ВЫСШАЯ ШКОЛА ЭКОНОМИКИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
Faculty of Social Science, Department of Psychology

Ergonomics of Human–Machine Interaction

Alexey A. Klimov, 2017

Teaching staff

Lecturer:
Alexey Klimov
PhD in Social Psychology (2015)
Faculty of Social Science, Department of Psychology
aklimov@hse.ru

Teaching assistant:
Anna Titova
AMI 3-rd year student
aetitova@edu.hse.ru

Course info

Program of course (draft):
https://www.hse.ru/edu/courses/184770928

LMS (in progress)

Main channel:
t.me/HSEHMI2018

Credits	5
Hours	16 lecture + 16 practice
Control	Exam + Project
Pre-requisites	Basic computer programming skills in R, Python/Pandas

Formula

Final =
0,2 * (Lecture & seminar attendance) + 0.4 * (Homework) +
0.4 * (Exam)

Course topics

1. Introduction to Ergonomics
2. Defining the machine
3. Human information processing
4. Human memory
5. Usability engineering or design
6. Design rules
7. Design process rationale
8. Human-Computer Interface
9. Overview of methods of design evaluation tools
10. Basic Experiment Designs
11. Analysis Experimental Design
12. Analysis Experimental Data
13. Gather user needs
14. Study of User Preferences by asking right questions
15. Review and final project (homework) description

Please check what's missing in program, here: https://www.hse.ru/edu/courses/184770928

References

Main:

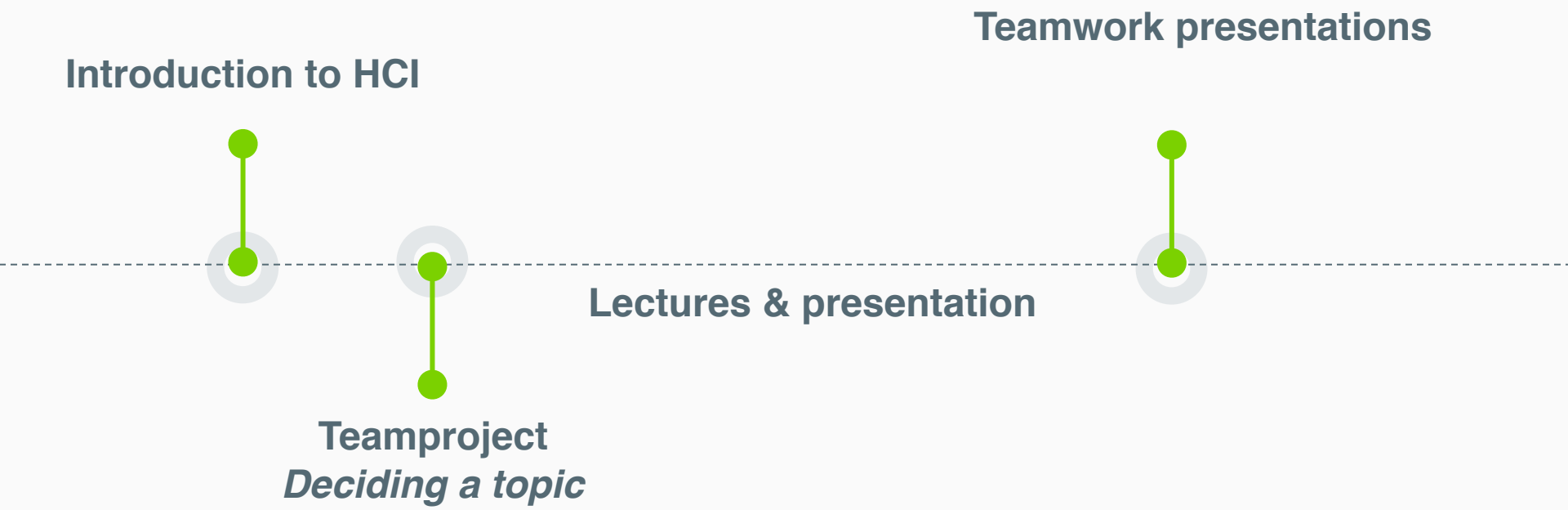
A. Dix, et al
Human Computer Interaction 3rd Ed., Prentice Hall, 2005

Additional

- CS, Stanford, "Introduction to Human Computer Interaction Design“
- CS, UC Berkeley, "User Interface Design, Prototyping, and Evaluation“

Special topics have required readings (like R-programing, statistical tests, UI guidelines etc.)!

Course outline



Course content will facilitate your project, but you can go step ahead

Project

Team project (you should do it in a team of two)

Make HCI study:

1. Find some UI / product
2. Make prototype
3. Propose enhancement
4. Measure change in different metrics

Main options

Project sources:

- Your own project (course project, part-time project)
- Market project (see habrahabr)

UI types:

- Desktop
- Web
- Mobile
- Static

Project

Team project (you should do it in a team of two)

Make HCI study:

1. Find some UI / product
2. Make prototype
3. Propose enhancement
4. Measure change in different metrics

UI/UX testing

Prototypes can be created with:

- Sketch
- UXPin
- Balsamiq / Axure (for mocks)
- or Photoshop?!

For review see:
https://uxdesign.cc/the-best-prototyping-tools-8d7dc5c8ee27

Project

Team project (you should do it in a team of two)

Make HCI study:

1. Find some UI / product
2. Make prototype
3. Propose enhancement
4. Measure change in different metrics

UI/UX testing

Metrics can be fetch by:

- Focus Groups
- Task analysis
- Gathered by GA, YM (like financial, e.g. ARPU)

Project

Team project (you should do it in a team of two)

Make HCI study:

1. Find some UI / product
2. Make prototype
3. Propose enhancement
4. Measure change in different metrics

In case of static (semi-static) computer interface

Two SMI eye trackers:
SMI RED-m and SR Research Eyelink 1000+

ВЫСШАЯ ШКОЛА ЭКОНОМИКИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
Faculty of Social Science, Department of Psychology

What is ergonomics?

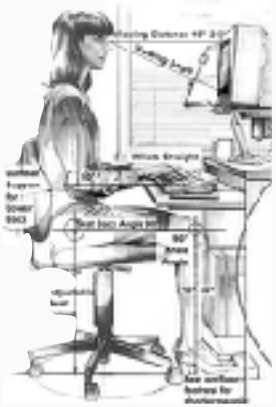
Alexey A. Klimov, 2017

Ergonomics

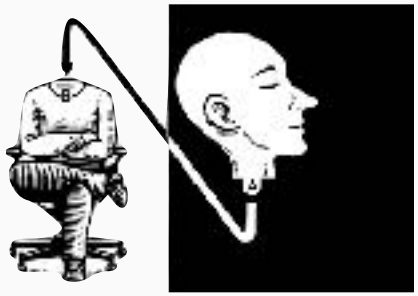
Ergonomics (or human factors) is the scientific discipline concerned with the understanding of interactions among humans and other elements of a system, and the profession that applies theory, principles, data and methods to design in order to optimize human well-being and overall system performance

Domains

- Physical ergonomics



- Cognitive ergonomics



- Organizational ergonomics

Human-Computer Interaction Field

Field structure are complex:
Computer interfaces (UI/UX, Human-Computer Interaction) — computer, interface
Design of devices, tools (Human-Machine Interaction) — machine, design
Human limitations (Human Factors)

Human-Computer Interaction

Major part of work for “real” programs approximately 50%

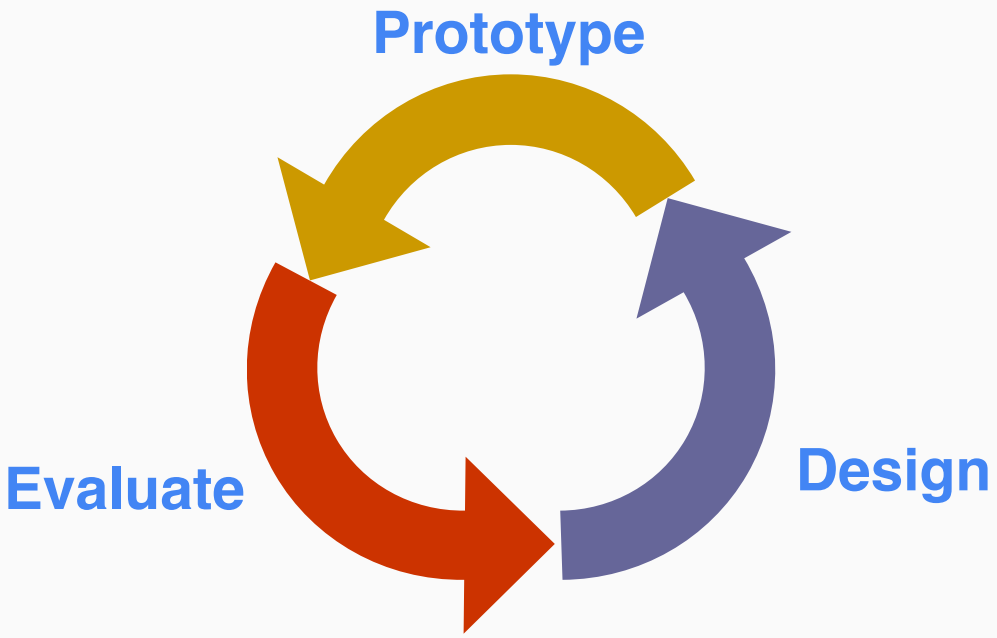
User interfaces are hard to get right

“Real” software is intended for people other than yourself

Bad user interfaces cost money (5% satisfaction -> up to 85% profits)

Main steps

- **Design**
- **Prototyping**
- **Evaluation**
- Implementation (of UIs)



Multi-role approach

Artist -

You need to care about the beauty to the user

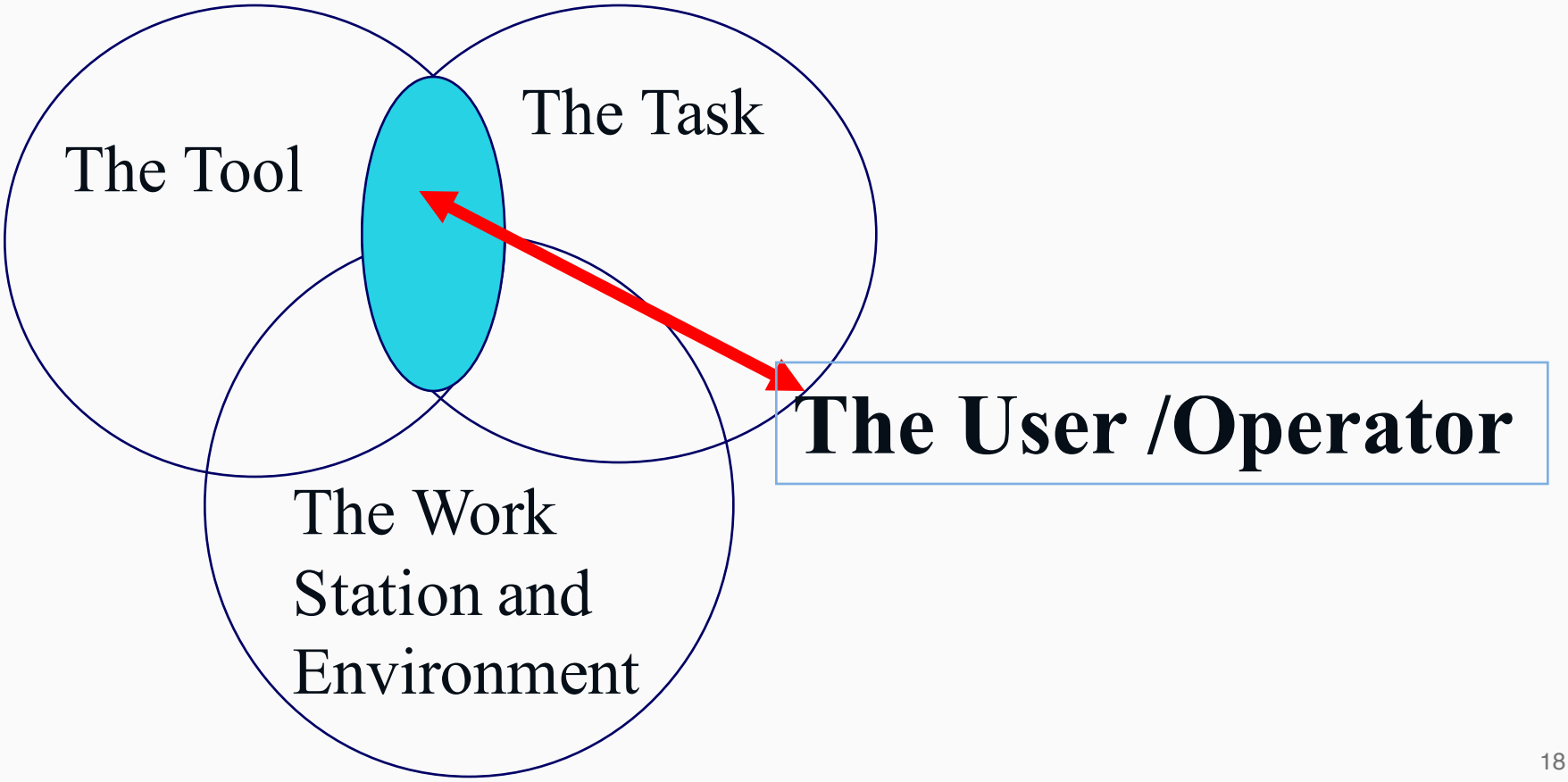
1. Design
2. Carry out
3. Analyze
4. Experiment

Designer

We need to get the fit between engineering and peoples need (UR)

Engineer-

Ergonomics focus



Main components

- Human
  - the end-user of a program
  - the others in the organization
- Computer
  - the machine the program runs on
  - often split between clients & servers
- Interaction
  - the user tells the computer what they want
  - the computer communicates results

Methods

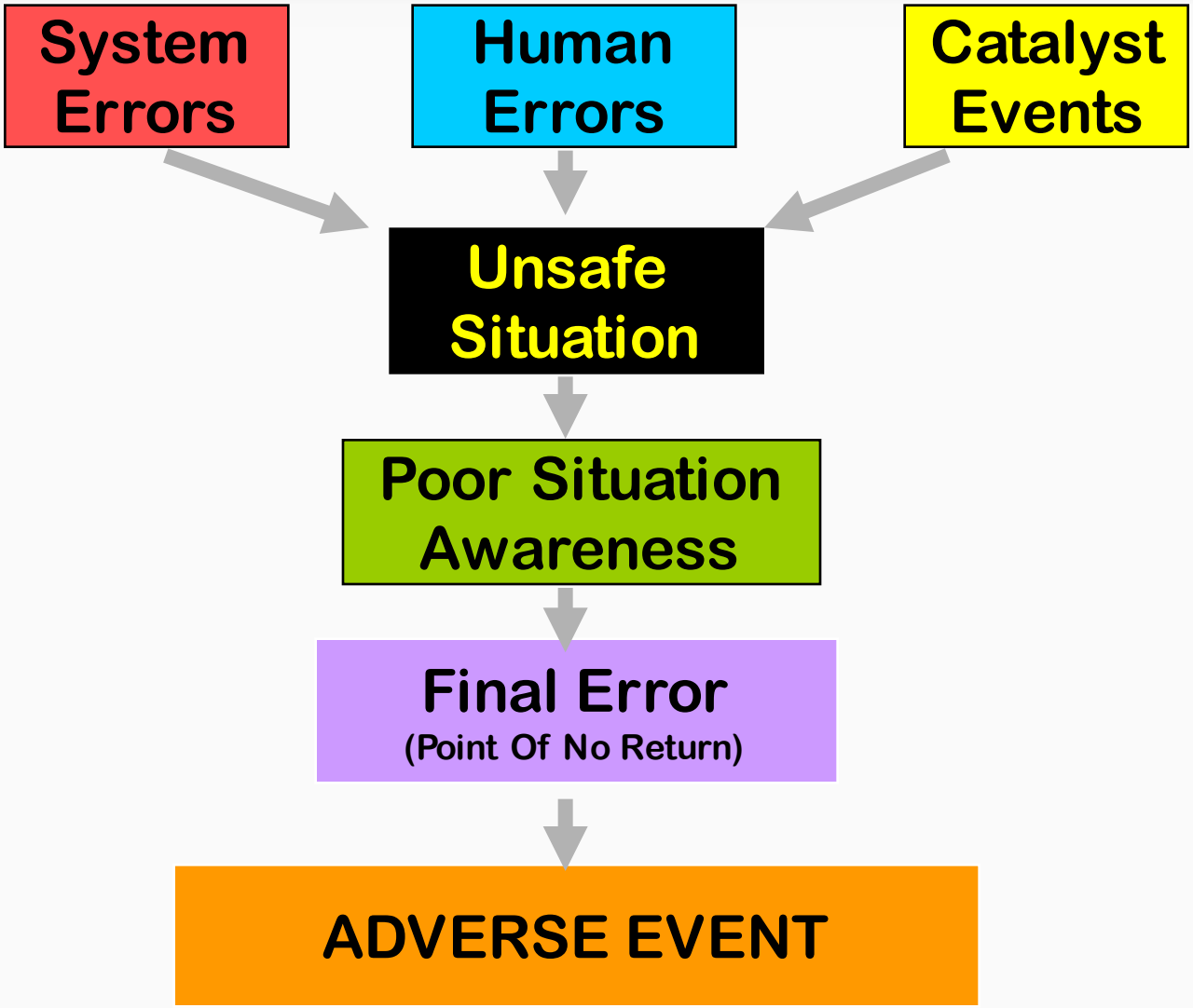
The limitations include (Dumas, 2006):

1. Usually take more time and resources than other methods
2. Very high effort in planning, recruiting, and executing compared with other methods
3. Much longer study periods and therefore requires much goodwill among the participants
4. Studies are longitudinal in nature, therefore, attrition can become a problem

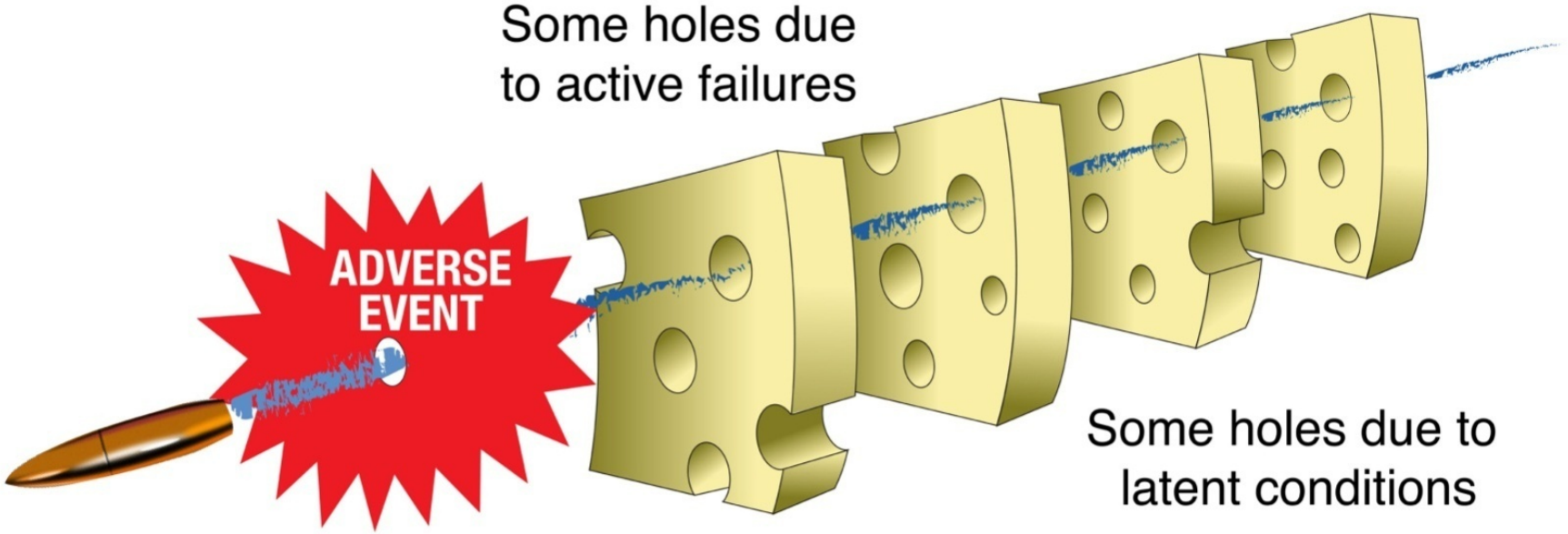
Examples of common methods:

- Focus Groups
- Iterative design (prototyping)
- Task analysis
- Work sampling

Error Chains



Swiss Cheese Model



Computer definition is easy?

- The devices dictate the styles of interaction that the system supports
- If we use different devices, then the interface will support a different style of interaction

The Computer

- input devices – text entry and pointing
- output devices – screen (small&large), digital paper
- virtual reality – special interaction and display devices
- physical interaction – e.g. sound, haptic, bio-sensing
- paper – as output (print) and input (scan)
- memory – RAM & permanent media, capacity & access
- processing – speed of processing, networks

Computer definition is easy?

1. computers in your pockets?
2. computers in your house?

Numeric keypads

WHY?!

http://ux.stackexchange.com/questions/16666/why-do-numpads-on-keyboards-and-phones-have-reversed-layouts

http://www.wimp.com/why-phone-buttons-are-laid-out-the-way-they-are/

Methods

The limitations include (Dumas, 2006):

1. Usually take more time and resources than other methods
2. Very high effort in planning, recruiting, and executing compared with other methods
3. Much longer study periods and therefore requires much goodwill among the participants
4. Studies are longitudinal in nature, therefore, attrition can become a problem

Examples of common methods:

- Focus Groups
- Iterative design (prototyping)
- Task analysis
- Work sampling

ВЫСШАЯ ШКОЛА ЭКОНОМИКИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
Faculty of Social Science, Department of Psychology

Design

Alexey A. Klimov, 2017

Design

Examples?

- Industrial design
- Display designs

- Product Design
- Interface Design
- Visual Design

https://medium.com/@markymark/three-types-of-design-3623c3243aa6

Design

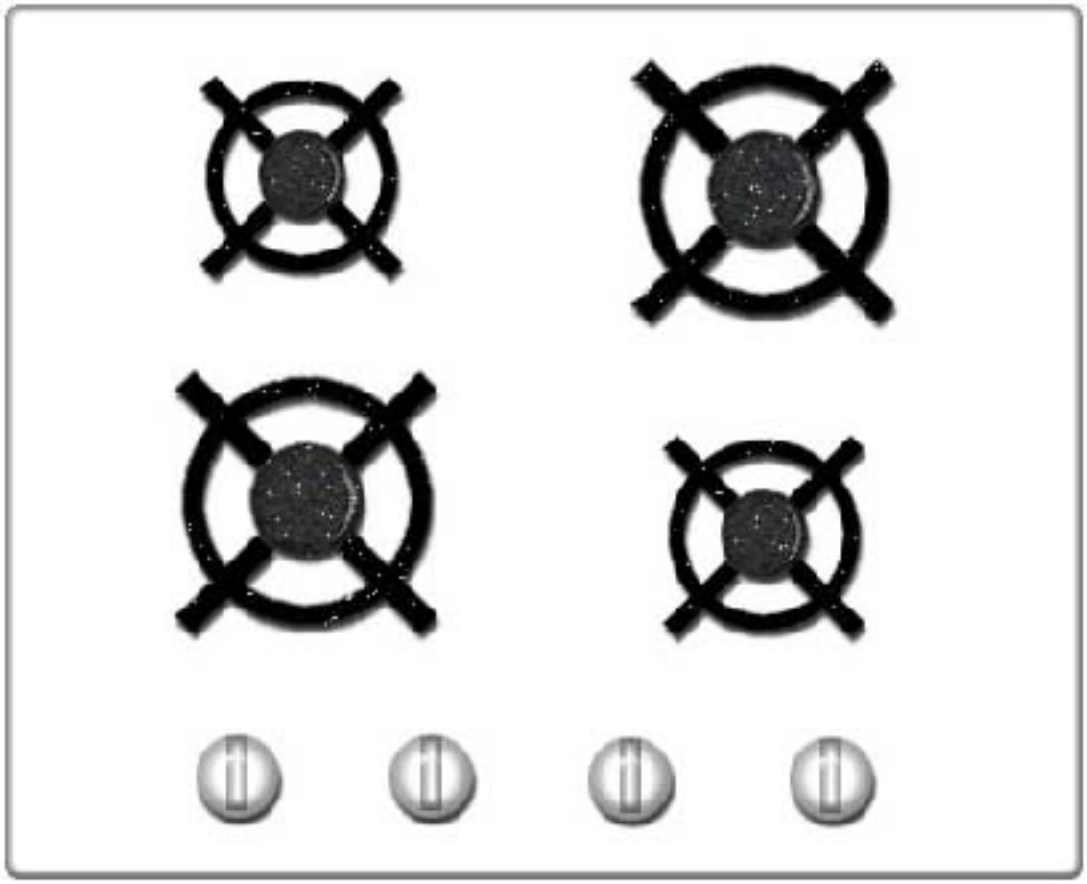
Typology

- Responsive
- Adaptive
- Optimistic
- Participative
- …
- ???

Design

Examples?

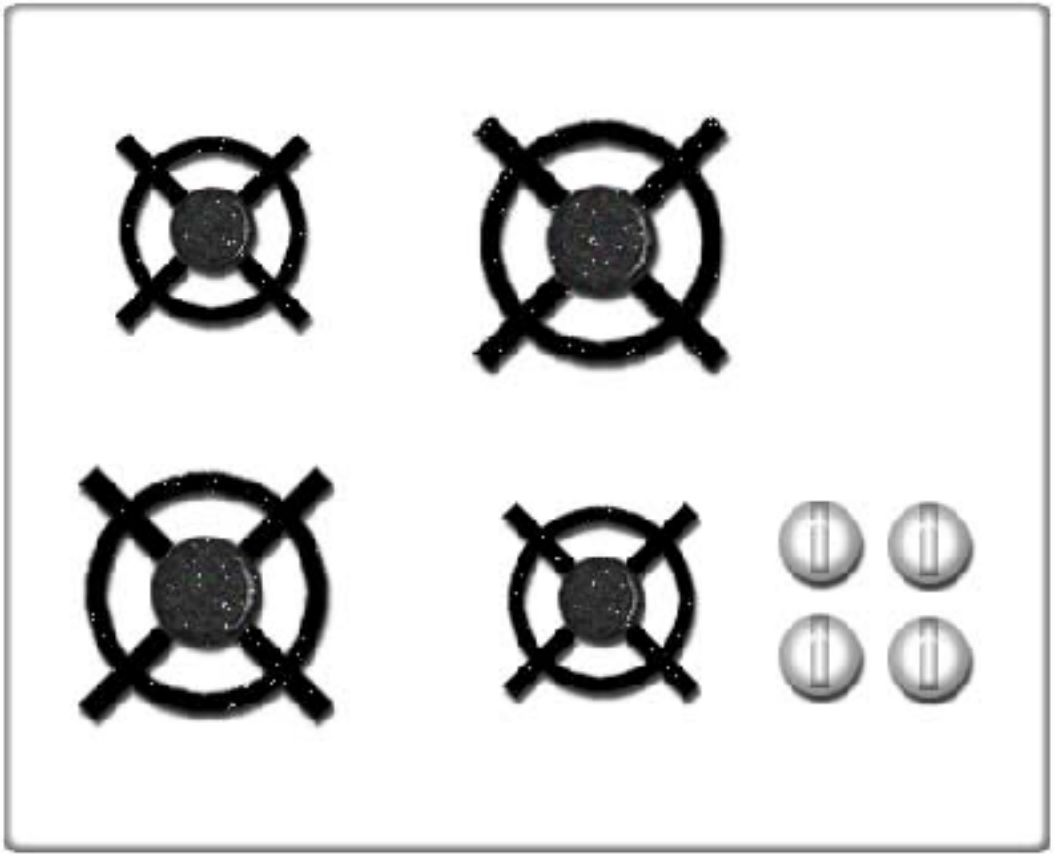
Industrial design



Design

Examples?

Industrial design



Jef Raskin in *The Humane Interface*

- First Law: A computer shall not harm your work or, through inactivity, allow your work to come to harm
- Second Law: A computer shall not waste your time or require you to do more work than is strictly necessary

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Design rationale is complex topic!

- the reasons behind a design decision,
- the justification for it,
- the other alternatives considered,
- the trade offs evaluated, and
- the argumentation that led to the decision.

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Capture methods

- Reconstruction
- Record-and-replay
- Methodological byproduct (bypass?)
- Automatic Generation

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Rationale representation

Process-oriented design rationale:

- IBIS

Structure-oriented design rationale:

- QOC (Questions Options and Criteria)
- Psychological design rationale

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Process-oriented design rationale representation

The primitives of this system are:

- **Issues**: questions that the design or argument is addressing.
- **Positions**: potential resolutions of an issue.
- **Arguments**: that support or refute a position.

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Process-oriented design rationale representation (IBIS)

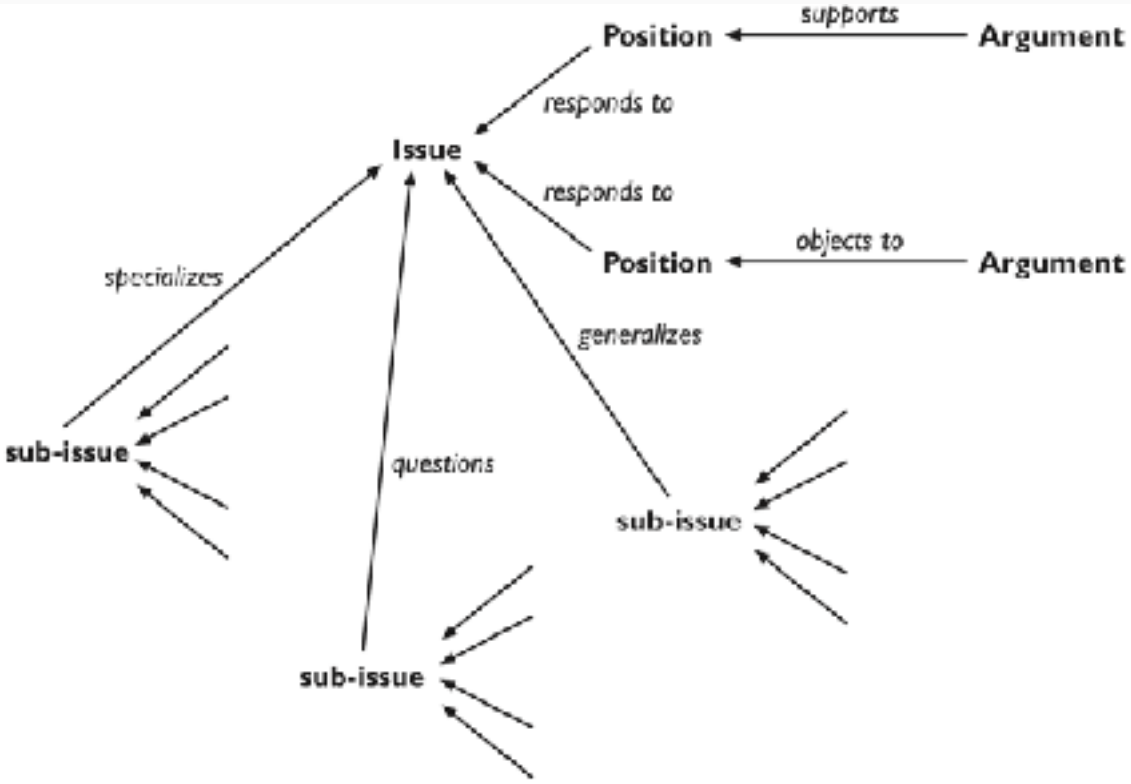


Figure 6.8 The structure of a gIBIS design rationale

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Structure-oriented design rationale

The primitives of this system are:

- **Issues**: questions that the design or argument is addressing.
- **Positions**: potential resolutions of an issue.
- **Arguments**: that support or refute a position.

Design rationale

A design rationale is an explicit documentation of the reasons behind decisions made when designing a system or artifact

Process-oriented design rationale representation (QOC)

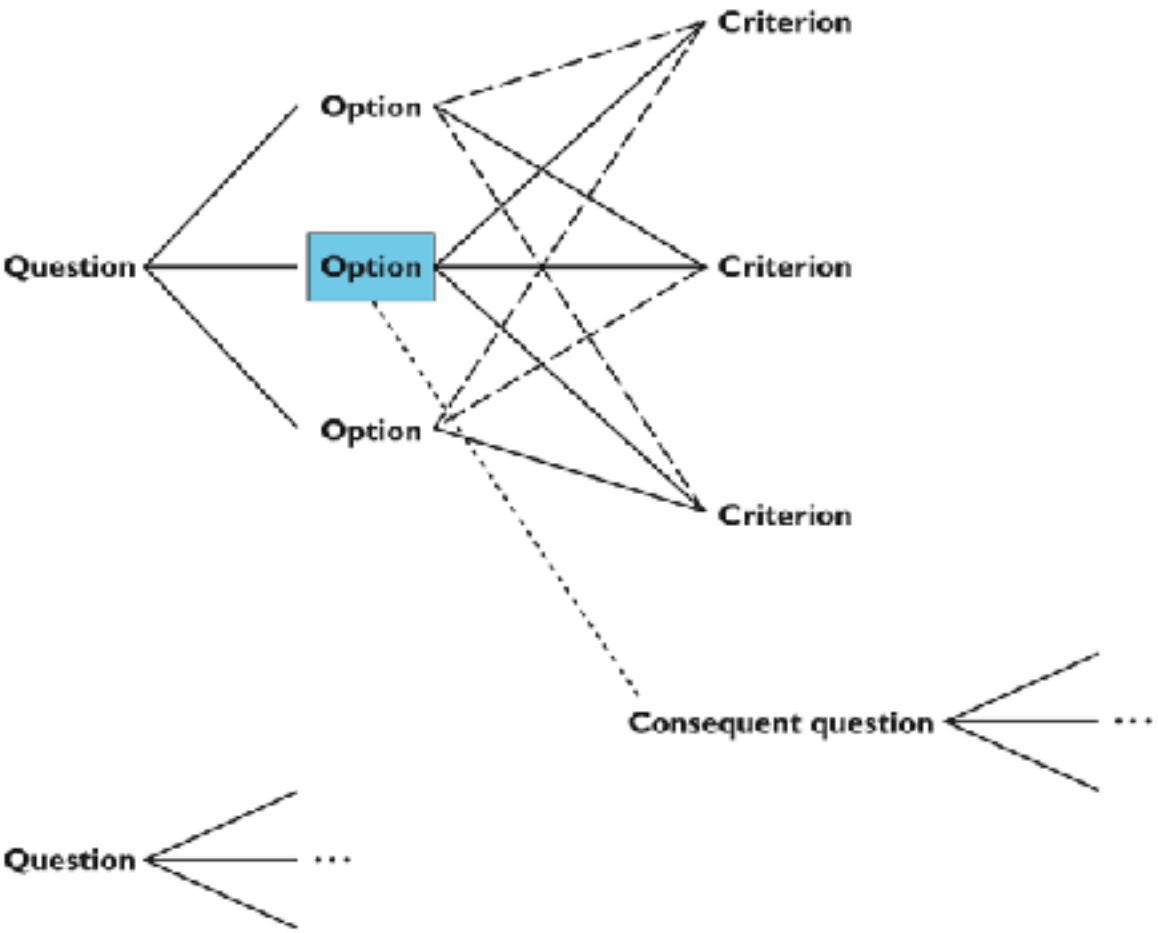


Figure 6.9 The QOC notation

Psychological design rationale

Task — artifact

The main emphasis is not to capture the designer's intention in building the artifact. Rather, psychological design rationale aims to make explicit the consequences of a design for the user, given an understanding of what tasks he intends to perform (Dix, p. 254)

Design process rationale

Burge, JE (2008). *Rationale-Based Software Engineering*. Heidelberg: Springer-Verlag

Process of design

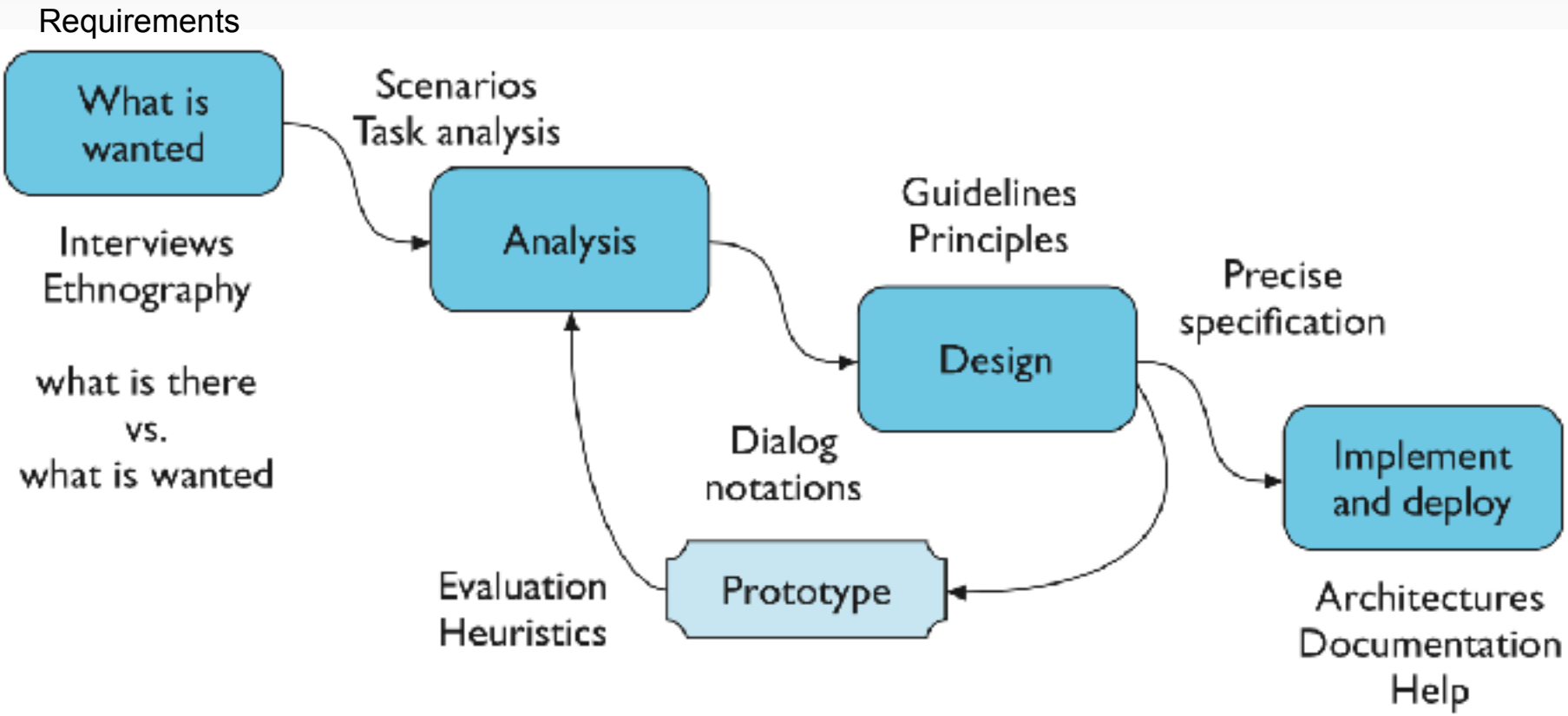


Figure 5.1 Interaction design process

Know you user (Who are they?)

Participatory design

- Focus groups
- Interviewing
- Ethnography

Know you user (Who are they?)

What to describe?

- Persona
- Scenarios

Persona – a rich description of Betty the Warehouse Manager (example)

Betty is 37 years old. She has been Warehouse Manager for five years and has worked for Simpkins Brothers Engineering for 12 years. She didn't go to university, but has studied in her evenings for a business diploma. She has two children aged 15 and 7 and does not like to work late. She did part of an introductory in-house computer course some years ago, but it was interrupted when she was promoted and could no longer afford to take the time. Her vision is perfect, but her right-hand movement is slightly restricted following an industrial accident three years ago. She is enthusiastic about her work and is happy to delegate responsibility and take suggestions from her staff. However, she does feel threatened by the introduction of yet another new computer system (the third in her time at SBE).

Scenario for proposed movie player (example)

Brian would like to see the new film *Moments of Significance* and wants to invite Alison, but he knows she doesn't like 'arty' films. He decides to take a look at it to see if she would like it and so connects to one of the movie-sharing networks. He uses his work machine as it has a higher bandwidth connection, but feels a bit guilty. He knows he will be getting an illegal copy of the film, but decides it is OK as he is intending to go to the cinema to watch it. After it downloads to his machine he takes out his new personal movie player. He presses the 'menu' button and on the small LCD screen he scrolls using the arrow keys to 'wi-fi connect' and presses the 'select' button. On his computer the movie download program now has an icon showing that it has recognized a compatible device and he drags the icon of the film over the icon for the player. On the player the LCD screen says 'downloading now', with a per cent done indicator and small whirling icon.

Scenario for proposed movie player (example)

During lunchtime Brian takes out his movie player, plugs in his earphones and starts to watch. He uses the arrow keys to skip between portions of the film and decides that, yes, Alison would like it. Then he feels a tap on his shoulder. He turns round. It is Alison. He had been so absorbed he hadn't noticed her. 'What are you watching', she says. 'Here, listen', he says and flicks a small switch. The built-in directional speaker is loud enough for both Brian and Alison to hear, but not loud enough to disturb other people in the canteen. Alison recognizes the film from trailers, 'surprised this is out yet' she says. 'Well actually . . .', Brian confesses, 'you'd better come with me to see it and make an honest man of me'. 'I'll think about it', she replies.

Standards / Guidelines

Presentation (+ detect principles, rationale, try to classify design)

- IOS: https://developer.apple.com/ios/human-interface-guidelines/
- Android: https://developer.android.com/design/index.html
- Windows Metro: https://developer.microsoft.com/en-us/windows/apps/design
- Website: https://developers.google.com/web/fundamentals/ https://design.atlassian.com ?

ВЫСШАЯ ШКОЛА ЭКОНОМИКИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
Faculty of Social Science, Department of Psychology

Примеры

www.useronboard.com/onboarding-teardowns/

Design

Interaction design

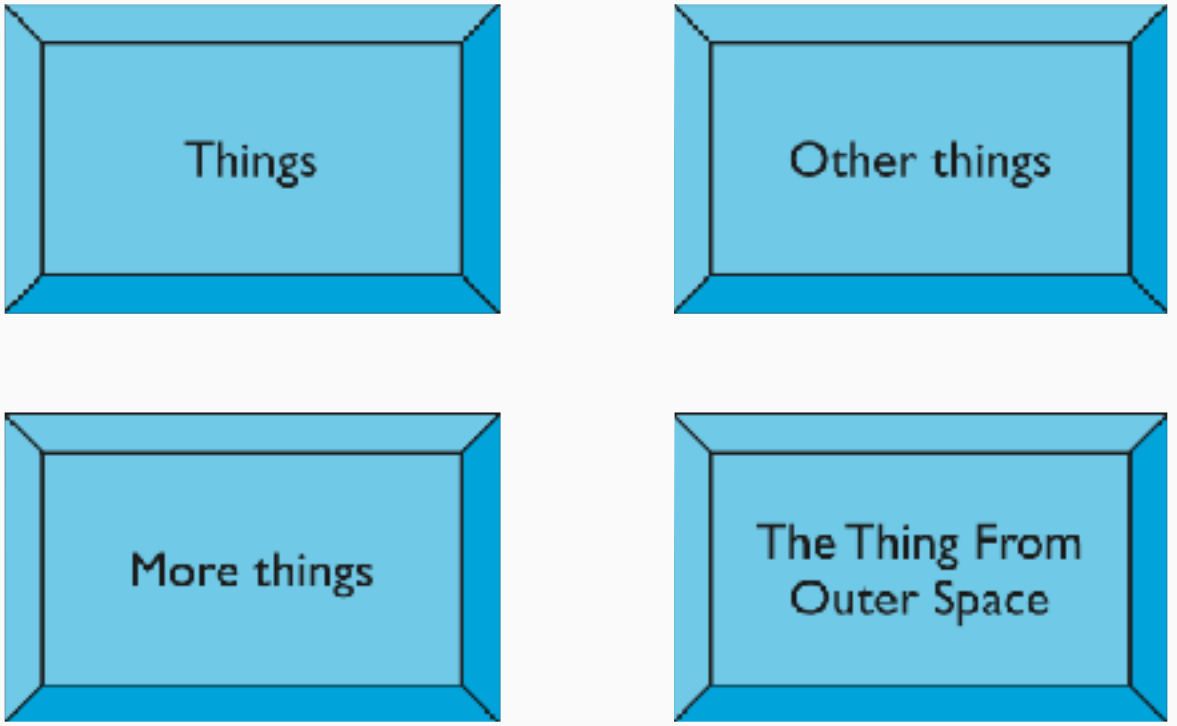
Levels of interaction

Table 5.1 Levels of interaction

PC application	Website	Physical device
Widgets	Form elements, tags and links	Buttons, dials, lights, displays
Screen design	Page design	Physical layout
Navigation design	Site structure	Main modes of device
Other apps and operating system	The web, browser, external links	The real world!

Plain design structure

«Kiosk» mode



Hierarchical organization

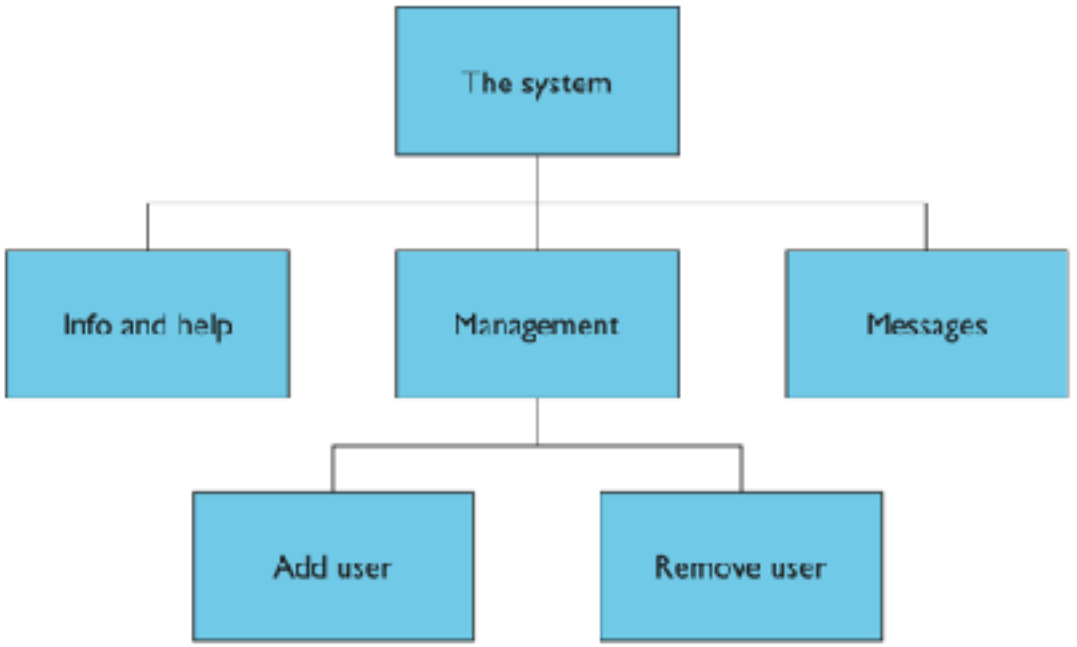


Figure 5.6 Application functional hierarchy

Dialog

Metaphor, not a modal window

- show what leads to what
- show what happens when
- include branches and loops
- be more task oriented than a hierarchy

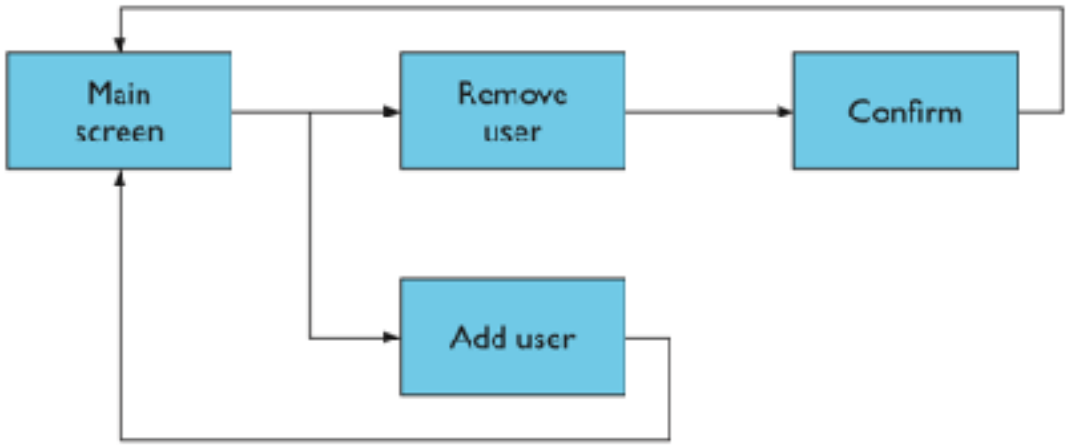
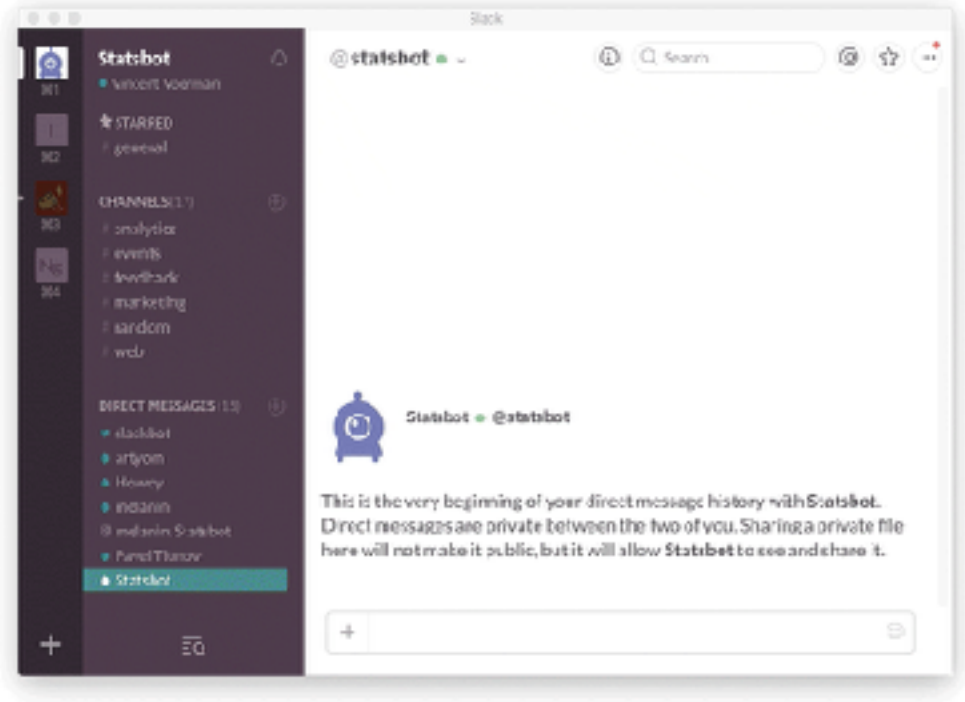


Figure 5.7 Network of screens/states



Dialog

Metaphor, not a modal window

- show what leads to what
- show what happens when
- include branches and loops
- be more task oriented than a hierarchy

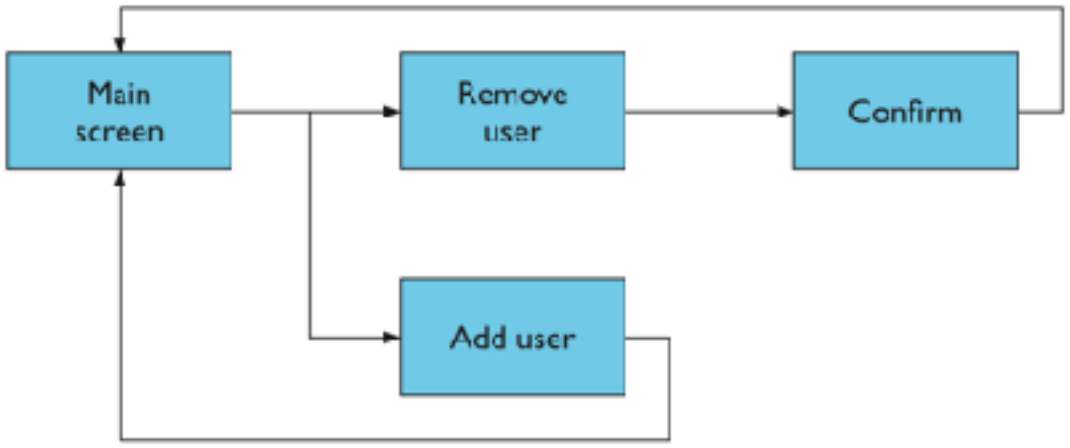
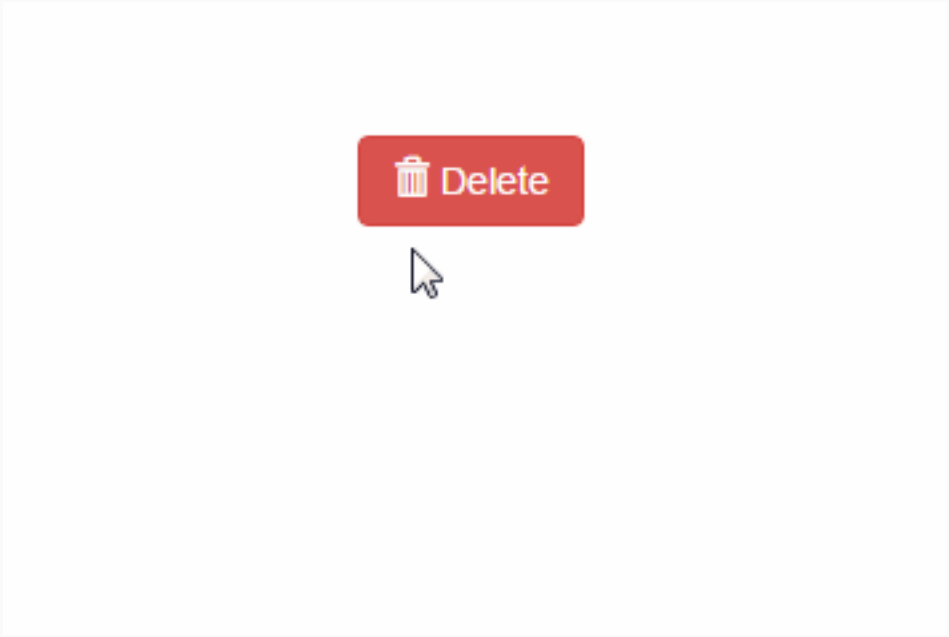


Figure 5.7 Network of screens/states



Layout (see Guidelines)

- Grouping and structure
- Order of groups and items
- Decoration
- Alignment
- White space

Interaction design in software process

Usability specification (VCR case)

Attribute:	Backward recoverability
Measuring concept:	Undo an erroneous programming sequence
Measuring method:	Number of explicit user actions to undo current program
Now level:	No current product allows such an undo
Worst case:	As many actions as it takes to program in mistake
Planned level:	A maximum of two explicit user actions
Best case:	One explicit cancel action

Whiteside, Bennett and Holtzblatt

Interaction design in software process

Make usability specification for:

- Diary app onboarding
- Calendar app

Interaction design in software process

Attribute:	Guessability
Measuring concept:	Ease of first use of system without training
Measuring method:	Time to create first entry in diary
Now level:	30 seconds on paper-based system
Worst case:	1 minute
Planned level:	45 seconds
Best case:	30 seconds (equivalent to now)

Interaction design in software process

Attribute:	Task migratability
Measuring concept:	Scheduling a weekly meeting
Measuring method:	Time it takes to enter a weekly meeting appointment
Now level:	(Time to schedule one appointment) × (Number of weeks)
Worst case:	Time to schedule two appointments
Planned level:	1.5 × (Time to schedule one appointment)
Best case:	Time to schedule one appointment

Usability engineering

1. Usability metrics

Usability metrics types

- Non-financial (cover user performance)
  - Quantitative
  - Qualitative
- Financial (cover costs and gains)

Usability metrics (quantitative)

Success Rate

- In a given scenario, was the user able to complete the assigned task?

Usability metrics (quantitative)

Error Rate

- Critical errors will prevent a user from completing a task
- Noncritical errors will simply lower the efficiency with which they complete it

Usability metrics (quantitative)

Time to Completion

- How much time did it take the user to complete the task?
- Can be particularly useful when determining how your product compares with your competitors (if you're testing both).

Usability metrics (qualitative)

Subjective Measures

- Numerically rank a user's self-determined satisfaction, ease-of-use, availability of information, etc. Surprisingly, you can actually quantify qualitative feedback by boiling this down to the Single Ease Question

Subjective Measures

Type	Example	Results
Verbal Response	Describe and demonstrate what, if anything, was most frustrating about this site.	Spoken answers correlate with where a participant is at in the study. Make great clips for a highlight reel.
Multiple Choice	Do you trust this company? • Yes • No	Great for collecting responses that are **categorical**. These can be nominal (cats or dogs?) dichotomous (yes or no) and even ordinal (**Likert** scale agree/disagree).
Rating Scale	How likely are you to return to this site again? *1 2 3 4 5* *Not at all likely* *Very likely*	Good for collecting ordinal variables (low, medium, high) and are very recognizable especially within the United States.
Written Response	What do you think is missing from this page, if anything?	Good for running post-study analysis. How many people used the same answers? Quick quotes for building user stories.

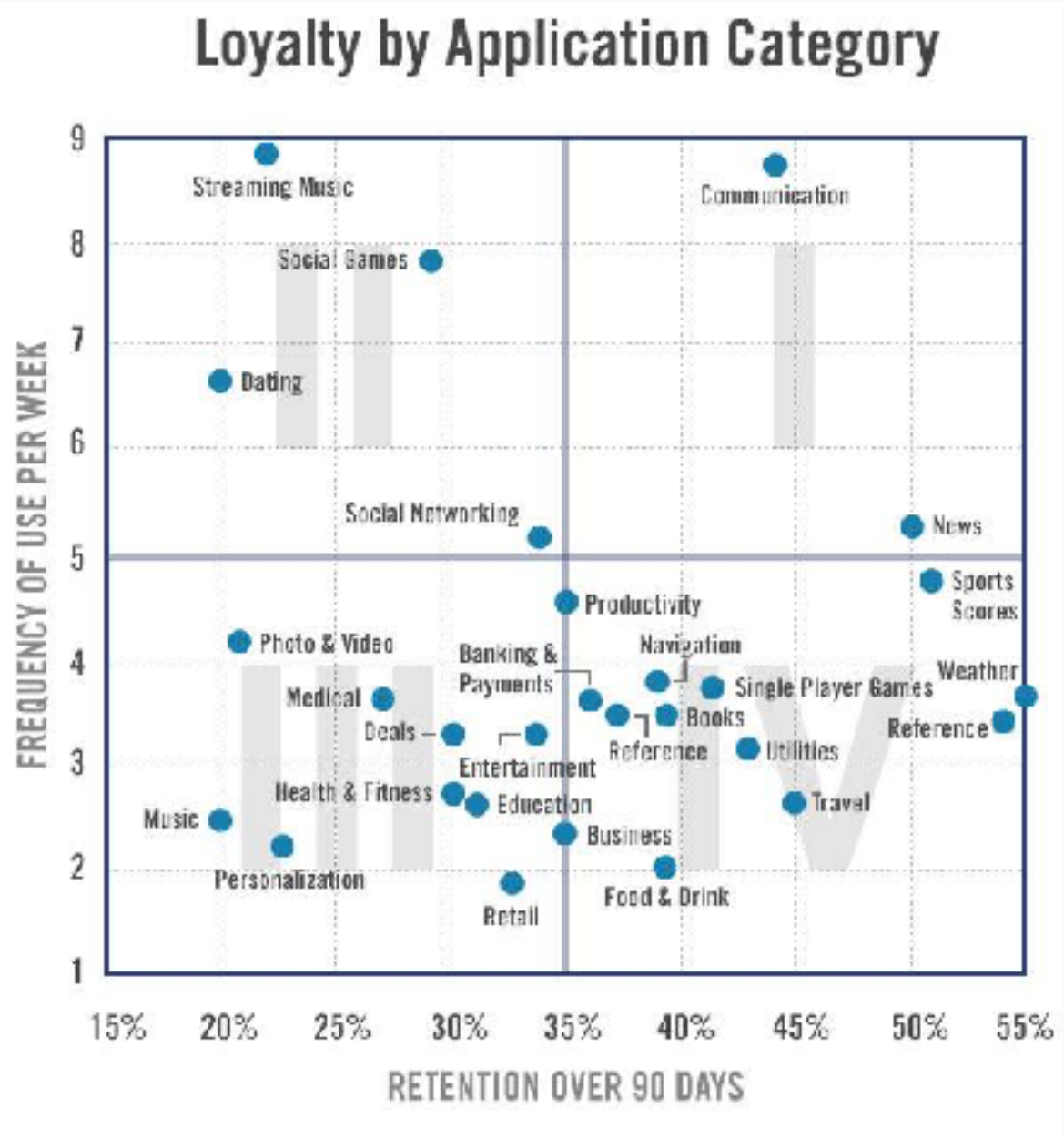
source: *User Testing Your Next Project*

Usability metrics (financial)

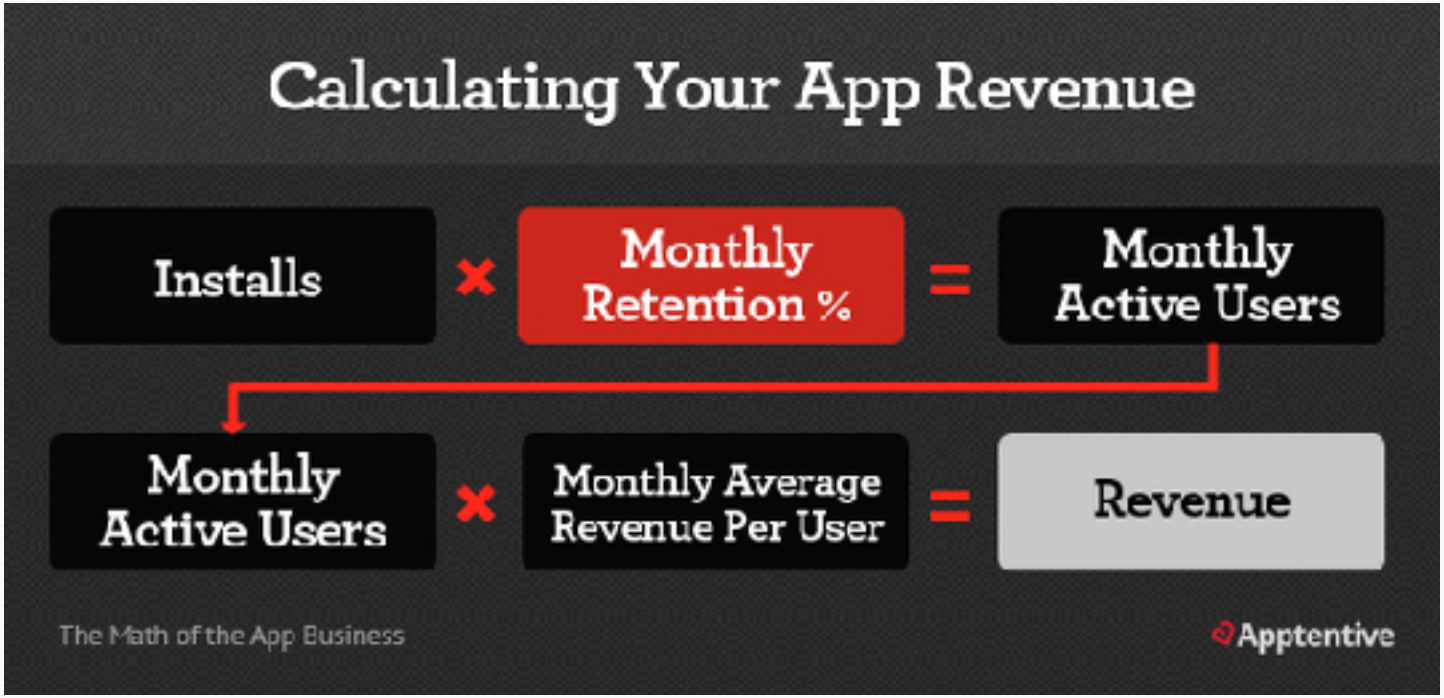
All about $

- Conversion (Conversion Rate)
- Average Revenue Per User
- Support Costs

Usability metrics examples



Usability metrics examples



Usability checklists

Usability checklists:

- https://stayintech.com/info/UX
- http://uxchecklist.github.io

Usability metrics (non-financial)

Table 6.2 Criteria by which measuring method can be determined (adapted from Whiteside, Bennett and Holtzblatt [377], Copyright 1988, reprinted with permission from Elsevier)

1.	Time to complete a task
2.	Per cent of task completed
3.	Per cent of task completed per unit time
4.	Ratio of successes to failures
5.	Time spent in errors
6.	Per cent or number of errors
7.	Per cent or number of competitors better than it
8.	Number of commands used
9.	Frequency of help and documentation use
10.	Per cent of favorable/unfavorable user comments
11.	Number of repetitions of failed commands
12.	Number of runs of successes and of failures
13.	Number of times interface misleads the user
14.	Number of good and bad features recalled by users
15.	Number of available commands not invoked
16.	Number of regressive behaviors
17.	Number of users preferring your system
18.	Number of times users need to work around a problem
19.	Number of times the user is disrupted from a work task
20.	Number of times user loses control of the system
21.	Number of times user expresses frustration or satisfaction

Usability metrics cutoff

Table 6.3 Possible ways to set measurement levels in a usability specification (adapted from Whiteside, Bennett and Holtzblatt [377], Copyright 1988, reprinted with permission from Elsevier)

Set levels with respect to information on:
1. an existing system or previous version
2. competitive systems
3. carrying out the task without use of a computer system
4. an absolute scale
5. your own prototype
6. user's own earlier performance
7. each component of a system separately
8. a successive split of the difference between best and worst values observed in user tests

Usability metrics

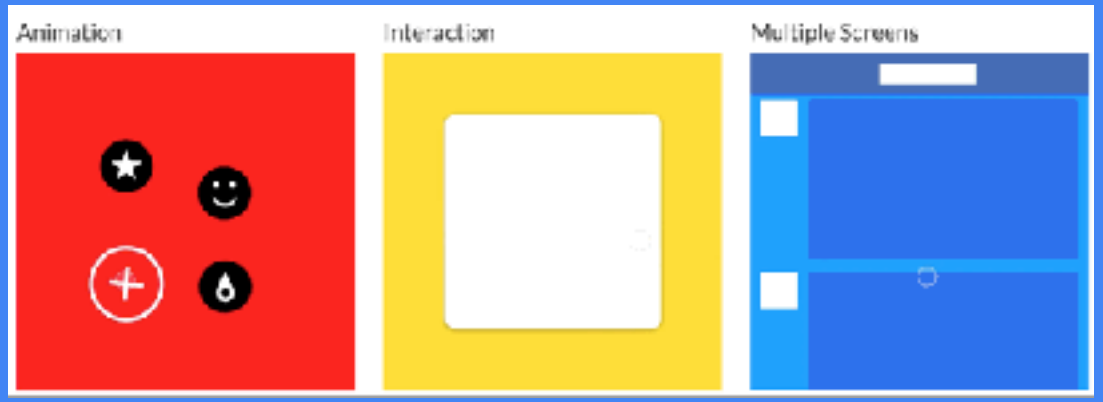
Table 6.4 Examples of usability metrics from ISO 9241

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Suitability for the task	Percentage of goals achieved	Time to complete a task	Rating scale for satisfaction
Appropriate for trained users	Number of power features used	Relative efficiency compared with an expert user	Rating scale for satisfaction with power features
Learnability	Percentage of functions learned	Time to learn criterion	Rating scale for ease of learning
Error tolerance	Percentage of errors corrected successfully	Time spent on correcting errors	Rating scale for error handling

Usability engineering

2. Software for making prototypes

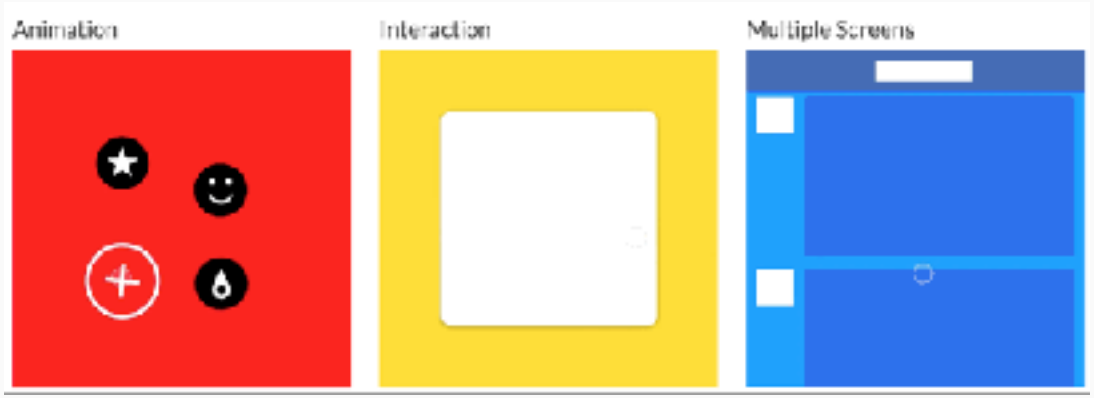
Prototypes by fidelity



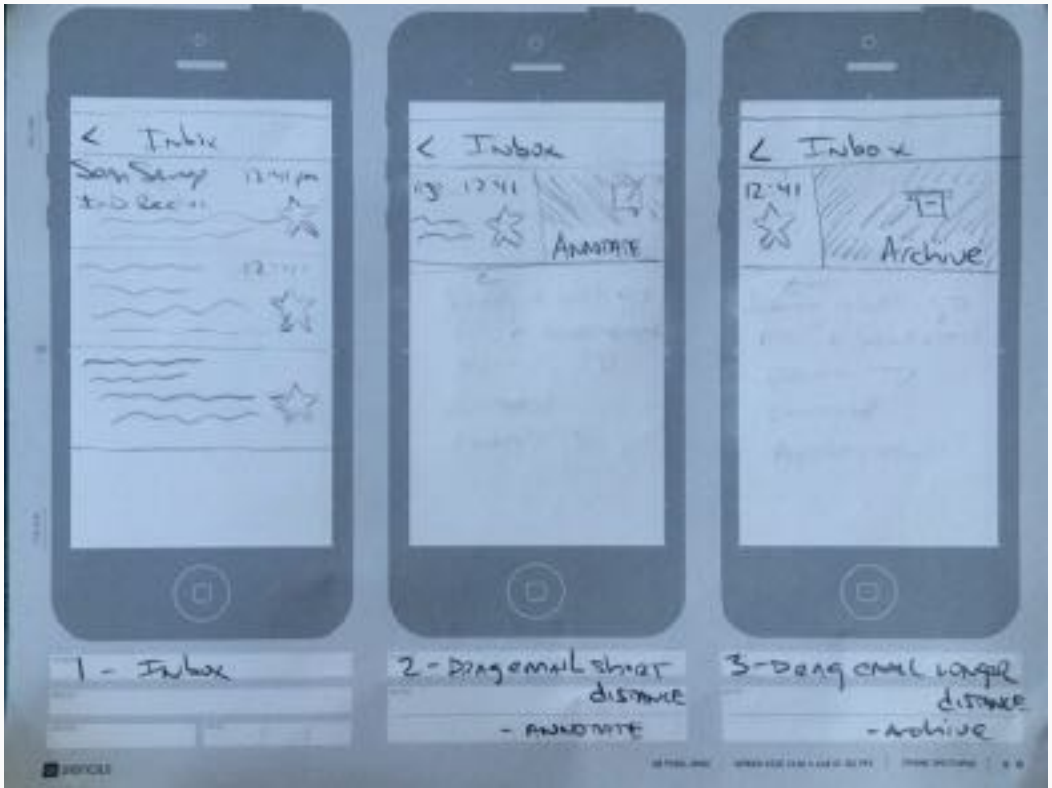
- Low-fidelity (also called Wireframes)
- Mid-fidelity (???)
- Hi-fidelity (detailed, also called prototypes):
  - Interactive prototypes
  - Non-interactive prototypes

Low-fidelity prototypes

- Cheap
- Fast

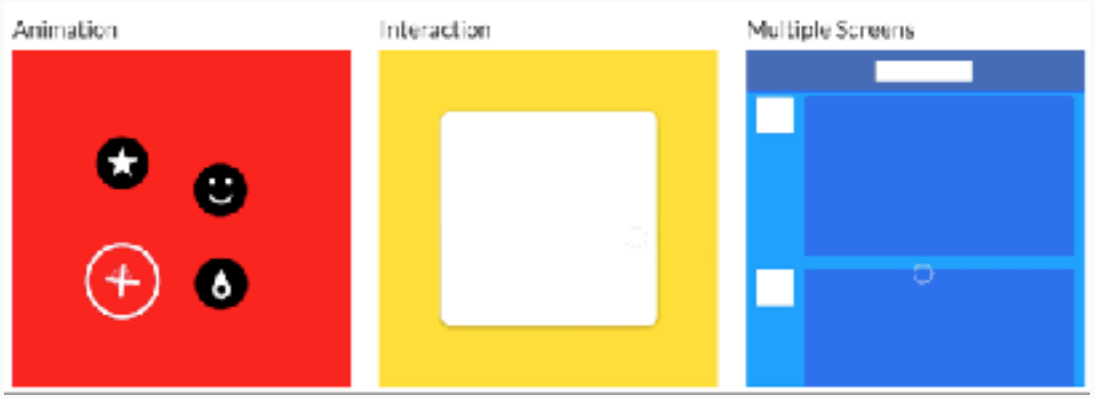


- Paper (good place to start)

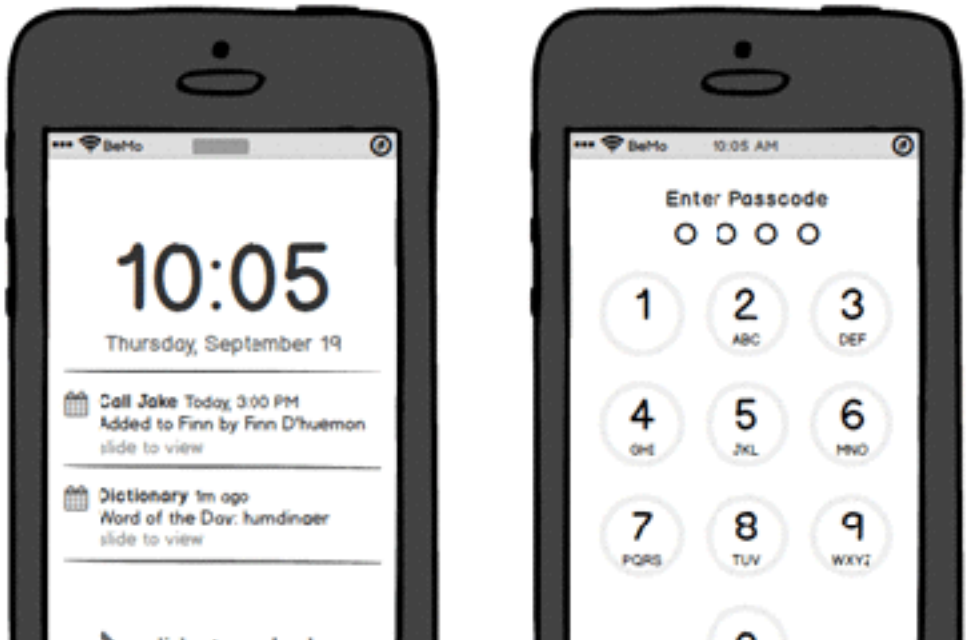


Low-fidelity prototypes

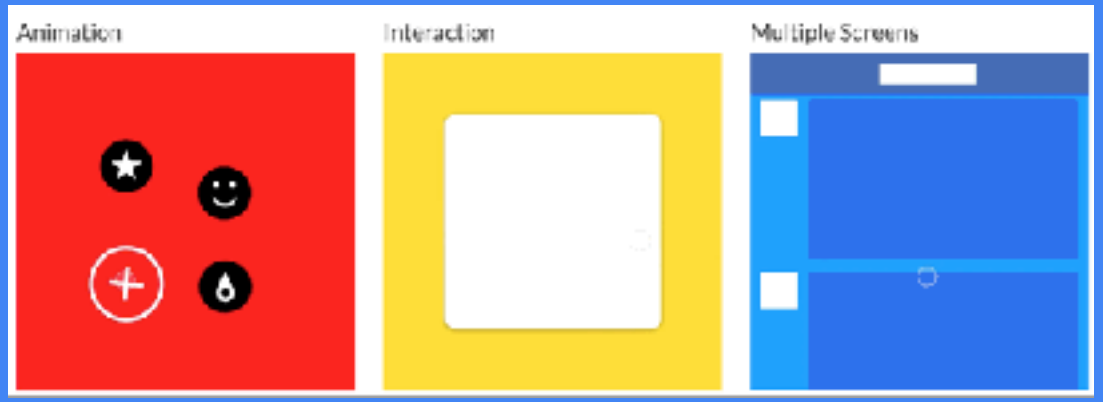
- Cheap
- Fast



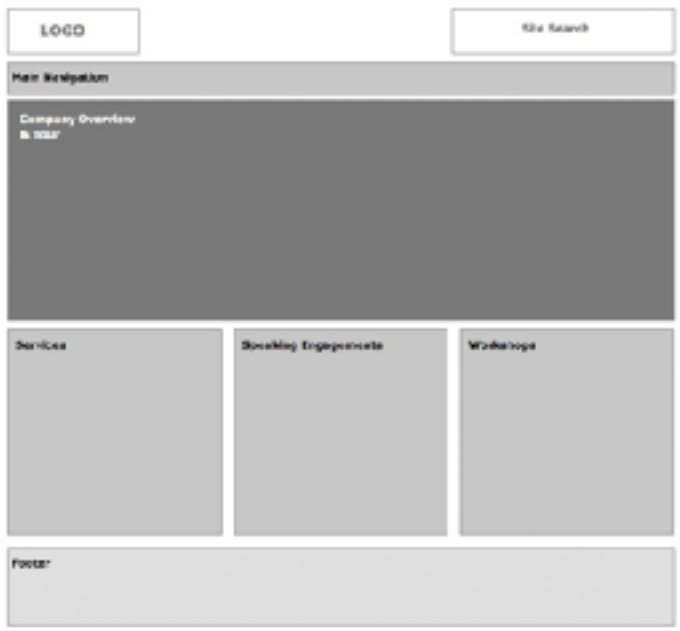
- Balsamiq (add functional details, layouts)

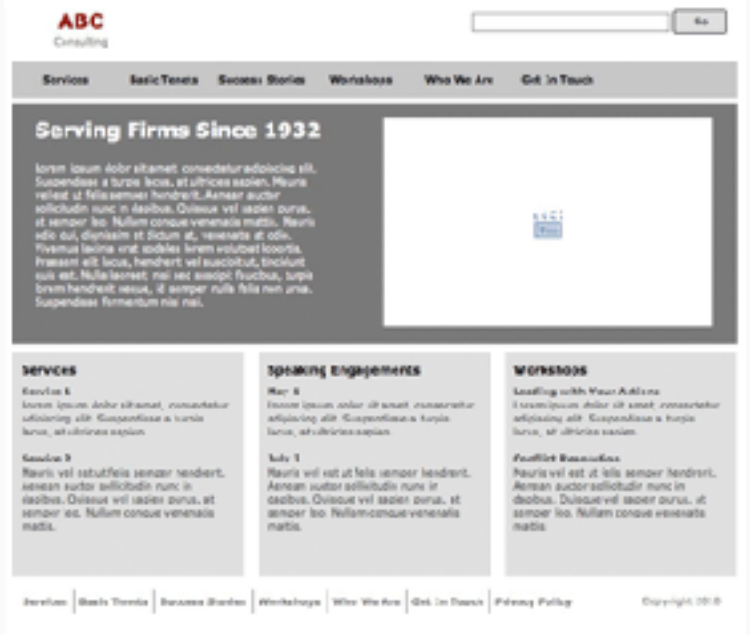


Prototypes fidelity



- Low-fidelity (also called Wireframes)

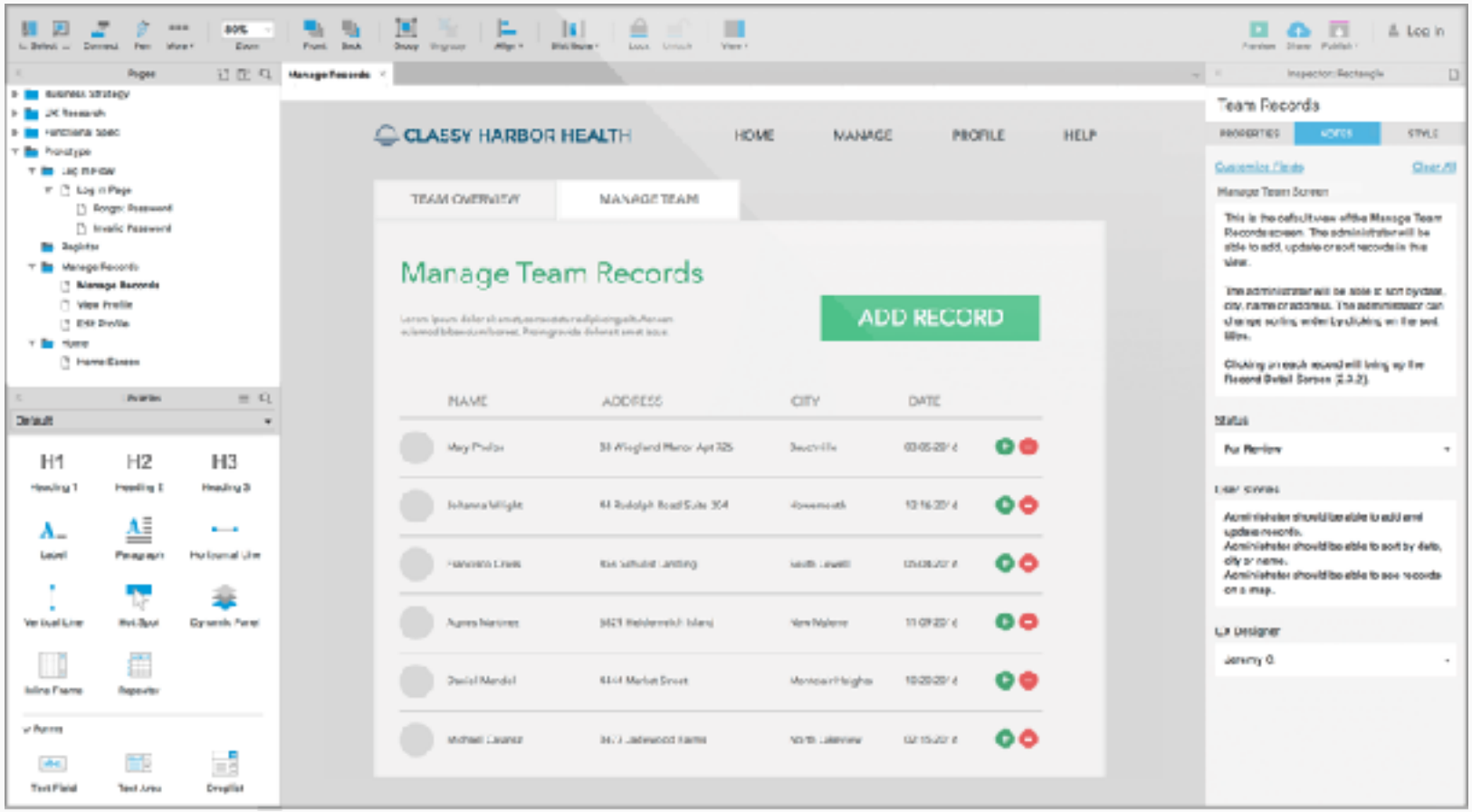




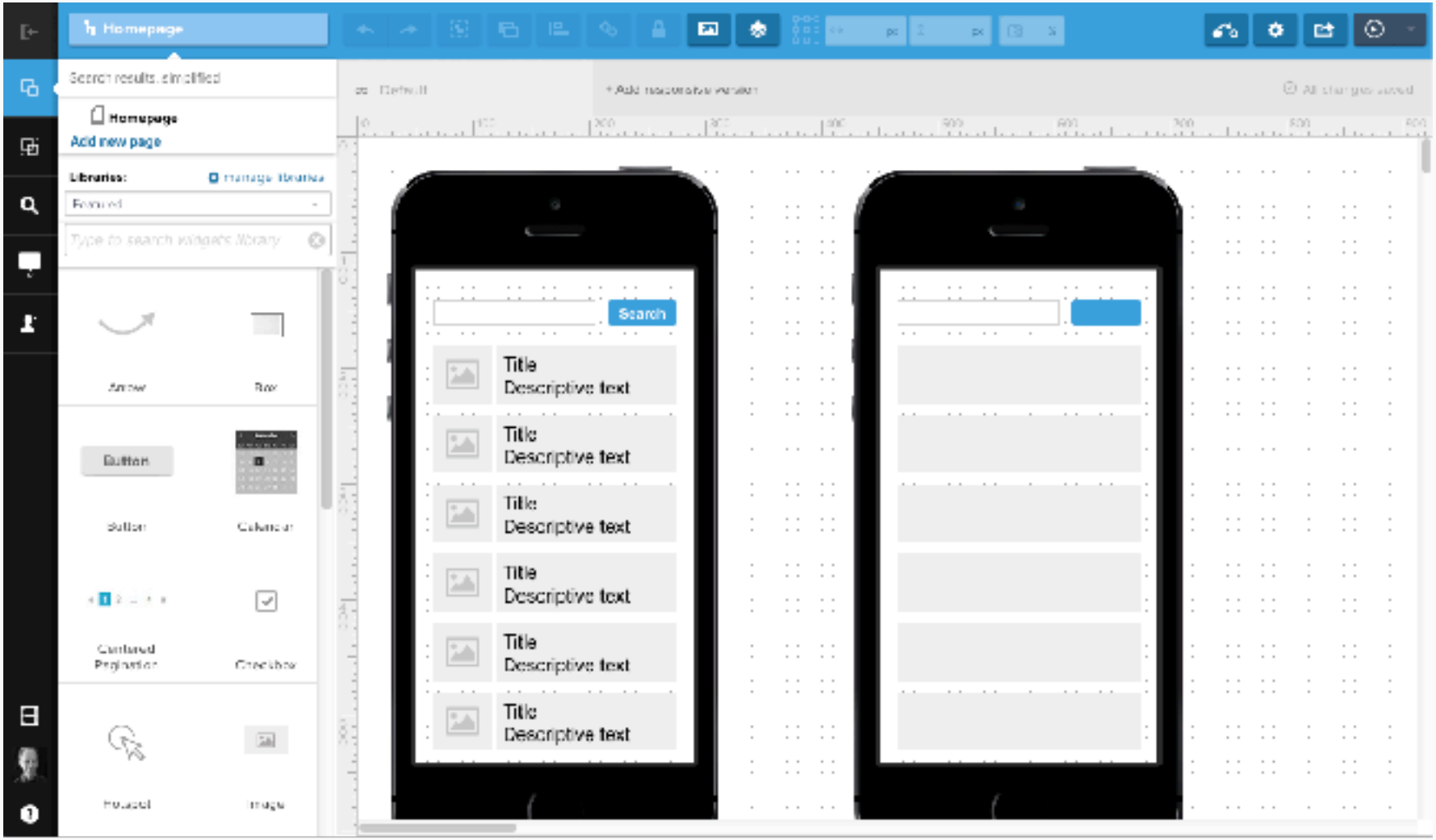
Hi-fidelity prototyping software

Axure RP

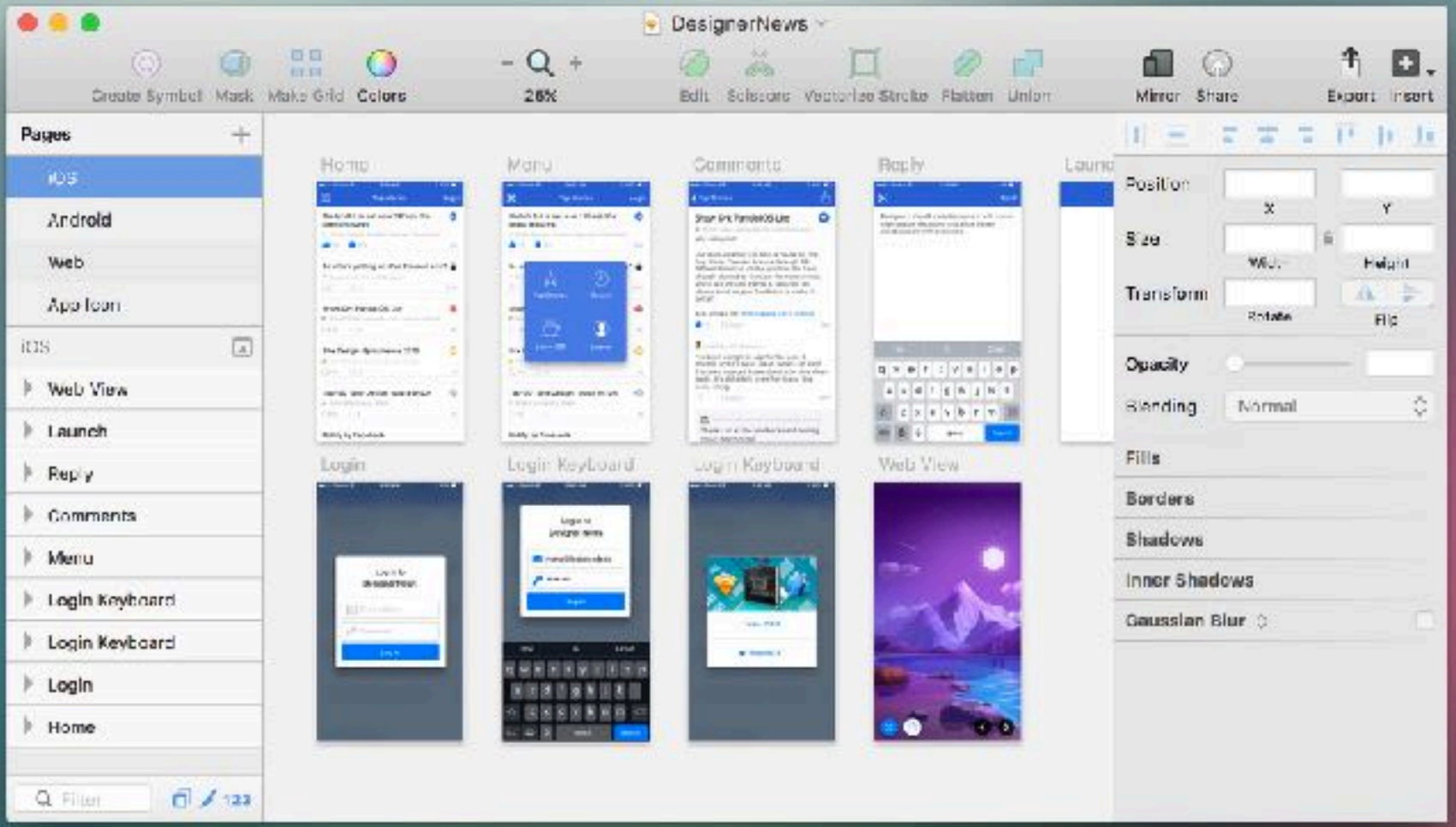
- http://wearebridge.co/ux-tools/Axure-Flat-UI-Kit-Preview/



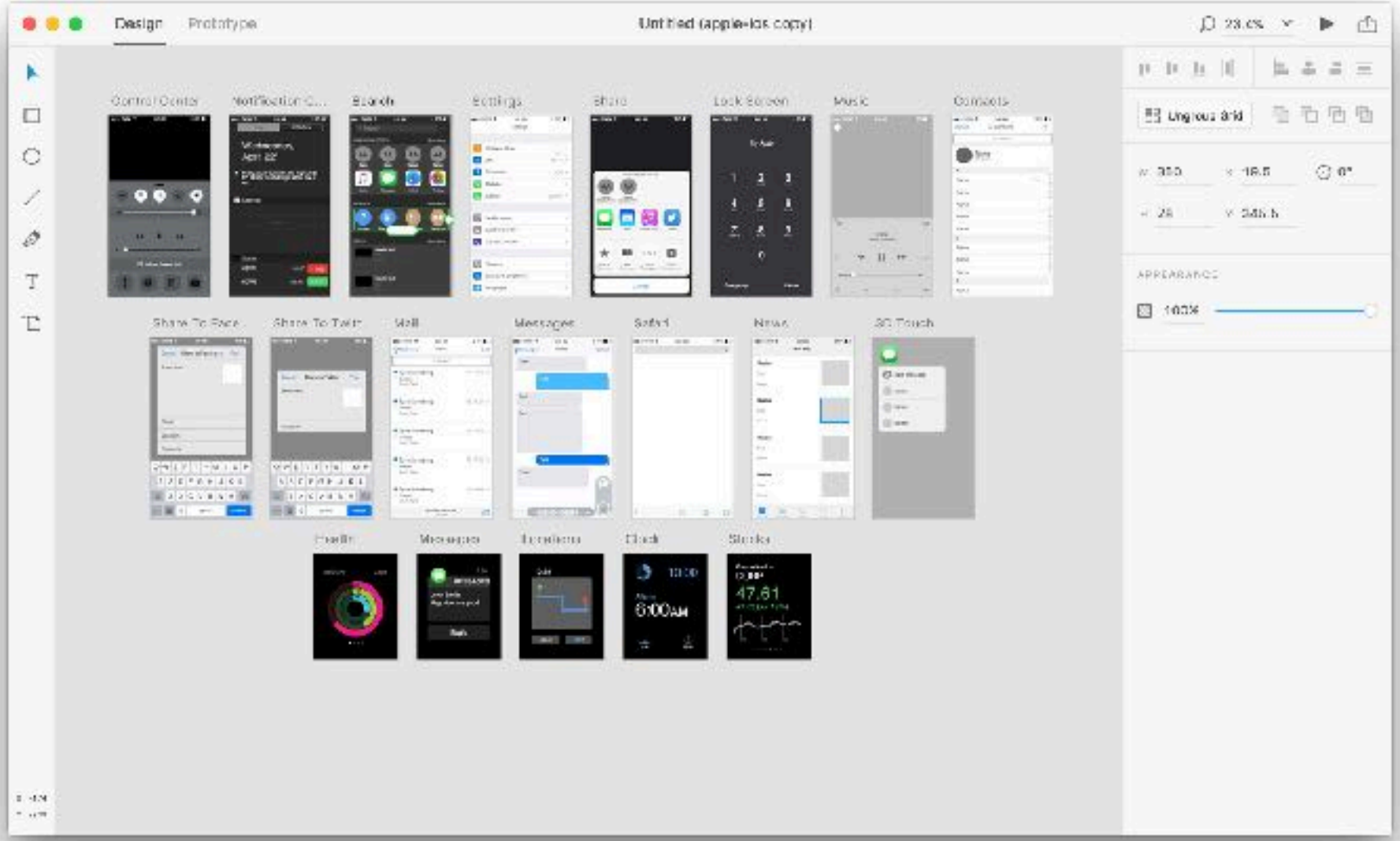
UXPin.com



Sketch (Sketch.app)



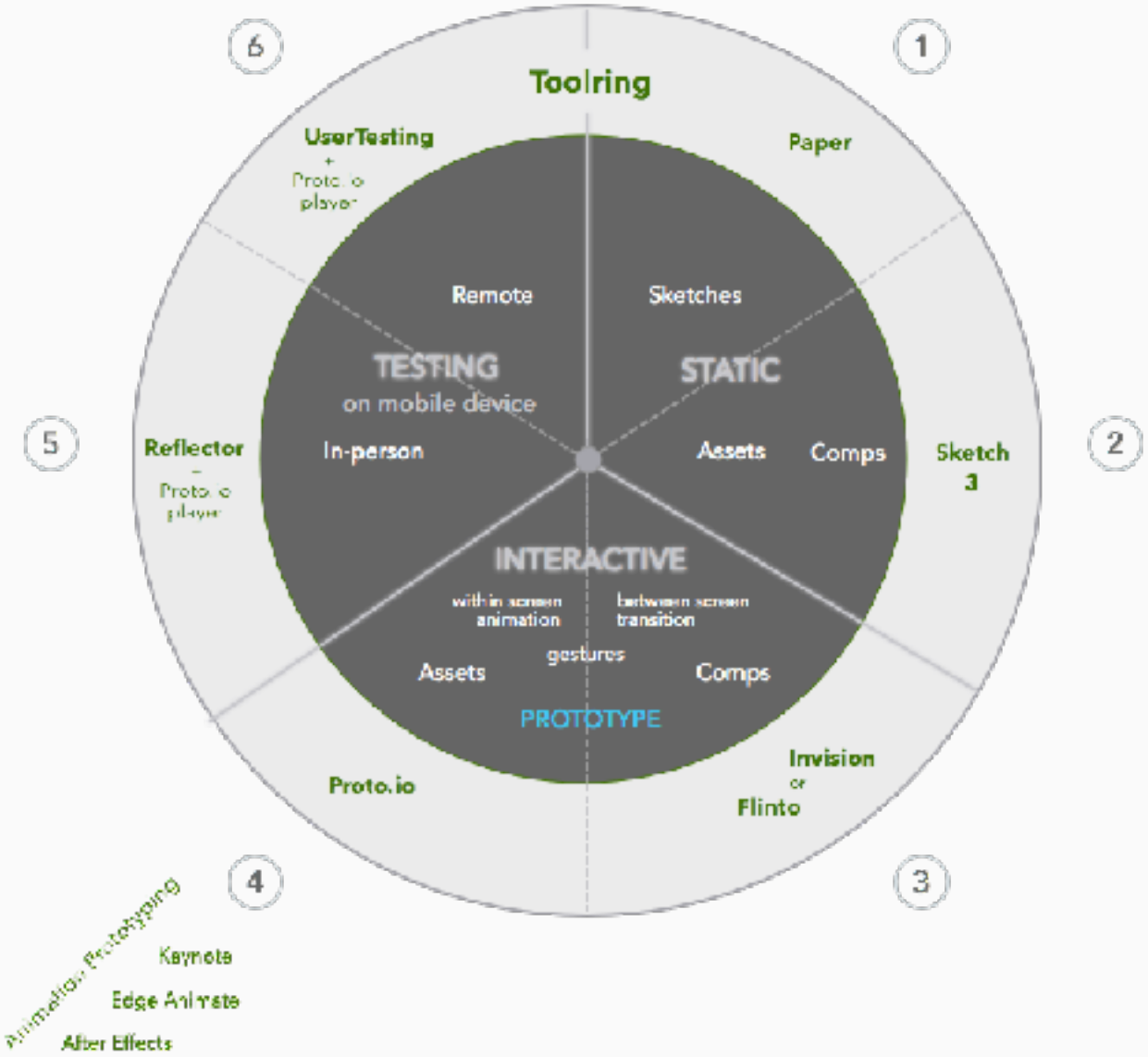
Adobe XD (Photoshop -> Fireworks -> XD)



Prototyping tools (hi-fidelity)

	Platform	Trial	Popularity	Could be interactive
Sketch	MacOS only!	+	5	— (except plugin with external serices)
UXPin	Web	+	4	?
Adobe XD	MacOS + Windows	+	3	—
Axure	MacOS + Windows	+	4	+ (need JavaScript)

Prototyping tools (hi-fidelity)



Prototyping tools (hi-fidelity)

Prototyping Tools		Mockplus	Axure	Balsamiq	JustInmind	Sketch	Adobe XD (Preview	Invision
Productivity	Learning Curve	Very Easy	Complex	Very Easy	Complex	Average	Average	Easy
	Integrated Efficiency	Fast	Average	Fast	Slow	Average	Average	Fast
	Interaction Design	Fast	Average	-	Average	Plug-in Required	Fast	-
	Build Widgets	Fast	Slow	Fast	Average	Slow	Slow	-
	Device Testing	Fast	Slow	-	Average	Plug-in Required	Average	Fast
Fidelity	Visual Fidelity	Average	Average	Low	High	High	High	High
	Interactive Fidelity	Average	High	-	High	High	High	Average
Professional Skill Requirement	Product Experience	Required	Required	Required	Required	-	-	Required
	Visual Design	-	-	-	Required	Required	Required	Required
	Programming Knowledge	-	Basic Knowledge	-	-	Basic Knowledge	-	-
Sharing		Average	Great	Average	Great	-	-	Great

Prototyping tools (hi-fidelity + interactive)

- Invision App
- Flinto
- proto.io

proto.io features

- gestures (tap, swipe, pinch, drag, scroll, etc)
- animations (fade, scale, move, resize, rotate, ease, delay)
- transitions (slide, fade, flip, turn, pop, flow)
- states (for screens and containers, using a timeline)
- variables (for conditional animation)

Lab 2 (usability check)

Task

1. Select any web site

2. Check how much items from this usability list (http://uxchecklist.github.io) are present and absent

3. Prepare short presentation about pitfalls and ways of improvement usability

Lab 3 (introduction to R)

Task

1. Log in to http://rstudio.projectrelay.ru:8787/
2. User: «user22», password: «hmiclass»

Nuclear plant exercise

- http://www.hcibook.com/e3/scenario/nuclear/

Examples of user onboard

- http://www.useronboard.com/onboarding-teardowns/