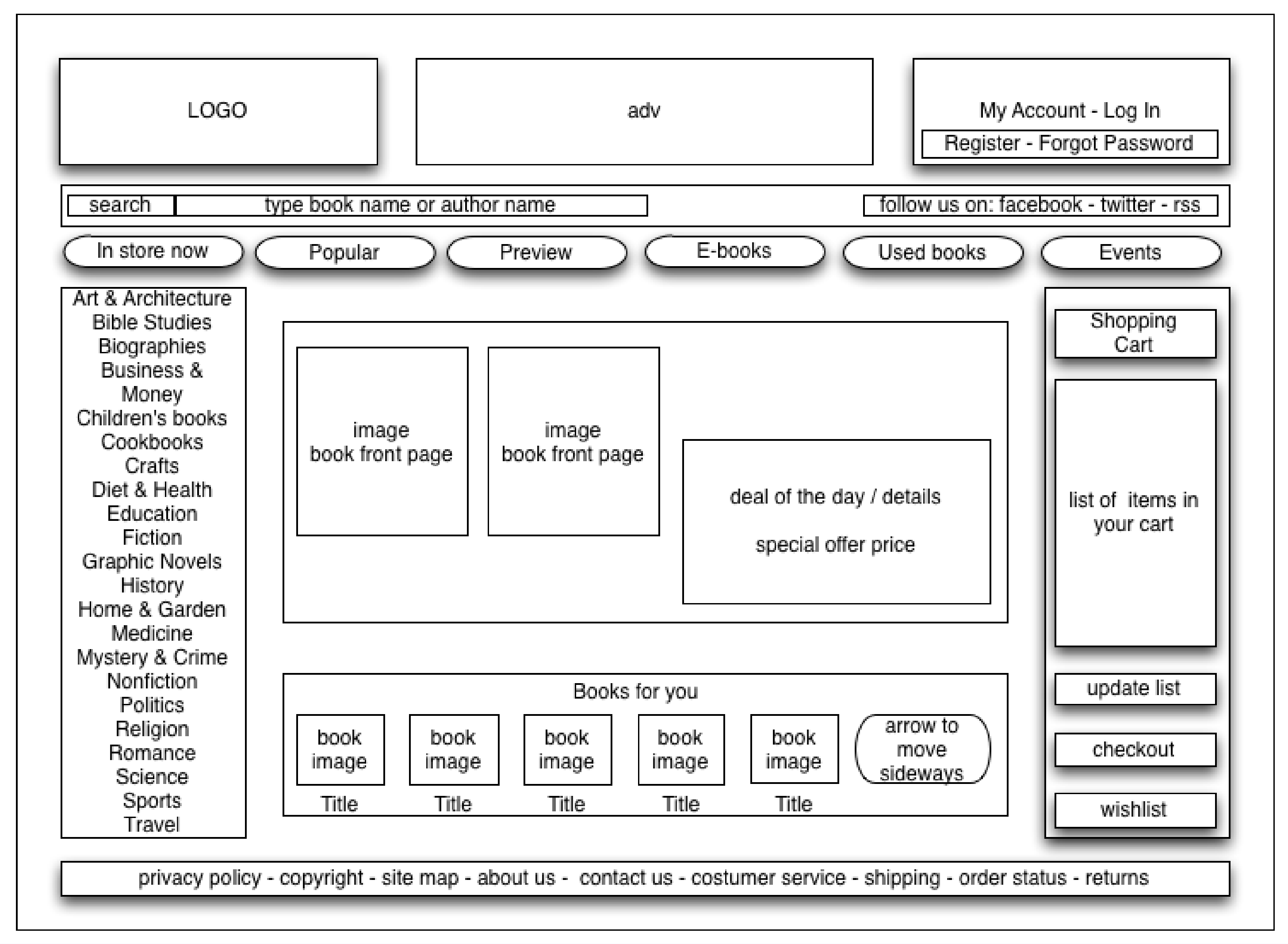
The Value of Wireframes

- Wireframes serve multiple purposes by helping to:
  - Connect the site's information architecture to its visual design by showing paths between pages
  - Clarify consistent ways for displaying particular types of information on the user interface
  - Determine intended functionality in the interface
  - Prioritize content through the determination of how much space to allocate to a given item and where that item is located

Elements of Wireframe

- Although wireframes differ from site to site, the following elements often are included as standard elements on wireframes:
  - Logo
  - Search field
  - Breadcrumb
  - Headers, including page title as the H1 and subheads H2-Hx
  - Navigation systems, including global navigation and local navigation
  - Body content
  - Share buttons
  - Contact information
  - Footer

Wireframe – An eCommerce Main Page Example



LOGO

adv

My Account - Log In

Register - Forgot Password

search | type book name or author name

follow us on: facebook - twitter - rss

In store now | Popular | Preview | E-books | Used books | Events

- Art & Architecture
- Bible Studies
- Biographies
- Business & Money
- Children's books
- Cookbooks
- Crafts
- Diet & Health
- Education
- Fiction
- Graphic Novels
- History
- Home & Garden
- Medicine
- Mystery & Crime
- Nonfiction
- Politics
- Religion
- Romance
- Science
- Sports
- Travel

image book front page

image book front page

deal of the day / details

special offer price

Books for you

book image Title | book image Title | book image Title | book image Title | book image Title | arrow to move sideways

Shopping Cart

list of items in your cart

update list

checkout

wishlist

privacy policy - copyright - site map - about us - contact us - costumer service - shipping - order status - returns

Types of Wireframes

- Wireframes can vary both in their production, from paper sketches to computer-drawn images and in the amount of detail that they convey. Low and high-fidelity are terms used to identify the level of wireframe production or functionality.

- **Low-fidelity wireframes** help facilitate project team communication and are relatively quick to develop. They tend to be more abstract because they often use simple images to block off space and implement mock content, or Latin (lorem ipsum) text as filler for content and labels.

- **High-fidelity wireframes** are better for documentation because of their increased level of detail. These wireframes often include information about each particular item on the page, including dimensions, behavior, and/ or actions related to any interactive element.
  - Sometimes referred to as Mockups that provide an accurate graphical representation (or screenshot) of a final user interface.

How to Create a Simple Wireframe

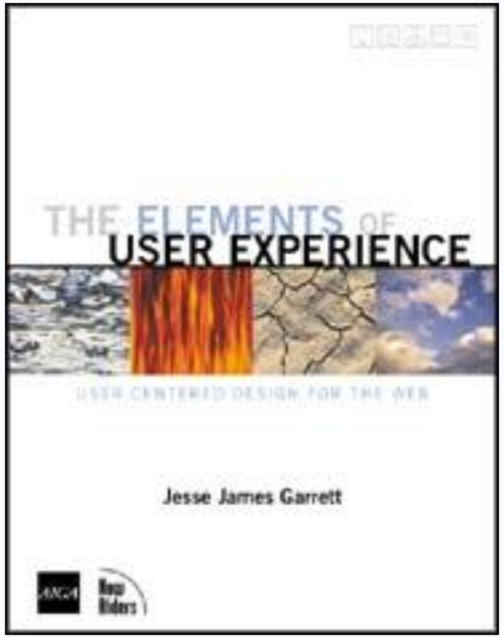
1. Identify the business goal of the website/app
2. Understand the user needs/flow of activities
3. Determine your website/mobile wireframe size
4. Begin your UI design with a low-fidelity website/app wireframe sketches.
5. Determine conversion points - decide exactly how the user should move through each step
   - At this stage, you'll determine what buttons, hyperlinks, images, or other elements on the page will guide the reader onto the next step until they reach the end goal that we discussed in step 1.
6. Remove redundant steps – Wireframing is an iterative process. It's rare to do a single round of sketching wireframes that are production-ready.
7. Get feedback on the wireframe

10 Free-to-Use Wireframing Tools

- Figma
- Penpot
- Miro
- Wireframe.CC
- Cacoo
- Jumpchart
- MockFlow
- Wirefy
- Draw.io
- Whimsical
- More Tools
- Images

Reference

- A Beginner’s Guide to Wireframing
- Website Wireframe Beginner's Guide: Processes, Tools, & Examples
- The Differences in Wireframe Fidelity: From Low to High Fidelity Wireframes



http://www.jjg.net/elements/

University of Southampton

YOUR QUESTIONS