

MAKING YOUR GATEWAY EASY AND PLEASANT TO USE

AN INTRODUCTION TO USABILITY AND USER-CENTERED DESIGN

Paul Parsons

November 8, 2017
SGCI Webinar



Paul Parsons

Assistant Professor

Purdue University, USA

web.ics.purdue.edu/~parsonsp

parsonsp@purdue.edu



Background

PhD, Computer Science

Specialty: Human-Computer Interaction (HCI)

BS, Computer Science & Cognitive Science

Teaching

Previous (2011-2014)

HCI for computer science students

Current

User Experience (UX) for undergrad and grad

OBJECTIVES

1. **value** the importance of usability and user-centered design
2. be able to **identify** some common usability problems
3. **appreciate** various strategies for identifying and fixing common usability problems
4. be able to **communicate** with usability experts in an informed way

HISTORY

Interface design was **system-centric**

Users had to adapt to the technology

Focus was on *training users*

Design was concerned with:

What can we build with such-and-such a platform?

How efficient is the code? How can resources be optimized?

Most users don't care about these *unless* they impact usability and the user experience



SYSTEM-CENTERED DESIGN



View of the user?

A person who should **adapt to your system**

SYSTEM-CENTERED DESIGN

View of the user?

The user is **just like me!**



<https://xkcd.com/619/>

FOR USABLE SOFTWARE...

Need to shift the focus from **system-centered** design to **user-centered** design

USER-CENTERED DESIGN

First Rule

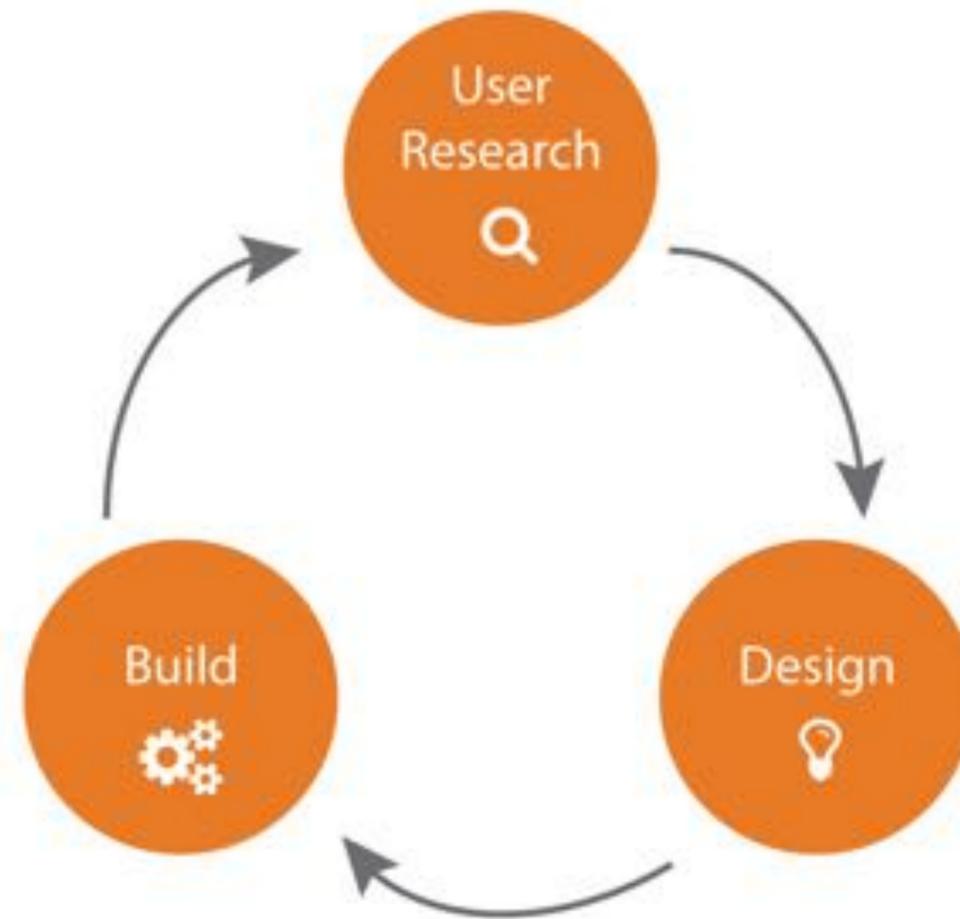
Know Thy Users!

And you are not thy users...

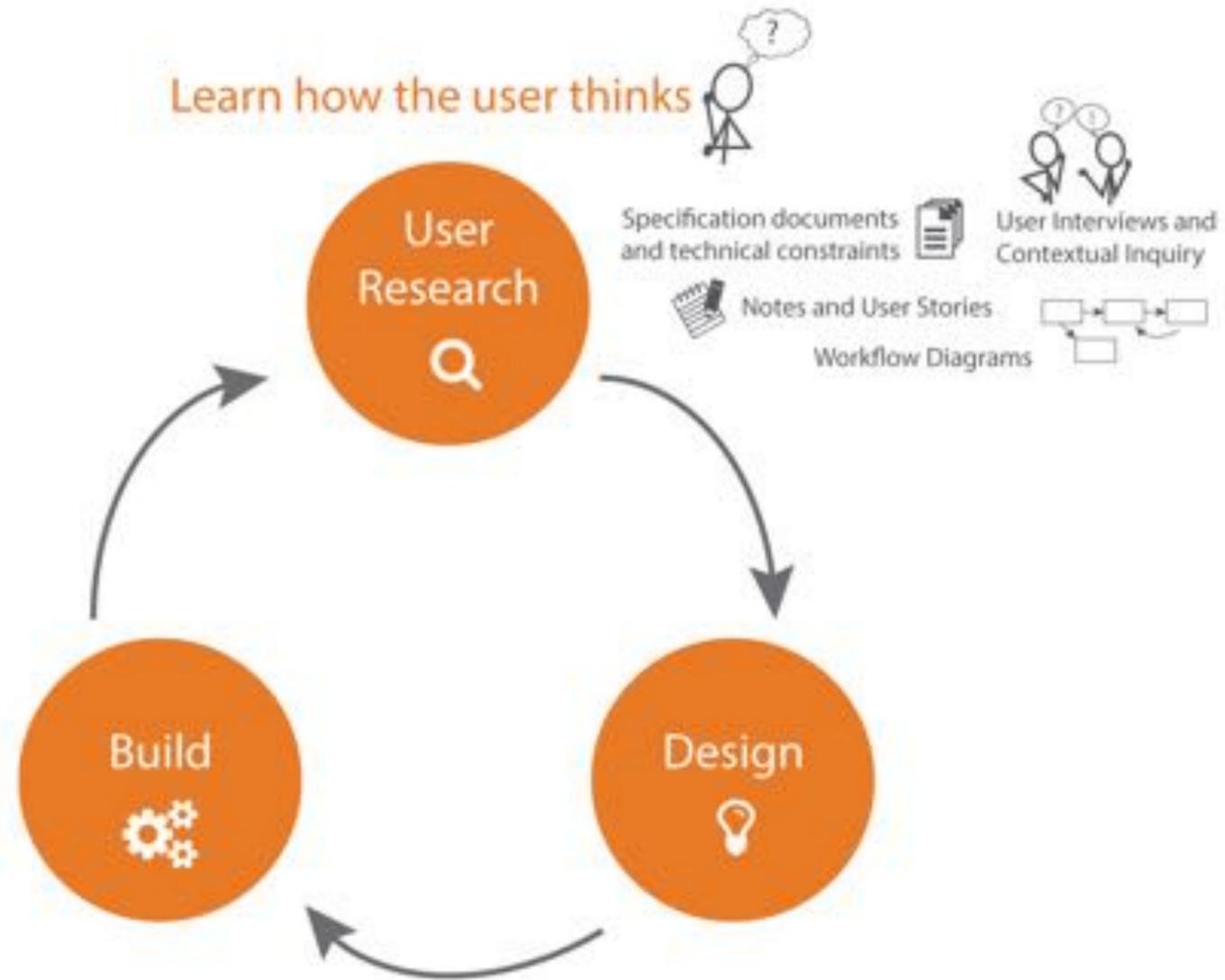
Corollary: if you think you know thy users, think again!



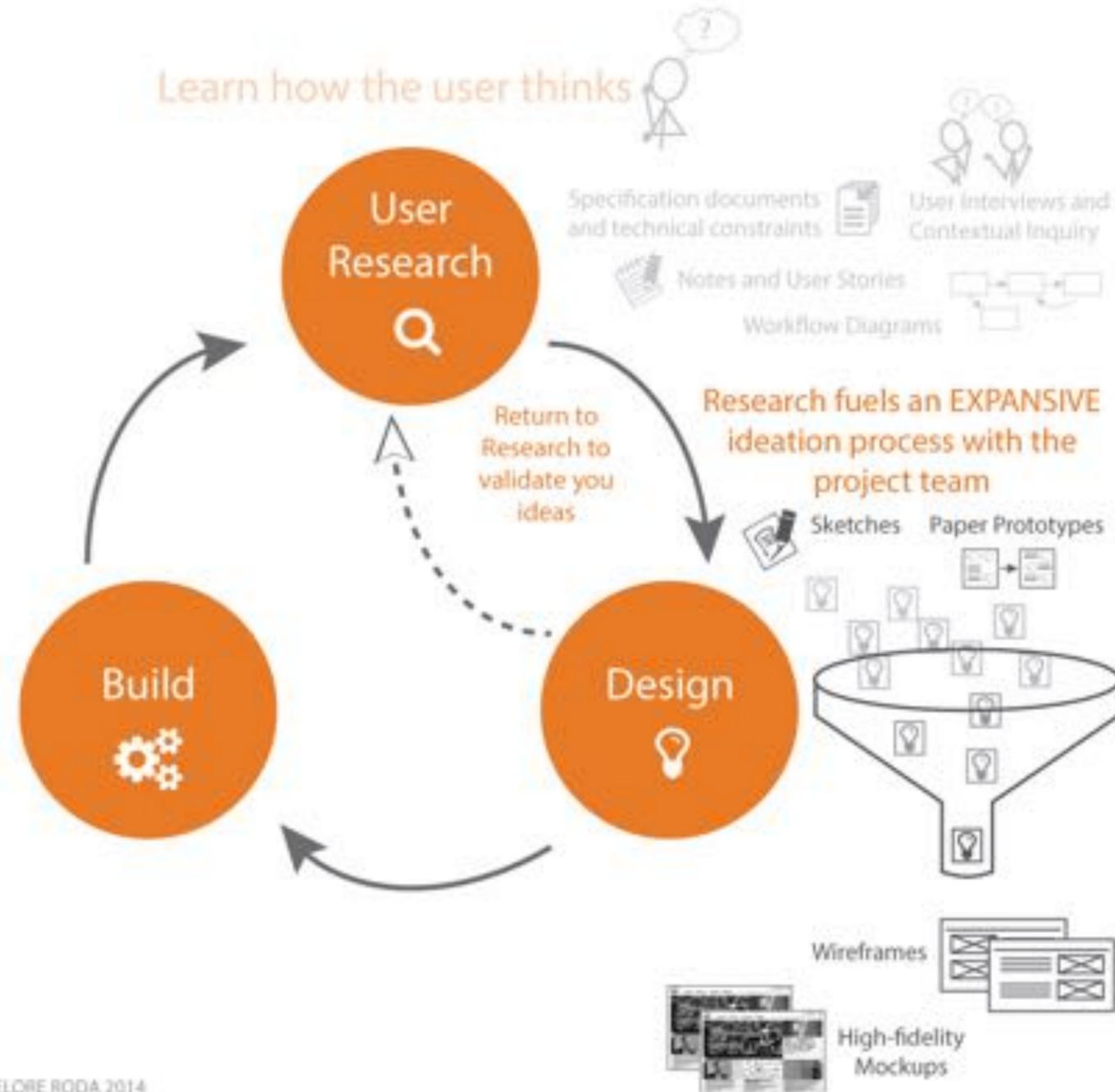
USER-CENTERED DESIGN



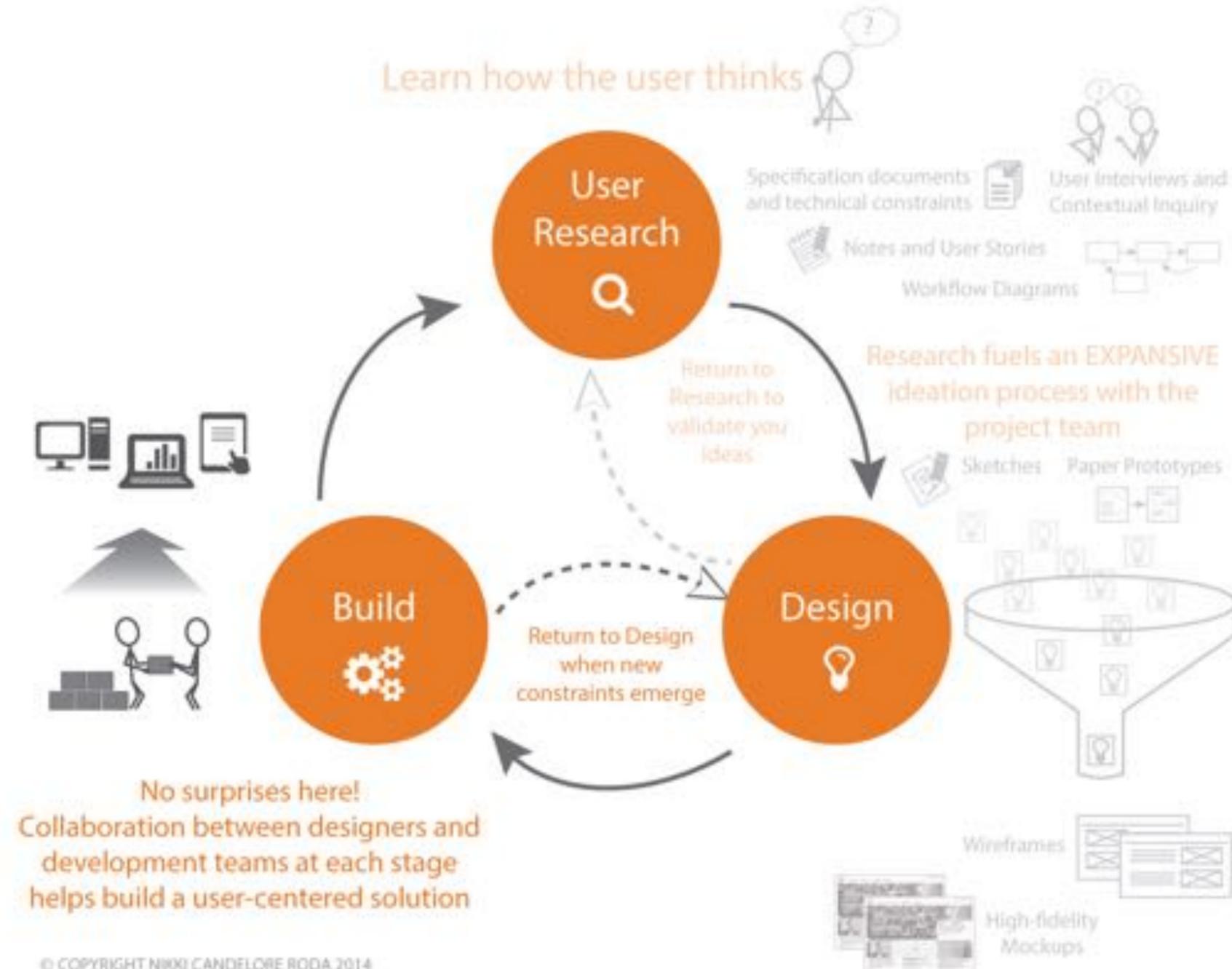
USER-CENTERED DESIGN



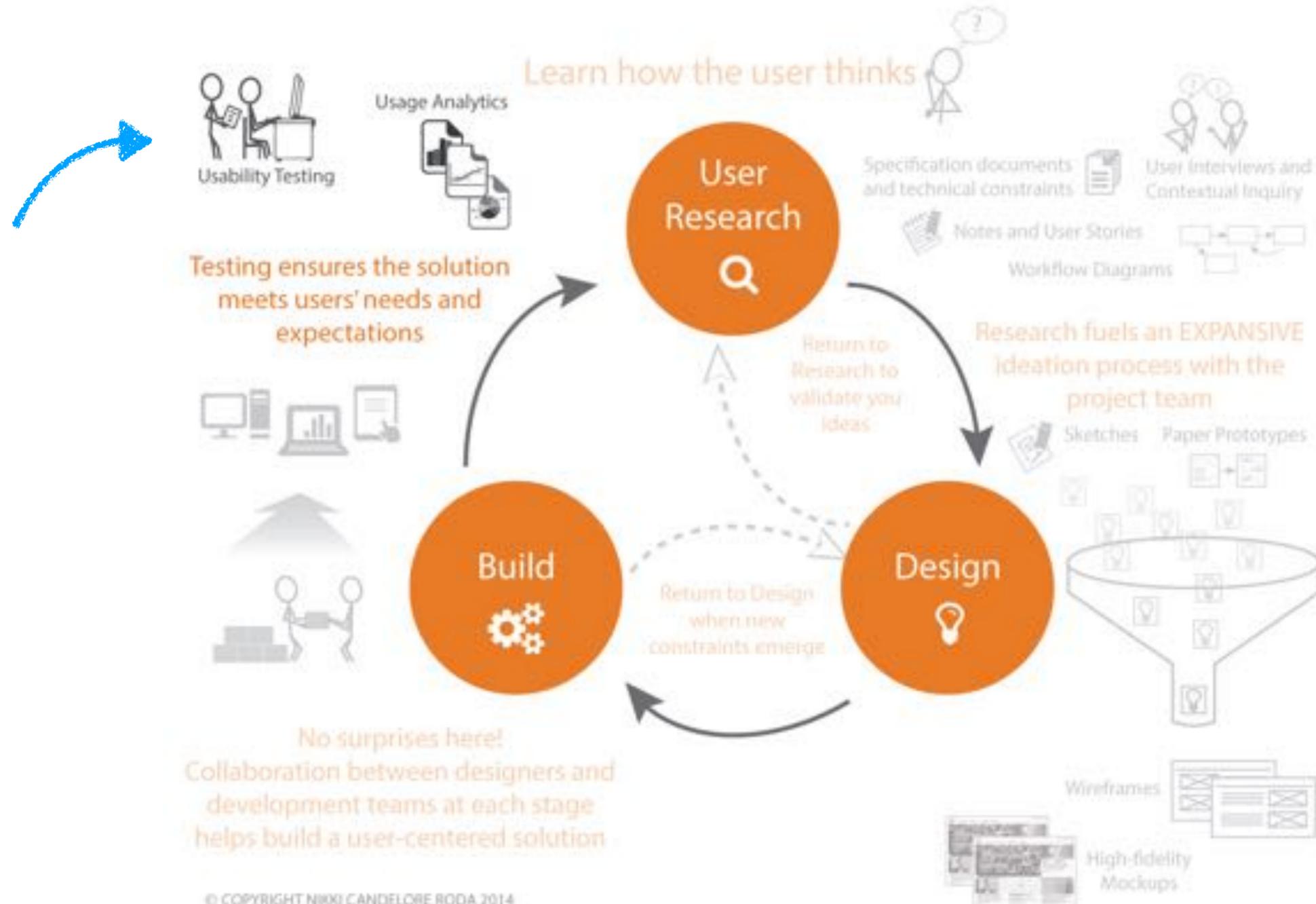
USER-CENTERED DESIGN



USER-CENTERED DESIGN



USER-CENTERED DESIGN



USABILITY

Learnability: How easy is it for users to accomplish basic tasks the first time they encounter the design?

Efficiency: Once users have learned the design, how quickly can they perform tasks?

Memorability: When users return to the design after a period of not using it, how easily can they re-establish proficiency?

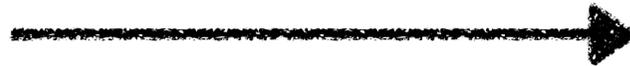
Errors: How many errors do users make, how severe are these errors, and how easily can they recover from the errors?

Satisfaction: How pleasant is it to use the design?

USABILITY

Some reasons for poor usability

not understanding users



solving the wrong problem

violating visual design principles

violating interaction design principles

PRINCIPLES

80/20 rule
affordances
alignment
chunking
closure
common fate
consistency
constraint

control
convergence
errors
feedback
figure-ground
Fitts' law
Hick's law
interference

...

layering
mapping
navigation
progressive disclosure
proximity
recognition over recall
similarity
visual hierarchy

VISUAL HIERARCHY

arrangement and styling of elements in a way that implies importance

visual hierarchy influences the order in which the human eye perceives what it sees

VISUAL HIERARCHY



The image shows a screenshot of a restaurant website. At the top, there is a navigation bar with several menu items: Home, Our Menu, Specials, Restaurant Locator, Employment, The Company, and Contact Us. Each item is accompanied by a small icon or image. Below the navigation bar, the main content area is divided into two columns. The left column features a section titled "A Reminder!" with a decorative border. The right column features a section titled "Fundraiser" with a decorative border. The "A Reminder!" section contains text about ordering items and catering. The "Fundraiser" section contains text about a neighborhood restaurant and a photo of food.

Home **Our Menu** **Specials** **Restaurant Locator** **Employment** **The Company** **Contact Us**

A Reminder!

Just a reminder that you can order all items on our menu to go. We would love to cater your next party or have you hold it here. Ask the Manager on duty about the details.

Imagine that You are sitting in a cozy neighborhood restaurant in Naples, drinking wine and enjoying delicious Italian food. Picture big portions of homemade ravioli, sumptuous lasagna, chicken parmigiana and heaping plates of spaghetti and hand-rolled meatballs. Feast at our Salad Bar with 26 delicious items to choose from. Sample the Minestrone Soup, five different kind of Pizza, baked chicked, sausage, and pasta. Can you see it? That's what it's like to visit Joe's Place. Except, well . . . Joe's is just down the street from your house. We have two locations in Washington Metro area. Click on the "Restaurant Locator" button to see which is nearest to your house...

Fundraiser



VISUAL HIERARCHY

Pizza

5555 LEE HIGHWAY
ARLINGTON, VA
703-532-0990

[MENU](#)

Pasta

Joe's
Place
PIZZA & PASTA

Sandwiches

OPEN EVERYDAY
FROM
10:30AM UNTIL 10:00PM

[ORDER ONLINE](#)

[ORDER ONLINE](#)

VISUAL HIERARCHY



Evented I/O for [V8 JavaScript](#).

An example of a web server written in Node which responds with "Hello World" for every request.

```
var http = require('http');
http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World\n');
}).listen(1337, '127.0.0.1');
console.log('Server running at http://127.0.0.1:1337/');
```

To run the server, put the code into a file `example.js` and execute it with the `node` program:

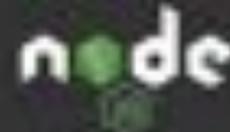
```
$ node example.js
Server running at http://127.0.0.1:1337/
```

Here is an example of a simple TCP server which listens on port 1337 and echoes whatever you send it:

```
var net = require('net');

var server = net.createServer(function (socket) {
  socket.write("Echo server\r\n");
  socket.pipe(socket);
});
```

VISUAL HIERARCHY



[HOME](#) | [ABOUT](#) | [DOWNLOADS](#) | [DOCS](#) | [FOUNDATION](#) | [GET INVOLVED](#) | [SECURITY](#) | [NEWS](#)

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, npm, is the largest ecosystem of open source libraries in the world.

Important security releases, please update now!

Download for macOS (x64)

8.9.1 LTS

Recommended For Most Users

[Other Downloads](#) | [Changelog](#) | [API Docs](#)

9.1.0 Current

Latest Features

[Or have a look at the LTS schedule.](#)

Sign up for Node.js Everywhere, the official Node.js Weekly Newsletter.

VISUAL HIERARCHY



GESTALT PERCEPTION

we tend to **order our experience** in a manner that is regular, orderly, symmetrical, and simple

proximity

similarity

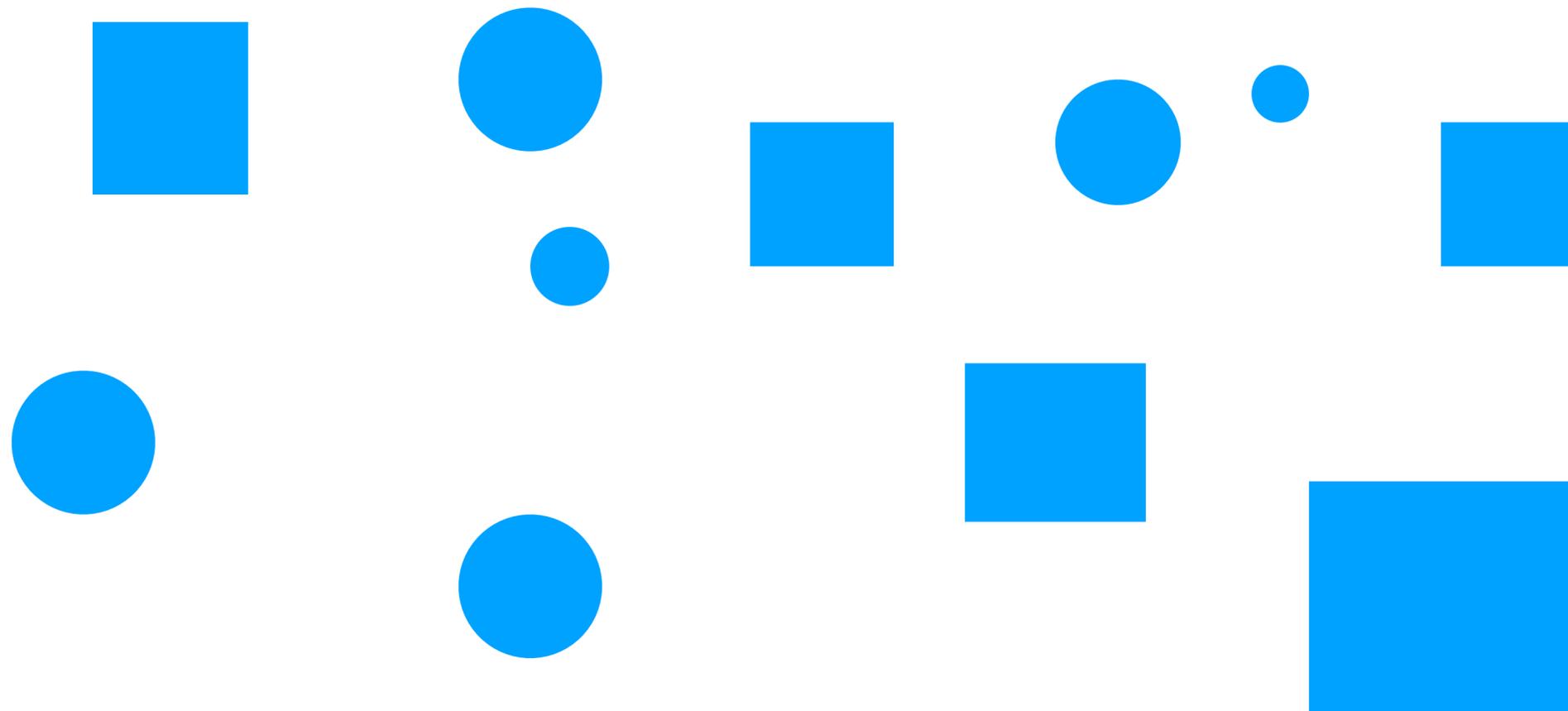
closure

symmetry

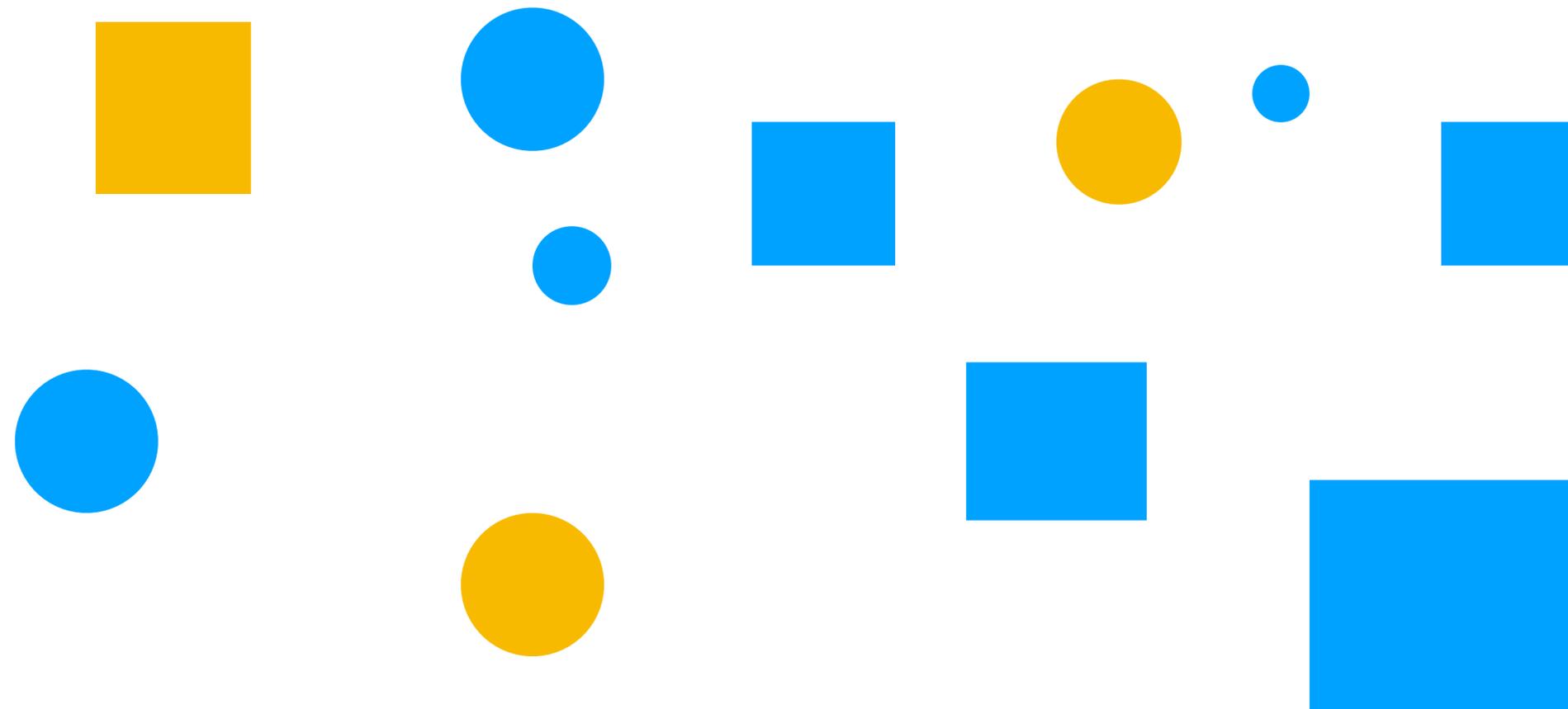
continuity

...

SIMILARITY



SIMILARITY



SIMILARITY

The screenshot shows the Bootstrap 4.0 documentation page. The main content area is titled "Quick start" and contains instructions for adding Bootstrap to a project. It includes two code snippets: one for the CSS link and one for the JavaScript scripts. The right sidebar contains a table of contents with arrows pointing to the "Quick start" and "JS" sections.

8 Home Documentation Examples Themes Jobs Blog

v4.0

Download

Search...

Getting started

- Introduction
- Download
- Content
- Browsers & devices
- JavaScript
- Theming
- Sub-views
- Icons
- Accessibility

Layout

- Content
- Components
- Utilities
- Extend
- Migration
- About

Quick start

Looking to quickly add Bootstrap to your project? Use the Bootstrap CDN, provided for free by the folks at MaxCDN. Using a package manager or need to download the source files? [Head to the downloads page.](#)

CSS

Copy-paste the stylesheet `<link>` into your `<head>` before all other stylesheets to load our CSS.

```
<link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/css/bootstrap.min.css" integrity="sha384-VeuzlhkhAAw1BNS086q2b303j8AJmMU1sk7F0yp5u7A5d06nmjjt7A9Kp0Pd6hh" crossorigin="anonymous">
```

JS

Many of our components require the use of JavaScript to function. Specifically, they require [jQuery](#), [Popper.js](#), and our own JavaScript plugins. Place the following `<script>`'s near the end of your pages, right before the closing `</body>` tag, to enable them. (jQuery must come first, then Popper.js, and then our JavaScript plugins.)

We use [jQuery's slim build](#), but the full version is also supported.

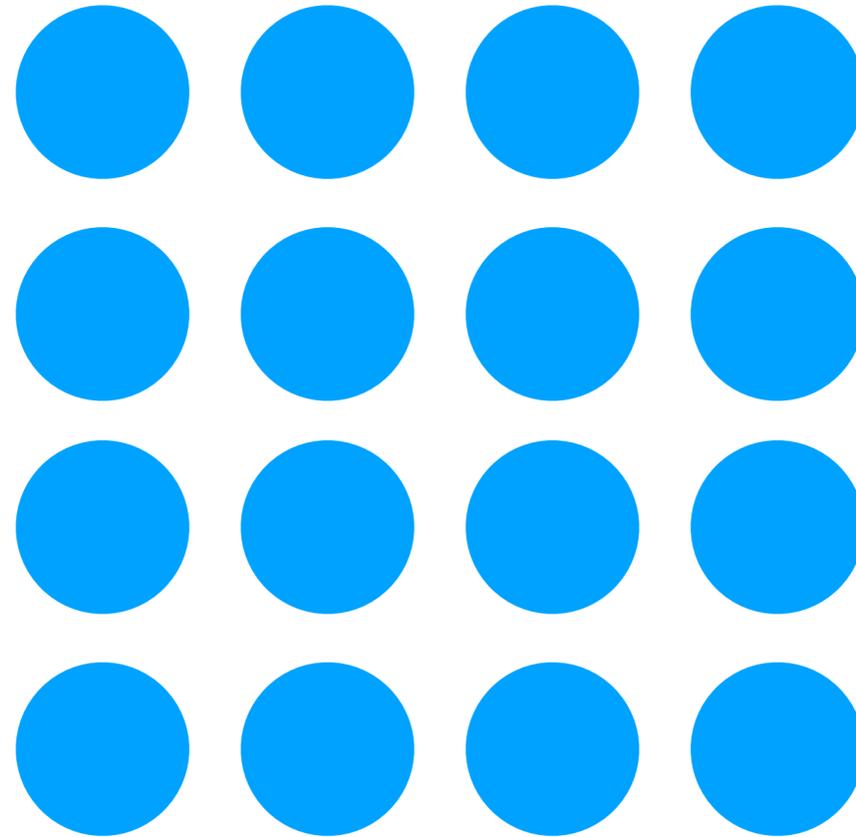
```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-k0W4z9zKPvQzIh9QjAo6TLxwD3KlHtA1z0K2PjyYLh2M9W3gZBeW2lQeR0qL" crossorigin="anonymous"></script>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.0/umd/popper.min.js" integrity="sha384-sfp实行菜等jr74w607cxf3bfH6sT64Uw50ud9tJ2HuB56j1qzn7St669qeB86C" crossorigin="anonymous"></script>  
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/js/bootstrap.min.js" integrity="sha384-41gtrLjBpS7R3b5PUHX241Ej084wLgVJOz8vV7L+VJoVid4pdgnQbZ1x5yo604p" crossorigin="anonymous"></script>
```

Curious which components explicitly require jQuery, our JS, and Popper.js? [Click the show components link below.](#) If you're at all unsure about the general page structure, keep reading for an [example page template](#).

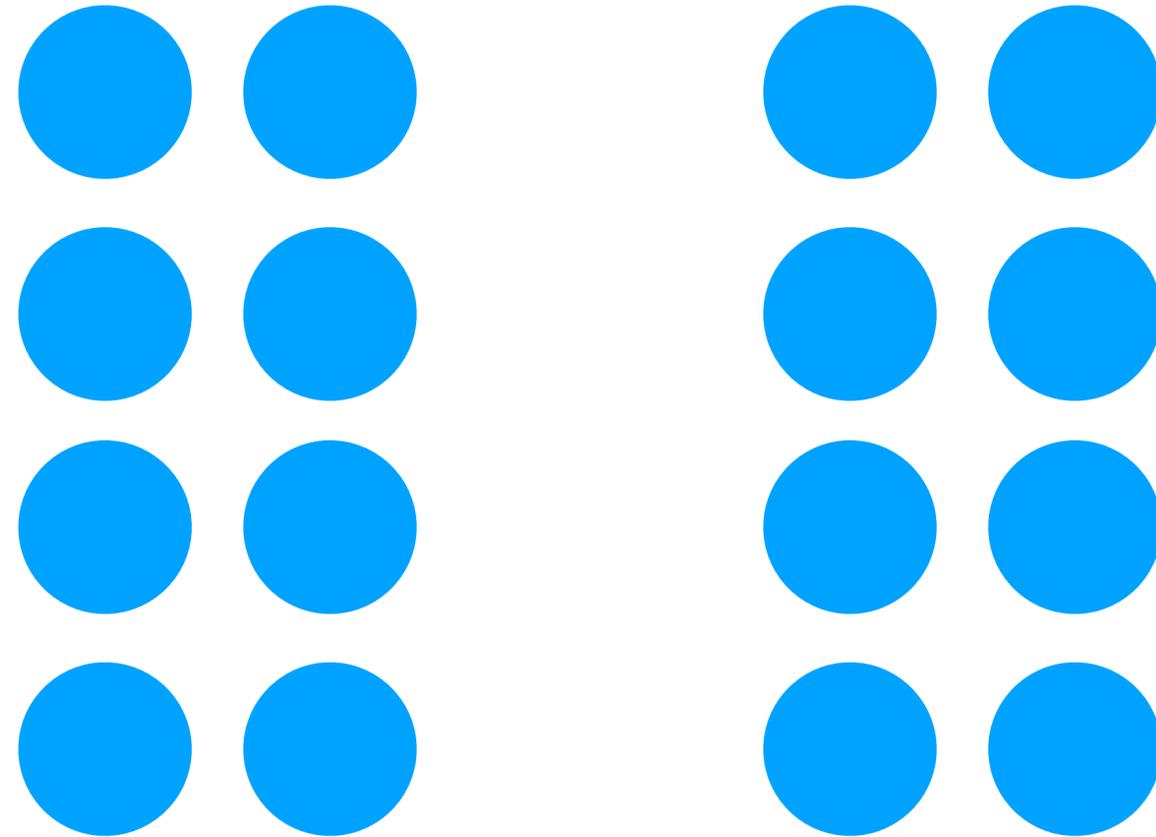
[Show components requiring JavaScript](#)

Quick start
CSS
JS
Getting started
Theming & icons
Sub-views
Icons
Accessibility
Layout
Content
Components
Utilities
Extend
Migration
About

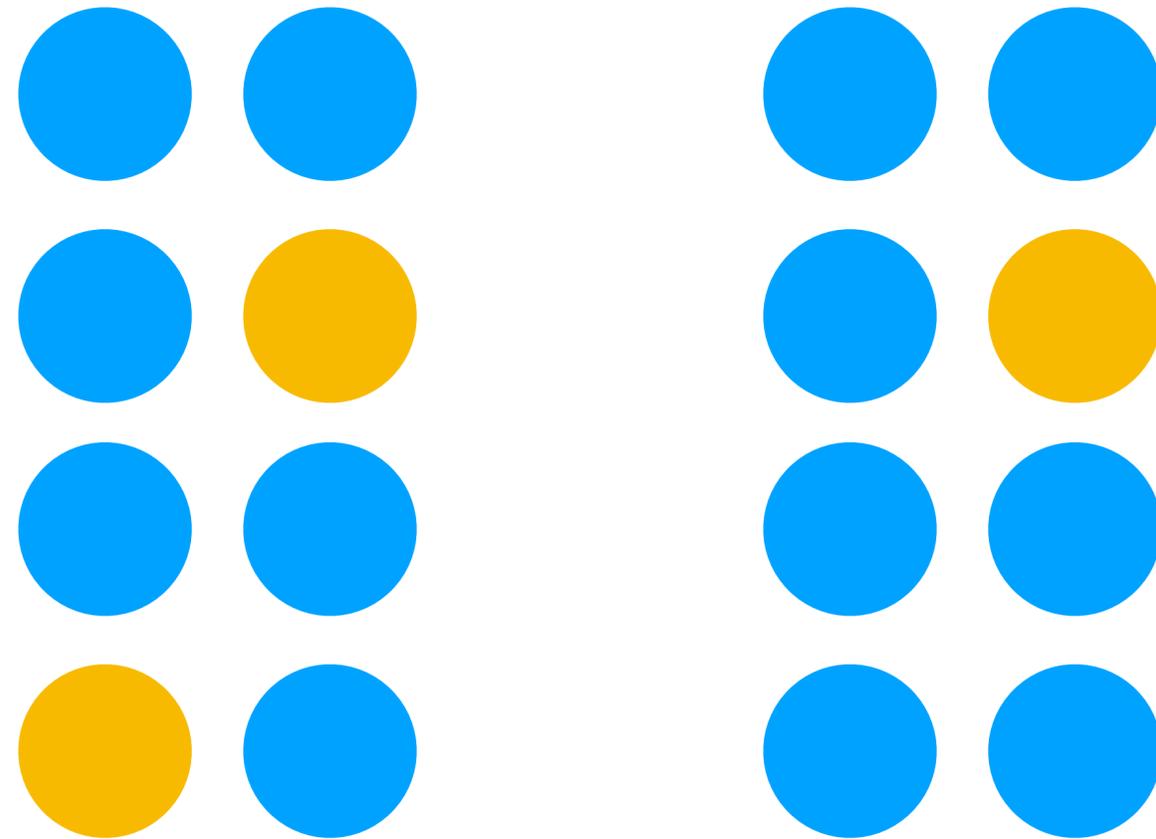
PROXIMITY



PROXIMITY



PROXIMITY



PROXIMITY



Features Business Explore Marketplace Pricing

Sign in

Sign in or Sign up

Built for developers

GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside millions of other developers.

Username

Pick a username

Email

you@example.com

Password

Create a password

We'll email you when you're invited, and when there are updates.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [Terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

NAVIGATION

1. Where **am** I?
2. Where have I **been**?
3. Where can I **go**?

NAVIGATION

ALFRED SUNG

NAVIGATION

Americas

7 November 2017



Canada police arrest five naked people after 'bizarre' car crash

10:00 AM



Mexican troops waging war on drug gangs not punished for rights abuses - report

10:00 AM

Canada groups launch court challenge to Quebec's face-covering ban

10:00 AM

Is the orthodoxy of Catholic poems coming to an end?

Andrew Ross

10:00 AM | 10:00 AM

Cycling the city: 'Take theft is not inevitable'. Vancouver milks out a cycle crime revolution

10:00 AM | 10:00 AM

6 November 2017



US polygamist detained in Mexico as police investigate deaths of three teens

10:00 AM

Argentinian lawyer Alberto Nisman was

environment

Tax savings are a snap.

METHODS

- a/b testing
- affinity diagramming
- bodystorming
- card sorting
- case studies
- cognitive walkthrough
- contextual inquiry
- cultural probes
- experience prototyping
- eye tracking
- focus groups
- heuristic evaluation
- interviews
- mental model diagramming
- participatory design
- personas
- questionnaires
- scenarios
- simulation
- storyboards
- surveys
- task analysis
- usability testing
- user journey mapping

METHODS

EMPIRICAL



with users
example: usability testing

ANALYTICAL



no users
example: heuristic evaluation

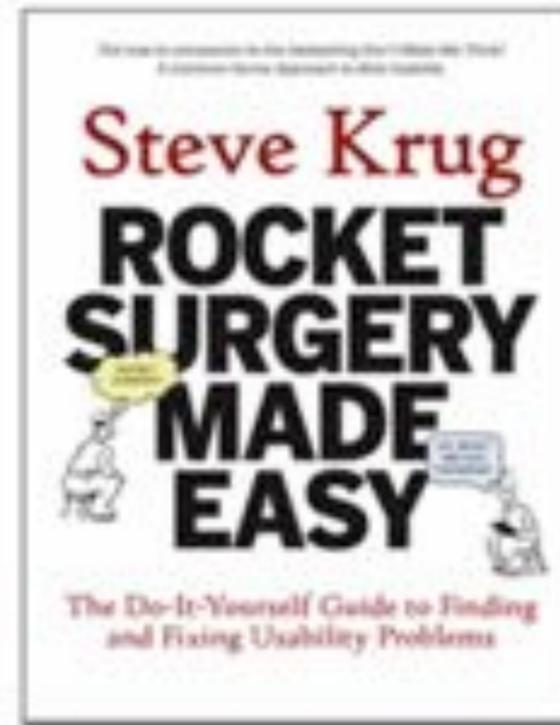
USABILITY TEST



USABILITY TEST

Demo Usability Test

for readers of



Rev. 1.1 / February 3, 2010
© 2010 Steve Krug
www.rocket surgerymadeeasy.com



Rocket Surgery Made Easy by Steve Krug: Usability Demo

156,692 views

👍 303 🗨️ 12 ➡️ SHARE 🌐 ⋮

HEURISTIC EVALUATION



HEURISTIC EVALUATION



Heuristic Evaluation - A System Checklist

1. Visibility of System Status

The system should always keep user informed about what is going on, through appropriate feedback within reasonable time.

#	Review Checklist	Yes	No	N/A	Comments
1.1	Does every display begin with a title or header that describes screen contents?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	No meaningful page headings.
1.2	Is there a consistent icon design scheme and stylistic treatment across the system?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
1.3	Is a single, selected icon clearly visible when surrounded by unselected icons?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
1.4	Do menu instructions, prompts, and error messages appear in the same place(s) on each menu?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Embedded error messages appear in different locations on the page
1.5	In multipage data entry screens, is each page labeled to show its relation to others?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
1.6	If overtype and insert mode are both available, is there a visible indication of which one the user is in?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
1.7	If pop-up windows are used to display error messages, do they allow the user to see the field in error?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
1.8	Is there some form of system feedback for every operator action?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Some feedback is too subtle
1.9	After the user completes an action (or group of actions), does the feedback indicate that the next group of actions can be started?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
1.10	Is there visual feedback in menus or dialog boxes about which choices are selectable?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
1.11	Is there visual feedback in menus or dialog boxes about which choice the cursor is on now?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
1.12	If multiple options can be selected in a menu or dialog box, is there visual feedback about which options are already selected?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
1.13	Is there visual feedback when objects are selected or moved?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
1.14	Is the current status of an icon clearly indicated?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
1.15	Is there feedback when function keys are pressed?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	

HEURISTIC EVALUATION

5. Help Users Recognize, Diagnose, and Recover From Errors

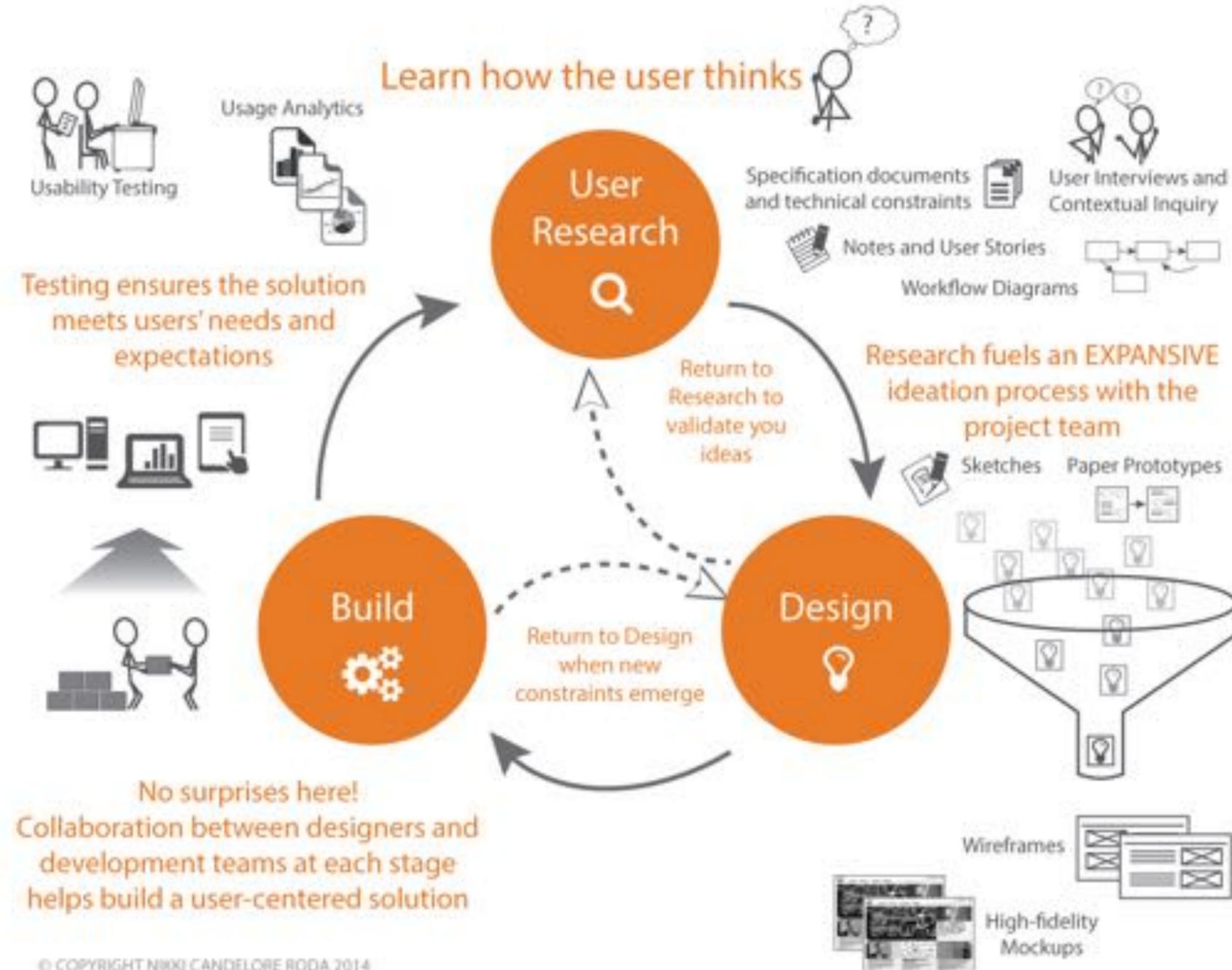
Error messages should be expressed in plain language(NO CODES).

#	Review Checklist	Yes	No	N/A	Comments
5.1	Is sound used to signal an error?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	
5.2	Are prompts stated constructively, without overt or implied criticism of the user?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.3	Do prompts imply that the user is in control?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.4	Are prompts brief and unambiguous.	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.5	Are error messages worded so that the system, not the user, takes the blame?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Example: "You must enter a name for this order form".
5.6	If humorous error messages are used, are they appropriate and inoffensive to the user population?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
5.7	Are error messages grammatically correct?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.8	Do error messages avoid the use of exclamation points?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.9	Do error messages avoid the use of violent or hostile words?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.10	Do error messages avoid an anthropomorphic tone?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.11	Do all error messages in the system use consistent grammatical style, form, terminology, and abbreviations?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.12	Do messages place users in control of the system?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	
5.13	Does the command language use normal action-object syntax?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
5.14	Does the command language avoid arbitrary, non-English use of punctuation, except for symbols that users already know?	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	
5.15	If an error is detected in a data entry field, does the system place the cursor in that field or highlight the error?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	Instances where users must hunt for the offending field.
5.16	Do error messages inform the user of the error's severity?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.17	Do error messages suggest the cause of the problem?	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	
5.18	Do error messages provide appropriate semantic information?	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	

HEURISTIC EVALUATION

	Problem	Heuristic broken	Recommendations	Severity rating (1-4)	Number of evaluators that found issue
CONSISTENCY	There is no one consistent method used for editing. This increases the learning curve for the user.	Consistency Efficiency	Revisit the editing layout and maintain consistency in all the editing layouts.	 ■■■■	👤👤 👤
	There is no way for the user to know that Identity Page is a Folotek product because of the lack of branding.	Consistency	Include elements of Folotek brand in the form of colors and logo.	 ■■■■	👤👤 👤
EFFICIENCY	The clickable area for editing and adding content is very small. This makes it harder for the user to click on it.	Feedback Efficiency	Increase the target area/ clickable area for "EDIT/ADD" option during editing.	 ■■■■	👤👤 👤
	There are no keyboard shortcuts for an expert user.	Memory Efficiency	Provide keyboard shortcuts to edit content on Identity Page. This change is required only if the users are highly skilled in using Identity Page.	 ■■■■	👤👤 👤

USER-CENTERED DESIGN



UCD

USER
RESEARCH

PROTOTYPING

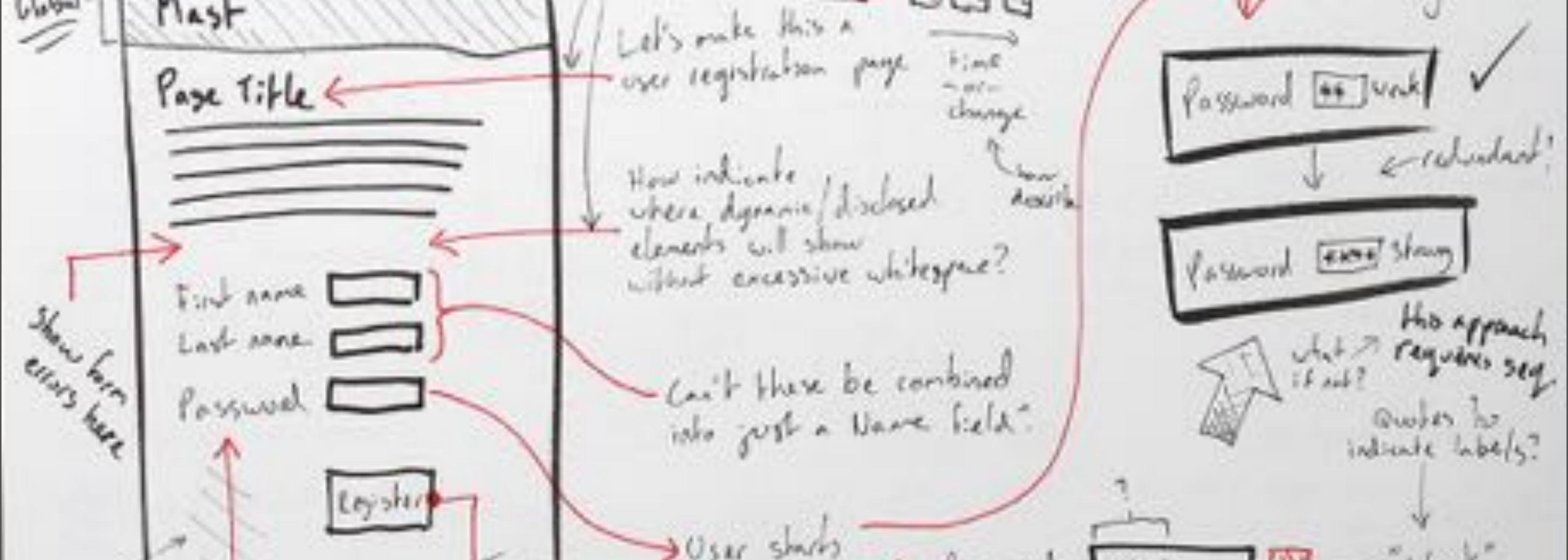
EVALUATION



USER RESEARCH



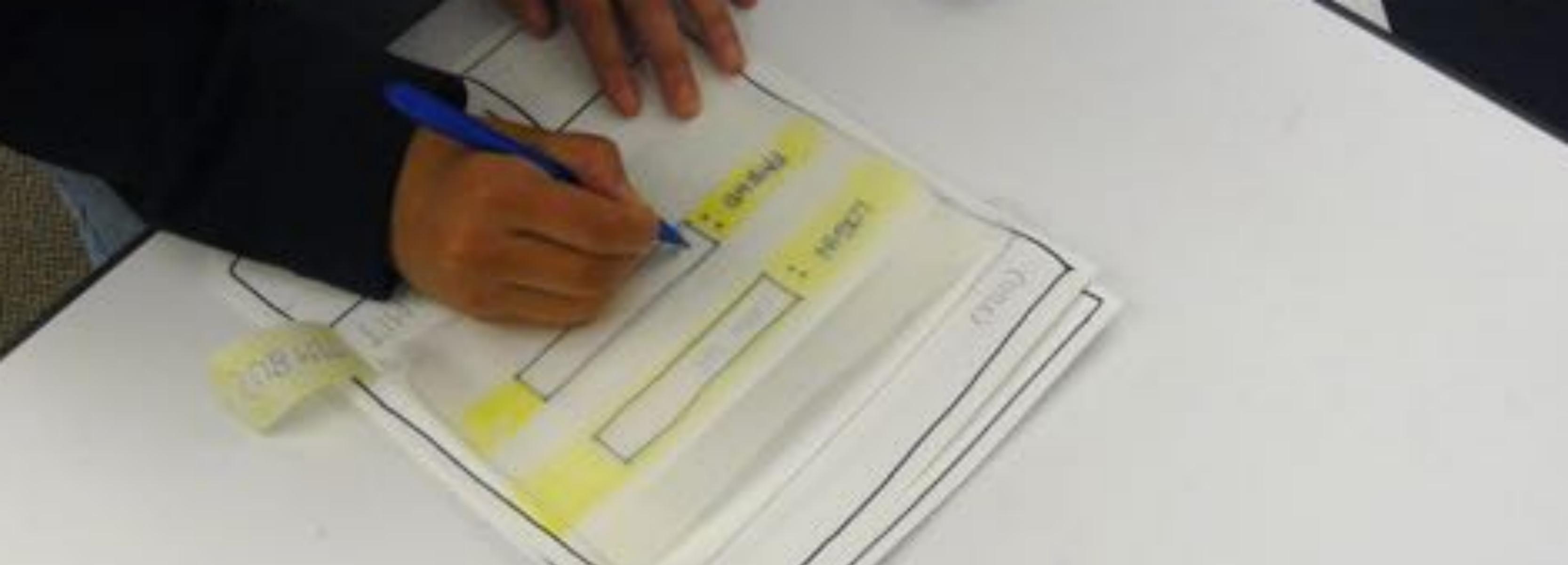
USER RESEARCH



PROTOTYPING



PROTOTYPING



EVALUATION



EVALUATION



EVALUATION



EVALUATION

OBJECTIVES

1. **value** the importance of usability and user-centered design
2. be able to **identify** some common usability problems
3. **appreciate** various strategies for identifying and fixing common usability problems
4. be able to **communicate** with usability experts in an informed way

