UX Design Methods

Embedded Interface Design

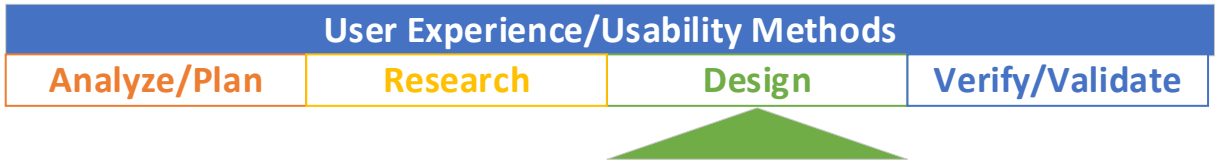
with **Bruce Montgomery**

Learning Objectives

Students will be able to…

- Understand considerations for selecting methods, including expected level of fidelity
- Review, select, and apply selected UX Design methods

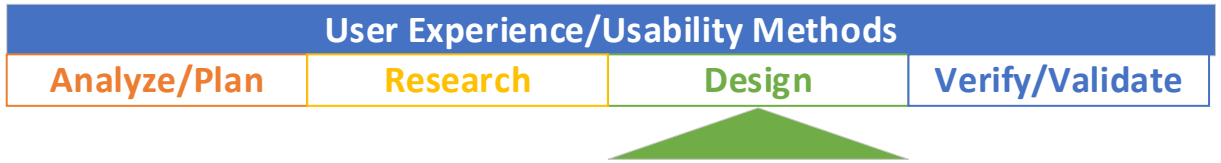
UX Design Methods



Selected Methods for Review

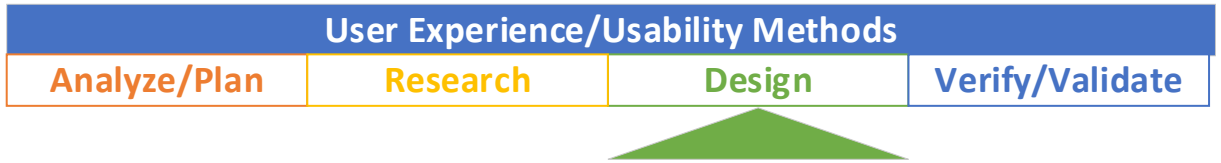
- Design Requirements
- Wireframes
- Prototypes
- Sketchboards
- Style Guides
- Paper Prototype Testing (aka Wizard of Oz Test)
- Parallel Design
- Usability Heuristics, Design Principles, Laws of UX (separate lecture)
- Sketching (separate lecture)

UX Design and Fidelity



- In use of these methods, we generally increase the fidelity of models and prototypes as we progressively elaborate our designs
- Fidelity: how closely the model resembles the final product, moving from low to higher fidelity through the design
- Aspects of fidelity which can vary in detail include:
  - Visual refinement
  - Breadth of represented functionality
  - Depth and details features
  - Interactivity and capability
  - Data model details
- Reference [1]

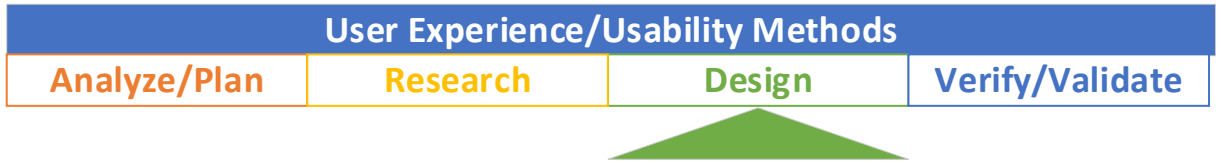
Selection Criteria



When looking at which methods are appropriate to your design cycle, consider these following method attributes

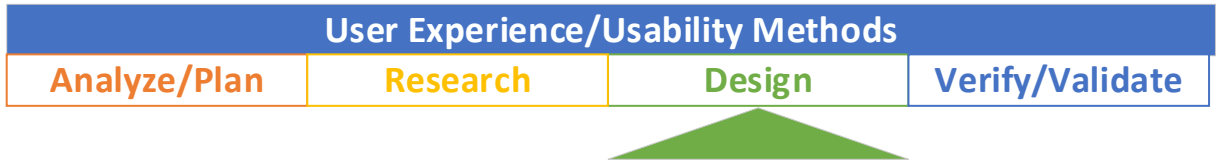
- Time (preparation, execution, follow-up)
- Complexity
- Goal (these design methods are generally focused on defining or exploring product concepts)
- Levels of fidelity – sketches are generally lower fidelity than an operating electronic prototype, for instance
- Fit to your overall project – available resources, skills, level of detail, etc

Order of Application



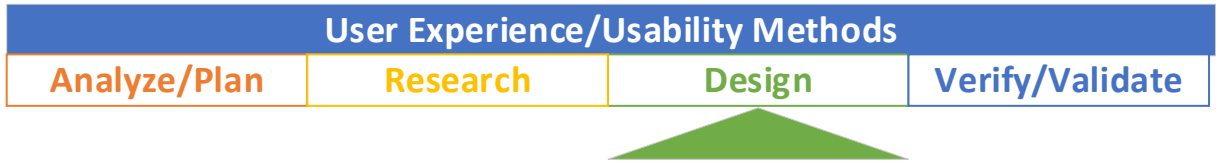
- Design methods can be used in any order that feels appropriate to your specific effort, however, they generally move towards tools capable of higher fidelity as the product design is refined
- A typical flow might move from:
  - Paper Sketches
  - Paper or Digital Wireframes
  - Digital Prototypes
  - Physical Prototypes (with increasing capability)
- Other elements, such as requirements and style guides will be used to guide design work and may also be detailed over time

Design Requirements



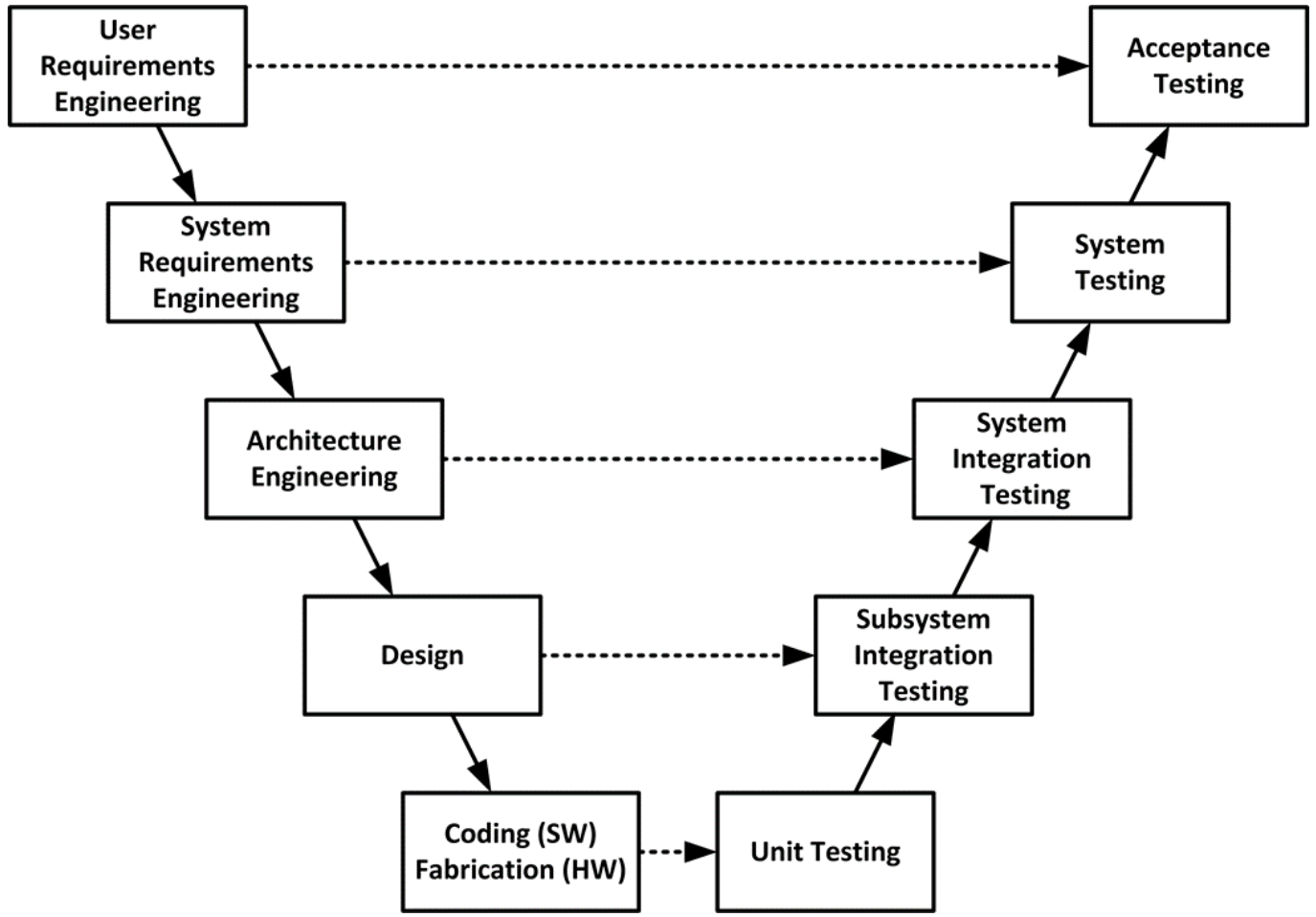
- Time: Days to weeks depending on level of detail
- Fidelity: Usually medium to high, as details of the design are considered
- Requirements gathering are an obvious step before entering into a design is writing down requirements you've gathered
- Can be done in an iterative fashion or for independent elements of a full design
- Not all development processes use formal requirements, but most capture the elements in some form, whether a standard design document or a requirements tool
- Best Practices for Requirements
  - Specific and distinct requirements
  - Often numbered or indexed for reference
  - Complete and well thought out
  - Consistent and prioritized based on the objectives
- Reference [2]

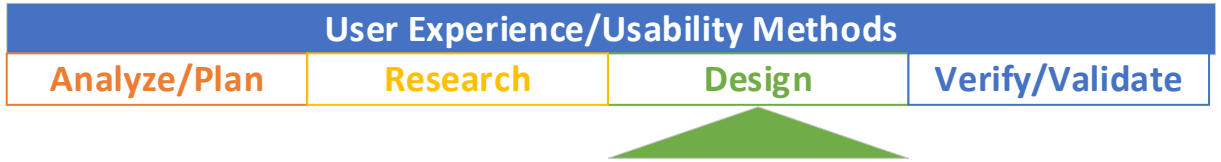
Design Requirements



- Requirements are often divided into functional (how to do), non-functional (how it works), and constraints (limits)
- Typical requirements might include
  - Business Requirements: objectives, problems to solve with the product
  - User Requirements: expectations, interactions, tasks to complete
  - Operating Requirements: details of how a product should behave
  - Quality-of-Service Requirements: characteristics to maintain effectiveness, constraints
  - Implementation Requirements: detail changes in process, team roles, migration
- Reference [2]

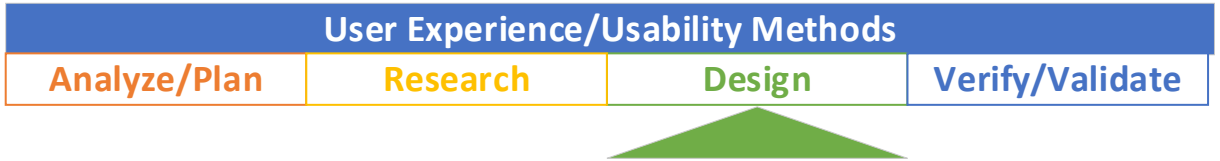
Design Requirements



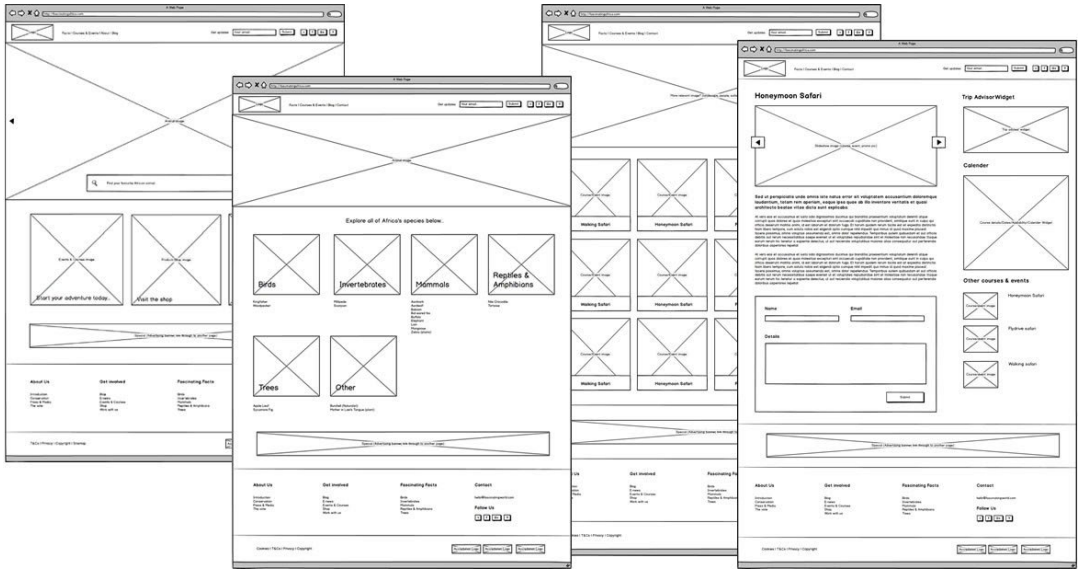


- Requirements should be verifiable
- The V diagram represents both the progressive elaboration of requirement details and their tie to different validation or verification methods
- Reference [3]

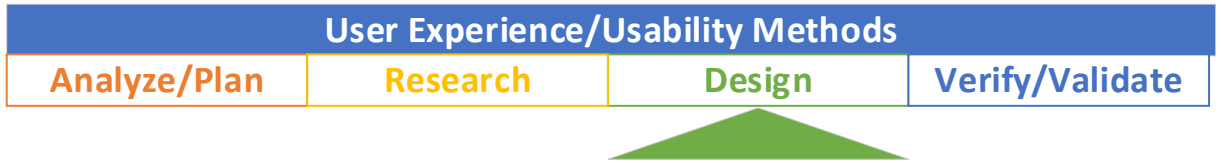
Wireframes



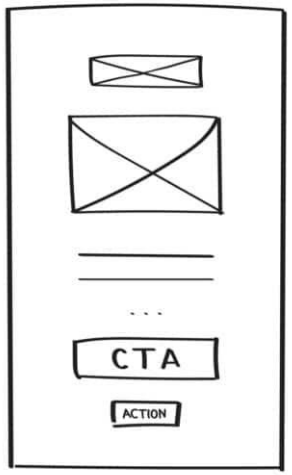
- Time: hours, days, or longer depending on details and level of review; Fidelity: varies
- Determine the intended functionality in the interface; show information priority in the layout
- Can show how to navigate through tasks, and have annotation for behavior and linked elements
- Clarify consistent display approach; generally do not include style, color, or graphics
- Reference [4], image [5]

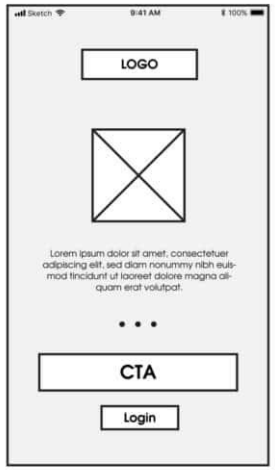


Wireframe Fidelity

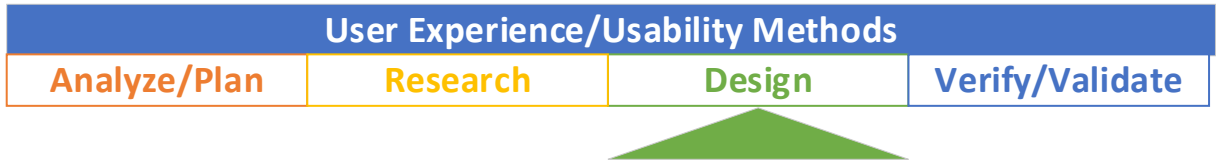


- Low-fidelity wireframes: facilitate project team communication, quick to develop, simple images to block off space; mock content, or Latin (lorem ipsum) text, as filler, not intended to represent final design
- High(er)-fidelity wireframes: better for design documentation, increased detail but still a step away from prototypes, may include notes on dimensions, behavior, actions of elements
- Reference [4], Image [6]



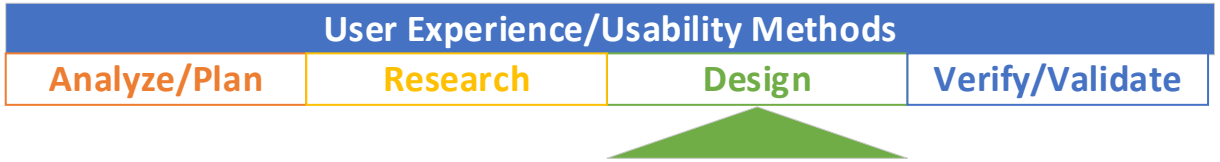


Prototypes



- Time: Varies with fidelity, which also can vary from low to high
- A prototype is a draft version of a product that allows you to explore your ideas and show the intention behind a feature or the overall design concept to users before investing significant time and money into development
- A prototype can be anything from paper drawings (low fidelity) to something with operation of some features to a fully functioning device (high fidelity):
  - "Looks like" – Prototypes look like the eventual product – Paper, 3D prints, etc.
  - "Works like" – Verify the functionality with easily added and configured elements – single board computers (SBCs), microprocessor development kits, breadboards or early printed circuits
- Reference [7]

Sketches vs. Prototypes

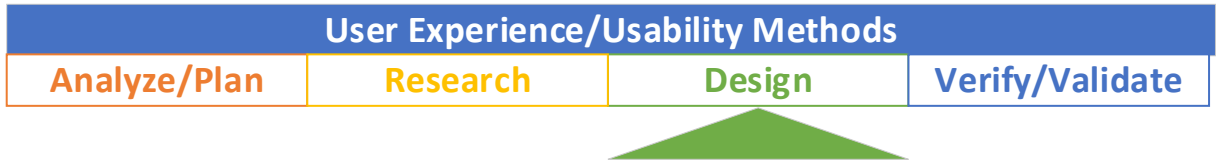


Sketch	Prototype
Suggest	Describe
Explore	Refine
Question	Answer
Propose	Test
Provoke	Resolve
Tentative	Specific
Non-committal	Depiction

- Sketching, which we'll discuss in a separate lecture, is generally more exploratory than a prototype
- Prototypes usually begin to lead to a more refined version of a product model
- Reference [7]

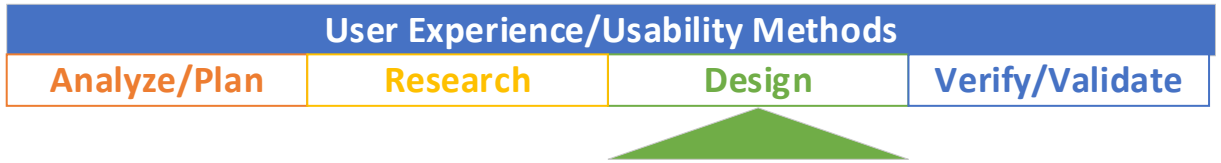
Sketchboards

- Time: Iterative over project; Fidelity: Low
- Provides a view of sketches of an overall system [8]
- A public and accessible posting of sketch work plus annotation, flow, and comments



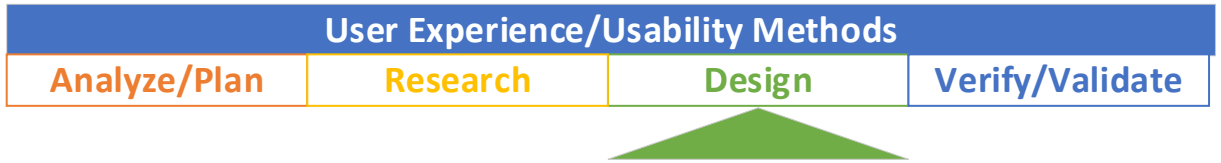
- Intended to be a focus for reviews and updates for an entire team
- Works best with scheduled reviews
- No “butts in seats” – everyone engaged
- Maintain scope for design issues and specific focus/questions
- Buley’s key approach to involving others in UX

Style Guide



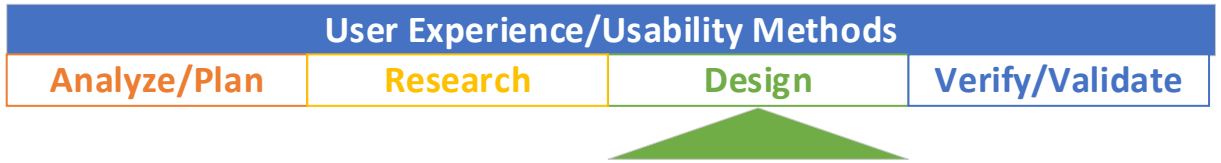
- Time: Depends on detail level of documentation and support; Fidelity: higher
- It is recommended during design to document UI standards and guidelines in a specification often called a “style guide” – an element, for example, of the UXPA UX process
  - These are often for GUI or Web elements, but could be provided for a device as well
- Example: Material Design is essentially a UI style guide for the web designed by Google [9]
  - It includes design guidelines, icons, components, tools, tutorial videos
- Example: Material Design Lite [10]
  - “It is a cross-browser, cross-OS web developer’s toolkit that can be used by anyone who wants to write more productive, portable, and — most importantly — usable web pages.”
  - Simple HTML, Javascript and CSS implementation
  - Typical Web UI Components [11]
  - Easy to include [12]
  - CSS Theme Builder [13]
- 193 Other Examples of Style Guides [14]

Paper Prototype Tests



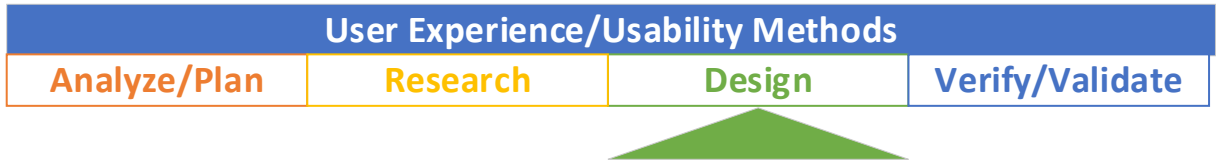
- Time: hours to prepare and perform, works with any fidelity of design element
- Process: define tasks to have the user perform, create appropriate paper prototype, test the tasks against the prototype, with the role of the computer or device stating what would happen based on the user's actions
- Roles for test: Computer or Device, User, Facilitator, Observer(s)
- Focus on key tasks to be validated
- Conduct pilot tests to ensure good outcomes
- Process described in [15]
- aka "Wizard of Oz" testing [16]
- For a complete handbook of the process, see the excellent book by Snyder called Paper Prototyping [17]

Parallel Design



- Time: One or two hours, including review; Fidelity: Low
- Process: several people create an initial design from the same set of requirements [18]
- Each designer works independently and shares their finished concepts with the group
- Design team considers each solution, and each designer uses the best ideas to further improve their own solution
- Focuses on ideation; allows a range of ideas to be generated quickly
- Cycling iteratively on this method allows best ideas from designs to move into final solutions
- Ideally, teams should have equivalent skills
- Clarity needed for fidelity of designs and evaluation approach
- Similar processes: Six-Up or a Six To One [8], [19]

Summary



- There is a lot to digest here – many methods, appropriate at different points in an iterative design cycle
  - As with all the discussions in the Embedded Interface Design lecture series, take some time to consider which tools appeal to you, and which you should explore further
  - Consider what you'll get from using each method, what fidelity is needed when, and how best to keep the user's needs present in the design
- We'll follow up with two more UX Design methods, both of which are extremely effective – usability heuristics and related design approaches, and some various ways to perform sketching – some of which may surprise you…

References

[1] Prototyping for Designers, McElroy, 2017, O'Reilly
[2] https://www.usability.gov/how-to-and-tools/methods/requirements.html
[3] https://insights.sei.cmu.edu/sei_blog/2013/11/using-v-models-for-testing.html
[4] https://www.usability.gov/how-to-and-tools/methods/wireframing.html
[5] https://medium.com/@robertsmith_co/validating-your-product-design-ideas-with-low-fidelity-wireframes-fba03b84af23
[6] https://mentormate.com/blog/low-fidelity-wireframes-vs-high-fidelity-wireframes/
[7] https://www.usability.gov/how-to-and-tools/methods/prototyping.html
[8] User Experience Ream of One, Buley, 2013, Rosenfeld
[9] https://material.io/
[10] https://getmdl.io/
[11] https://getmdl.io/components/index.html
[12] https://getmdl.io/started/index.html
[13] https://getmdl.io/customize/index.html
[14] http://styleguides.io/examples.html
[15] http://uxify.net/info/2015/workshops/Paper-Prototyping.pdf
[16] http://www.usabilitynet.org/tools/wizard.htm
[17] Paper Prototyping, Snyder, 2003, Morgan Kaufmann
[18] https://www.usability.gov/how-to-and-tools/methods/parallel-design.html
[19] http://blog.thiga.com.au/ux-design/how-to-run-a-6up