

UX in Agile

Tools and Methods

Julián Grigera, LIFIA - UNLP / CONICET - Argentina



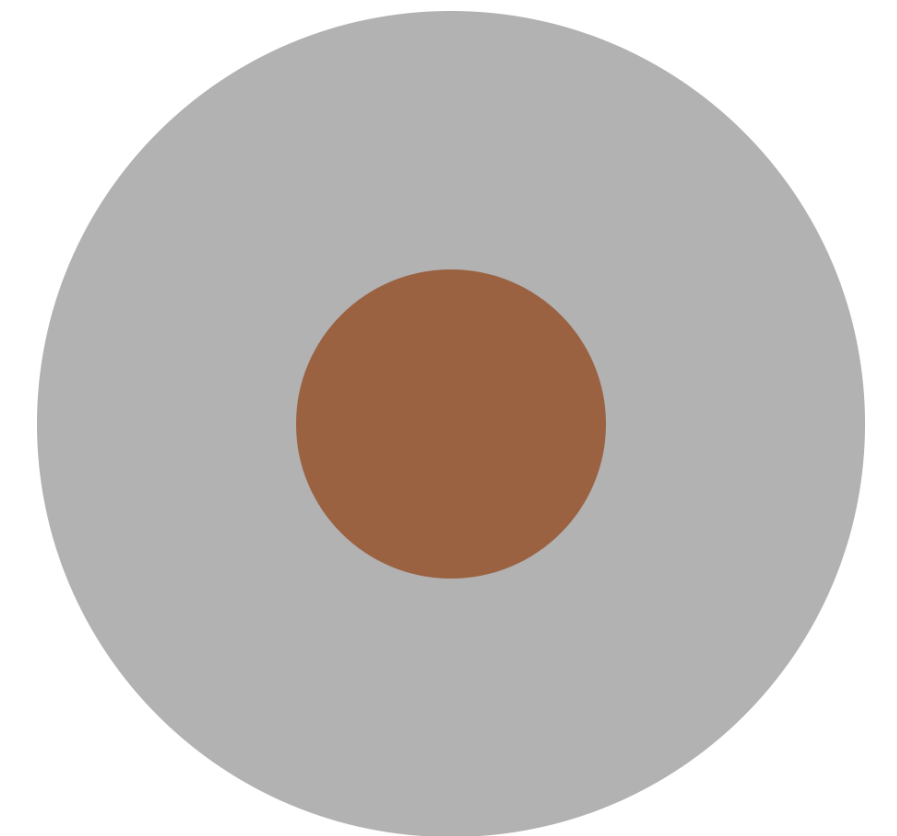
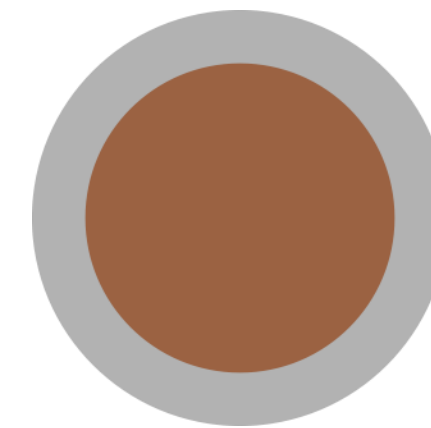
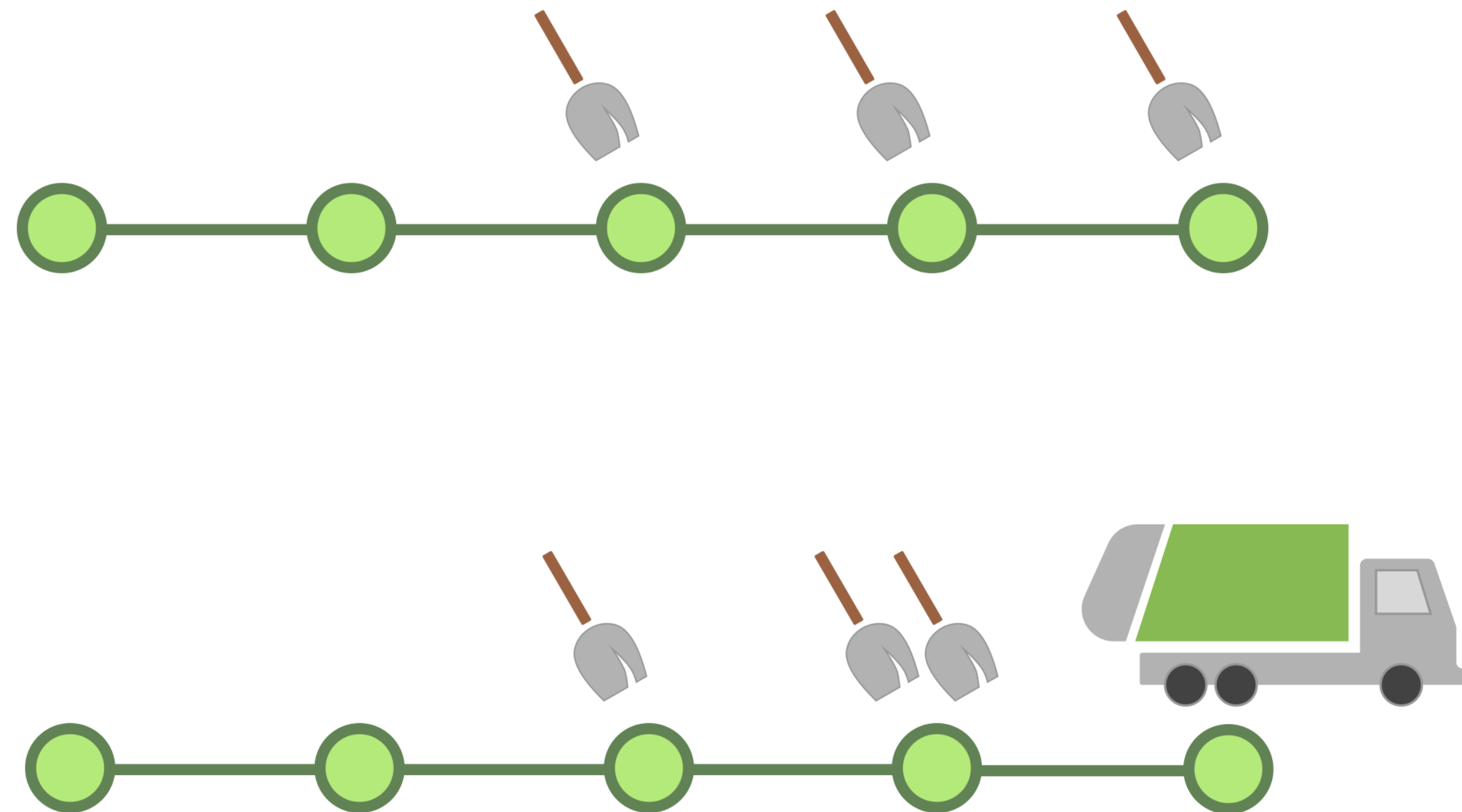
UX Debt



Technical Debt

Technical Debt

Cost of short-term solutions



“A collection of design or implementation constructs that are expedient in the short term, but set up a technical context that can make future changes more costly or impossible”

Avgeriou et al., 2016

Report from Dagstuhl Seminar 16162

Managing Technical Debt in Software Engineering

Edited by

Paris Avgeriou¹, Philippe Kruchten², Ipek Ozkaya³, and Carolyn Seaman⁴

¹ University of Groningen, Groningen, NL, paris@cs.rug.nl

² University of British Columbia, Vancouver, BC, CA, pbk@ece.ubc.ca

³ Carnegie Mellon University, Pittsburgh, PA, US, ozkaya@sei.cmu.edu

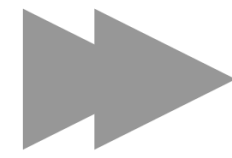
⁴ University of Maryland, Baltimore County, MD, US, cseaman@umbc.edu

Abstract

This report documents the program and outcomes of Dagstuhl Seminar 16162, “Managing Technical Debt in Software Engineering.” We present the key findings of the seminar.

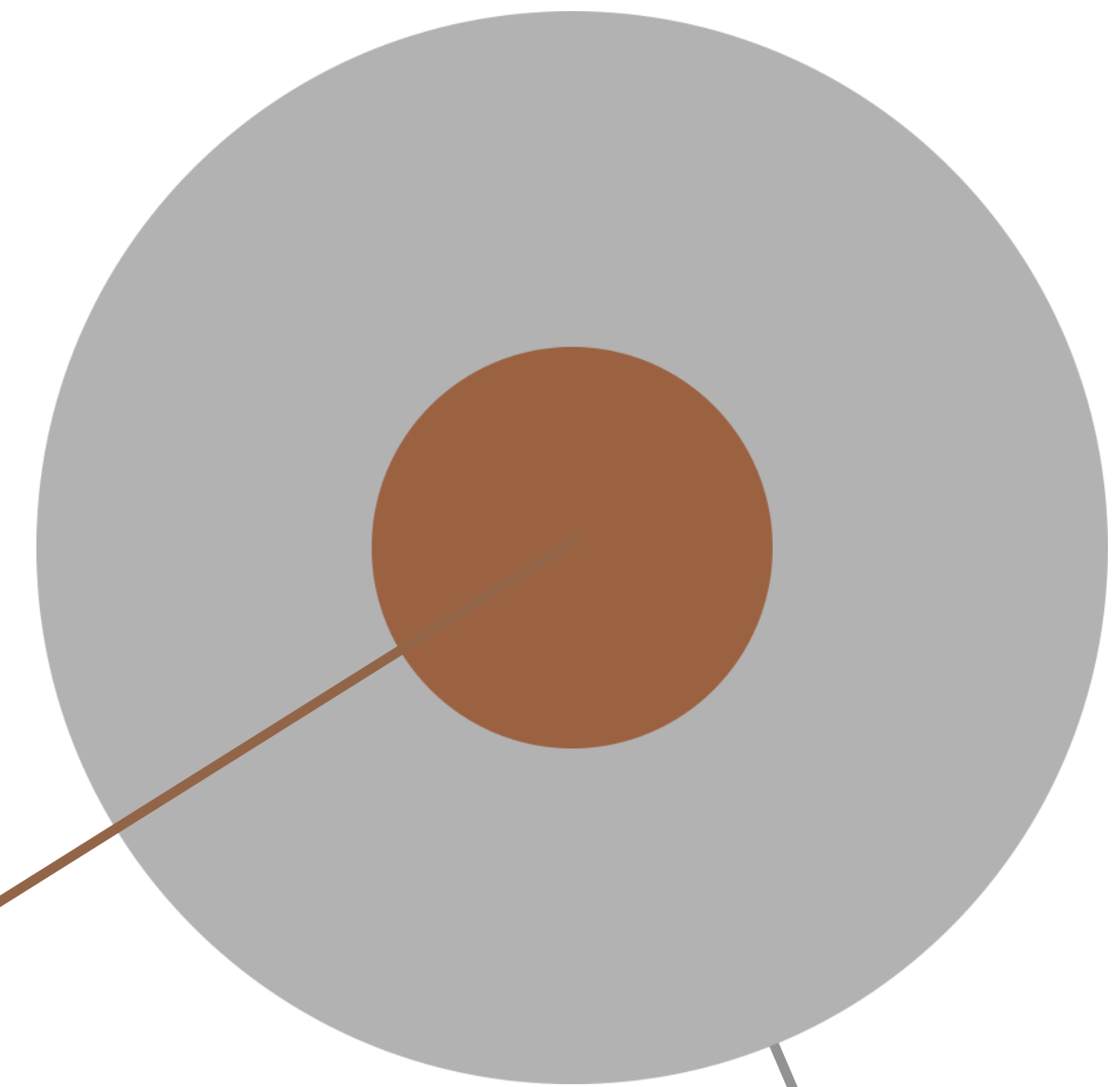
Technical Debt

Components



Refactoring

Cunningham, 1992

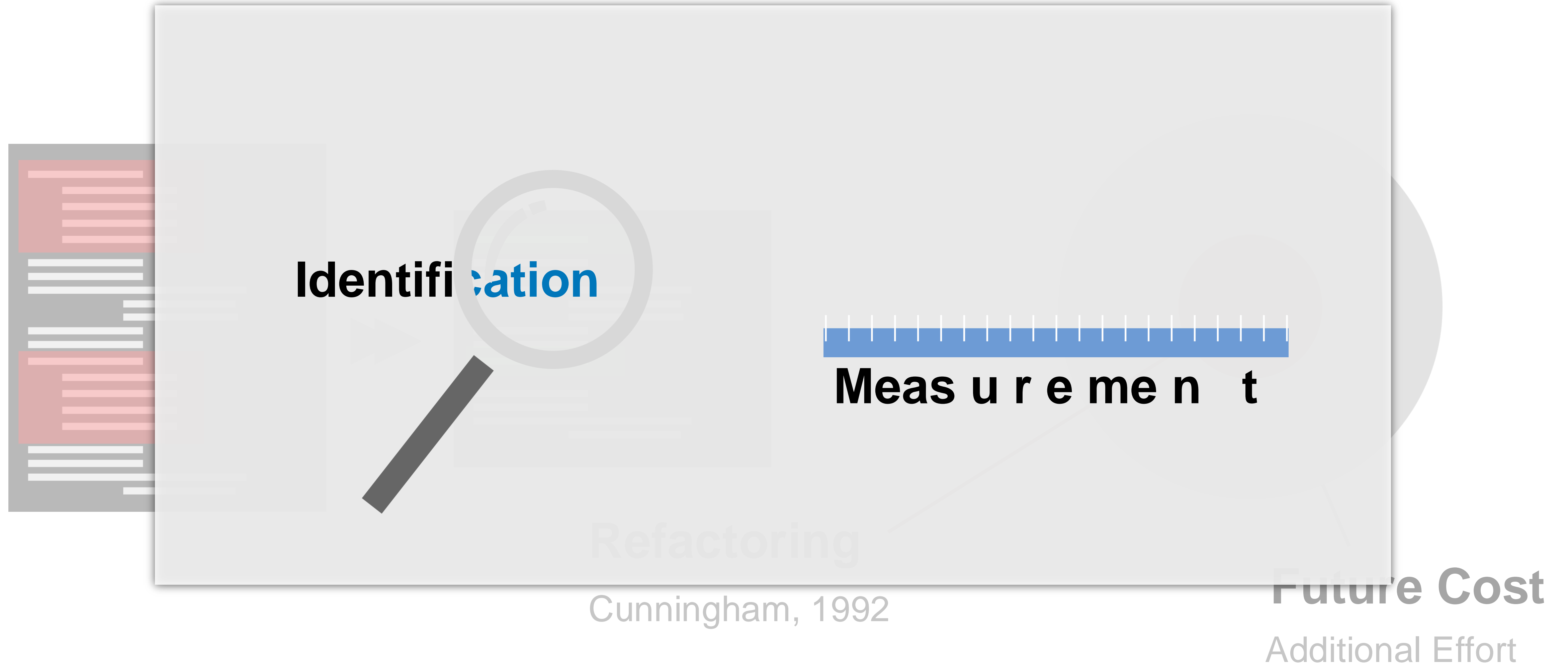


Future Cost

Additional Effort

Technical Debt

Components



UX Debt

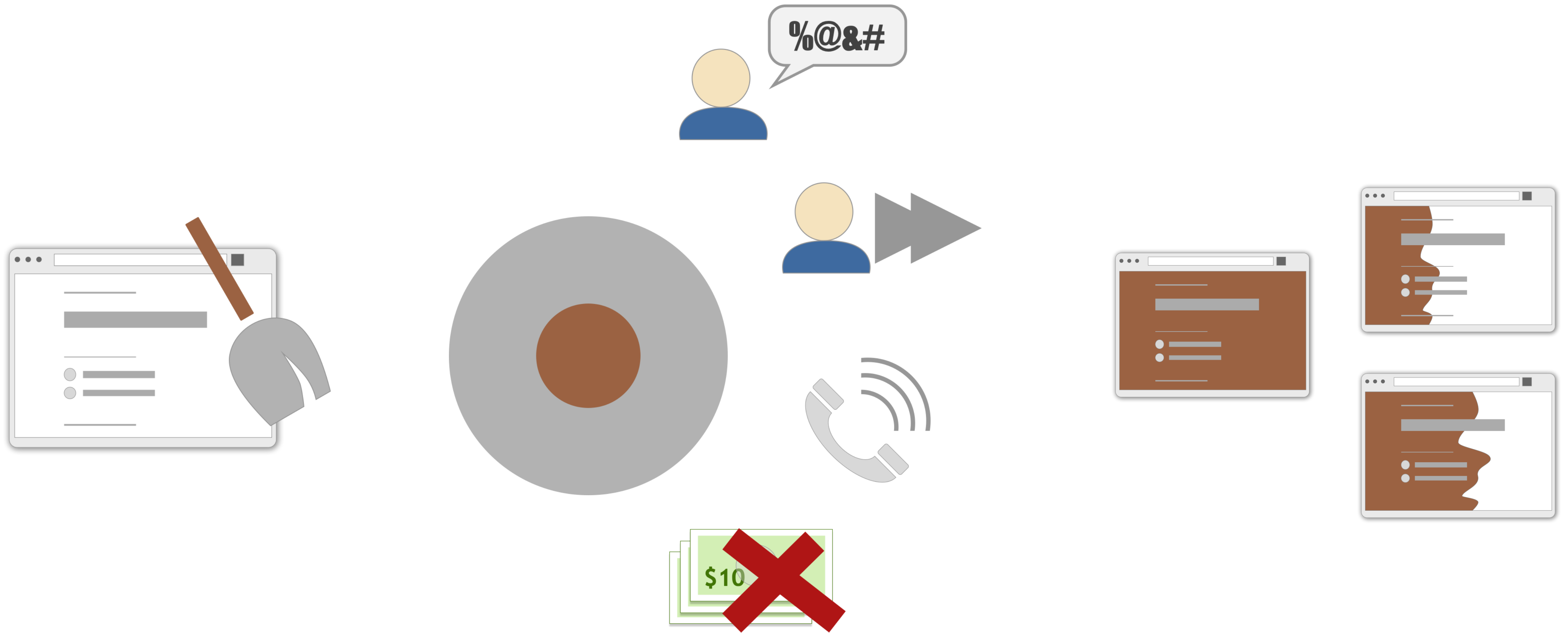
UX Debt

UX components

- **Pragmatic aspects of interaction**
Functionality, interactive behavior, user skills, context of use
- **Hedonic**
Brand image, presentation, user previous experiences

UX Debt

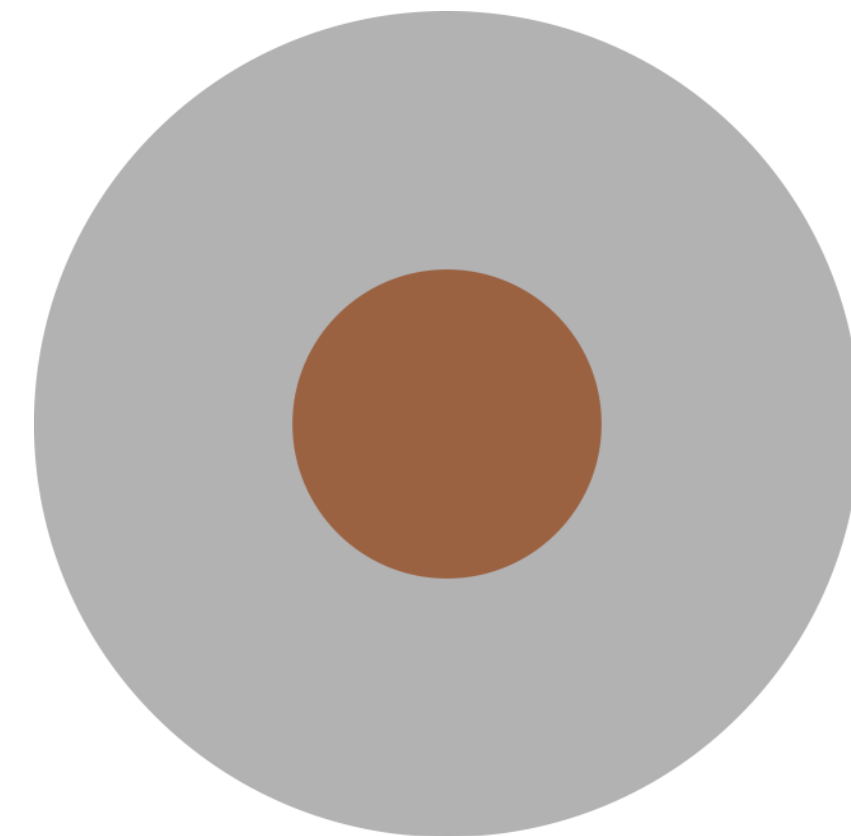
As a kind of TD



UX Debt

Concepts

- UXDebt has a cost with both principal and interest components
- For end-users but also for the development team
- Technical and non-technical effects



UX Debt

State of the Art

- Literature indicate that developers pay little attention to UX issues **after the initial design** stage
- UXDebt as a type of TD can **raise awareness** in the agile team and stakeholders

UX Debt

State of the Art

- “not quite right design which we postpone making it right” (Kuusinen, 2016)
- Literature indicate that developers pay little attention to UX issues after the initial design stage
- UX Debt as a type of Debt can raise awareness in the agile team and stakeholders



Identification



Measurement

UX Debt: Study

Findings

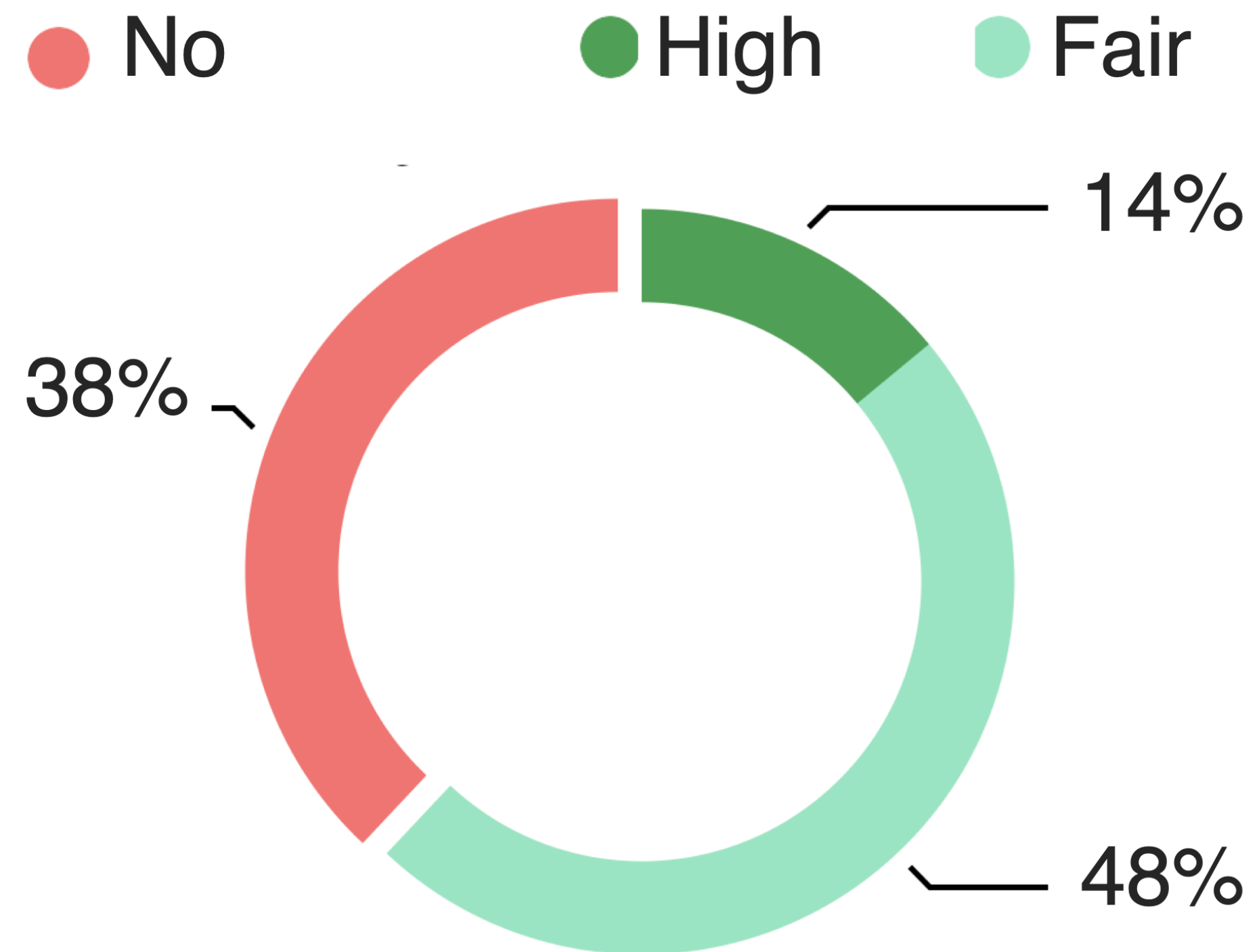
UX Debt Study

1. To what extent are practitioners familiar with the concept of UXDebt?
2. What kind of UX tasks and products generate the most negative impact on UXDebt?
3. How much work time is wasted because of UXDebt?
4. How much is the team aware of the acquired UXDebt?
5. What is the difference between the collective management of UXDebt and individual (UX-related subgroup) management?
6. In what way does the age of the software system affect the precedent questions?
7. In what way are the different roles in the team affected by the precedent questions?
8. How is UXDebt tracked? What tools do practitioners use?

Findings

UX Debt Study

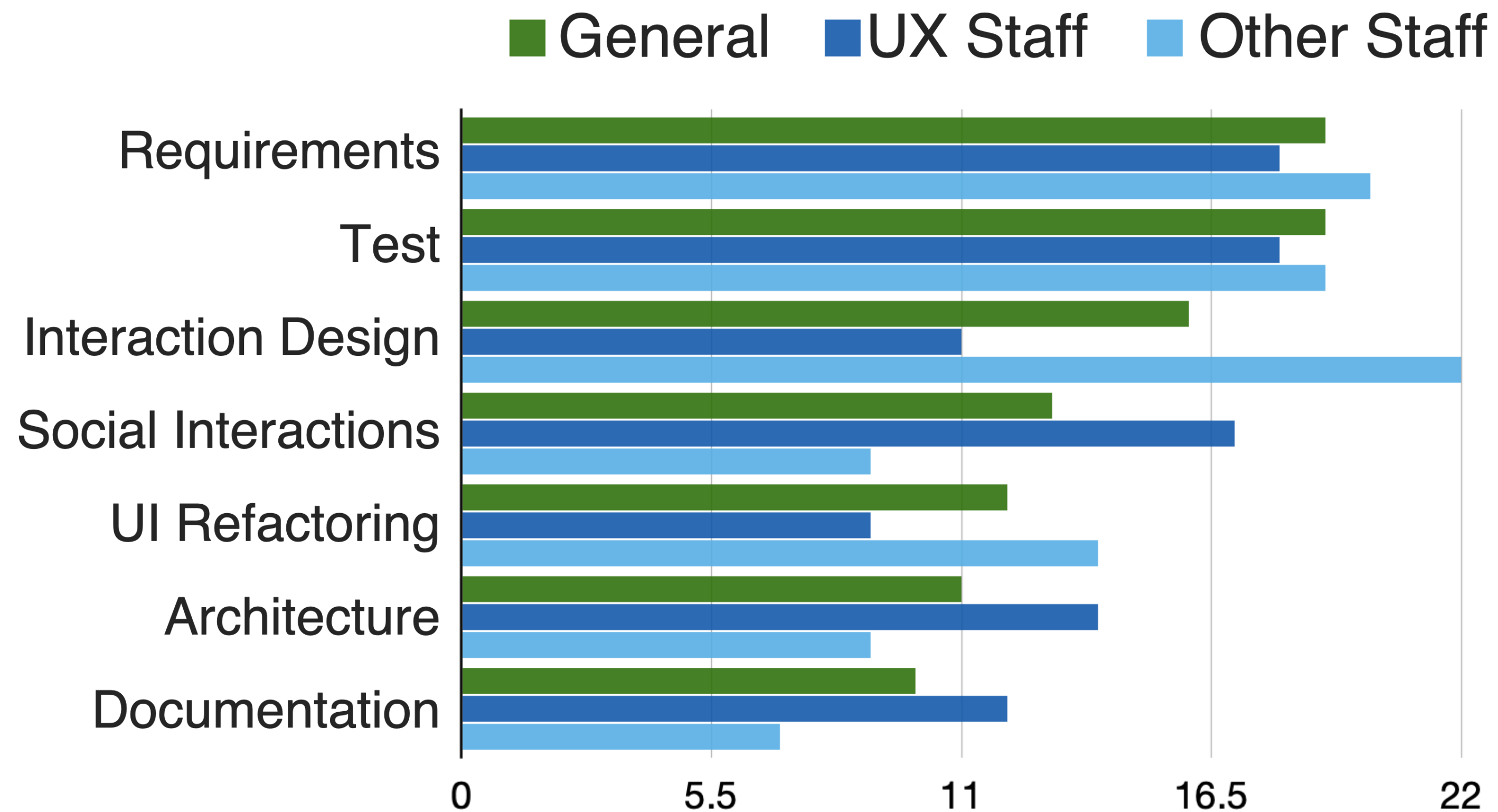
1. To what extent are practitioners familiar with the concept of UXDebt?



Findings

UX Debt Study

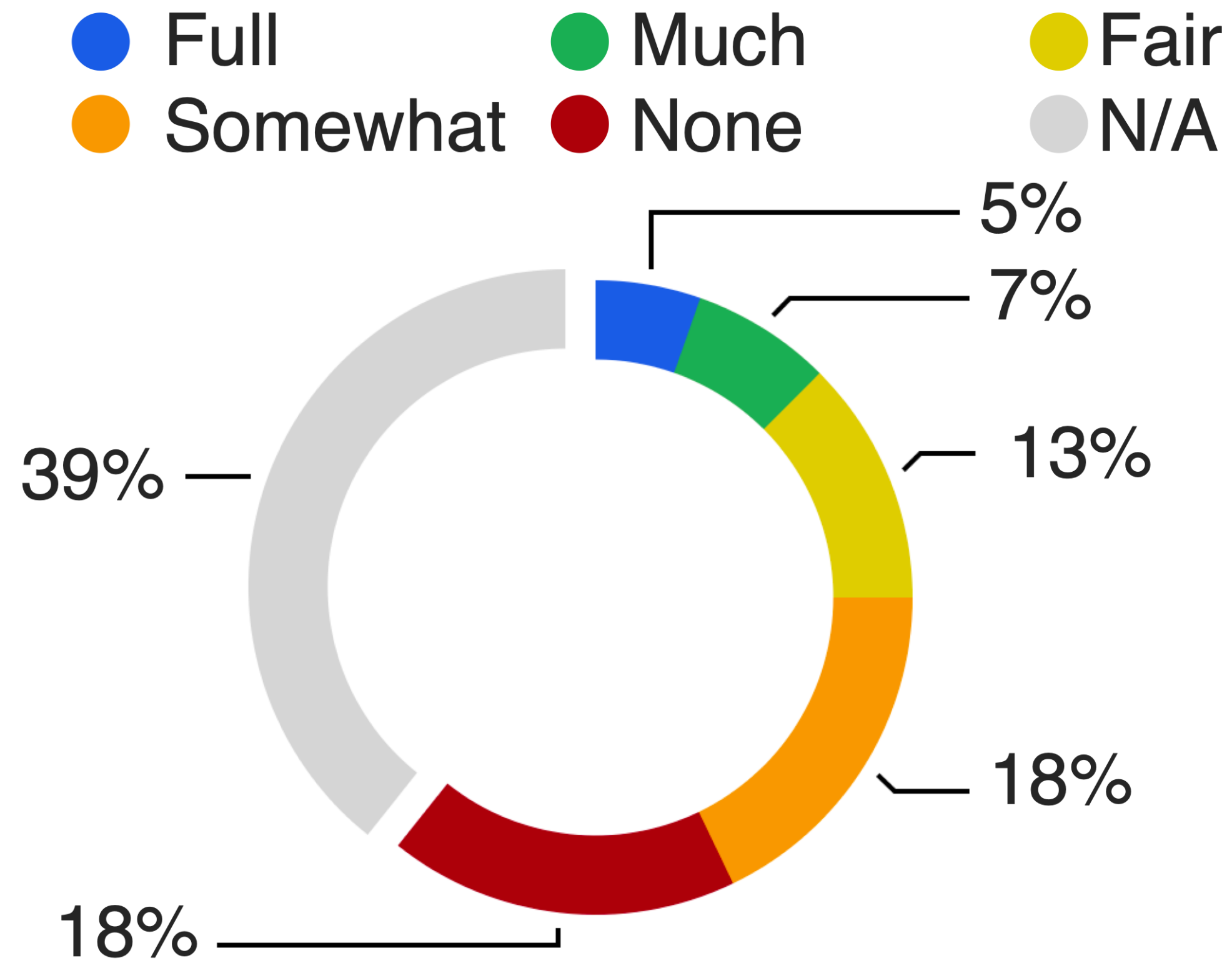
2. What kind of UX tasks and products generate the most negative impact on UXDebt?



Findings

UX Debt Study

4. How much is the team aware of the acquired UXDebt?



Findings

UX Debt Study

Lesson 1: UXDebt is a phenomenon recognized by practitioners

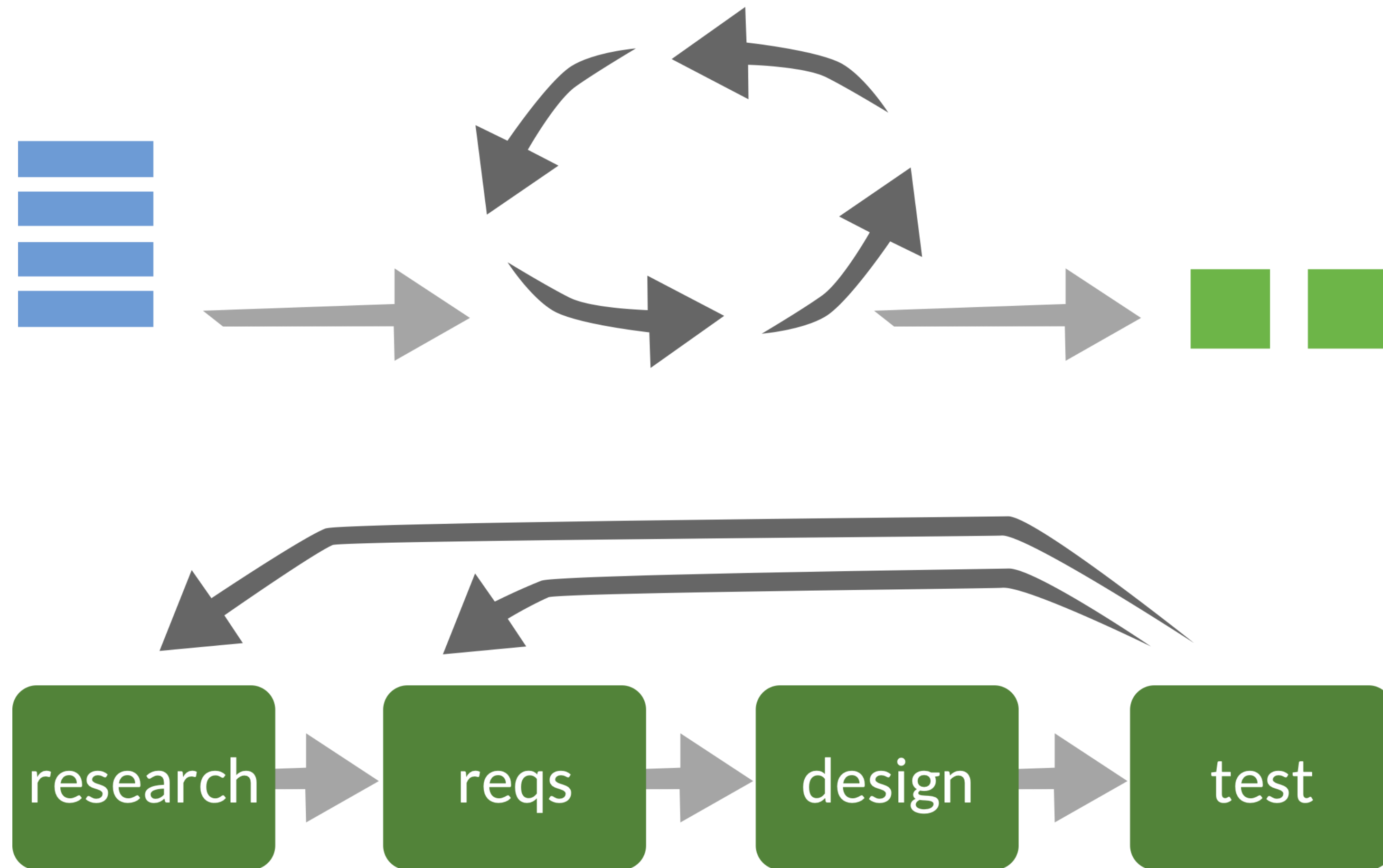
Lesson 2: UXDebt receives little attention and resources

UX + Agile Challenges



Agile and UX

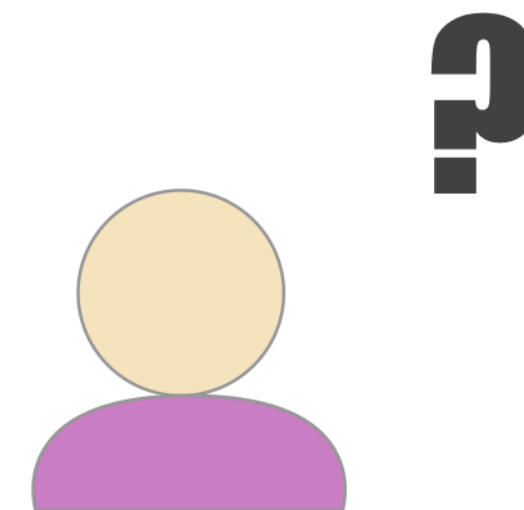
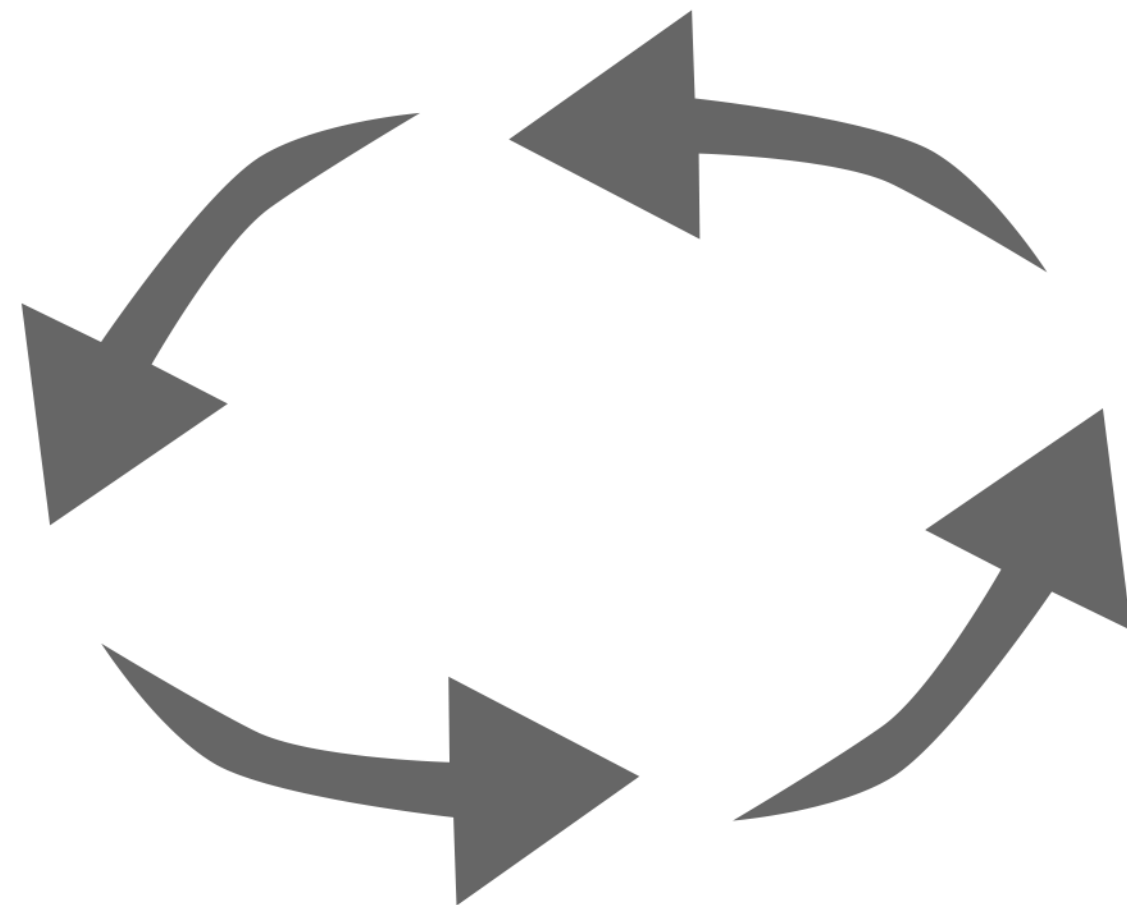
Challenges



Agile and UX

Challenges

- The **role** and **responsibility** of UX professionals within an agile team are not clearly defined
- Neither is the **timing** of their responsibilities within an agile cycle



Agile UX

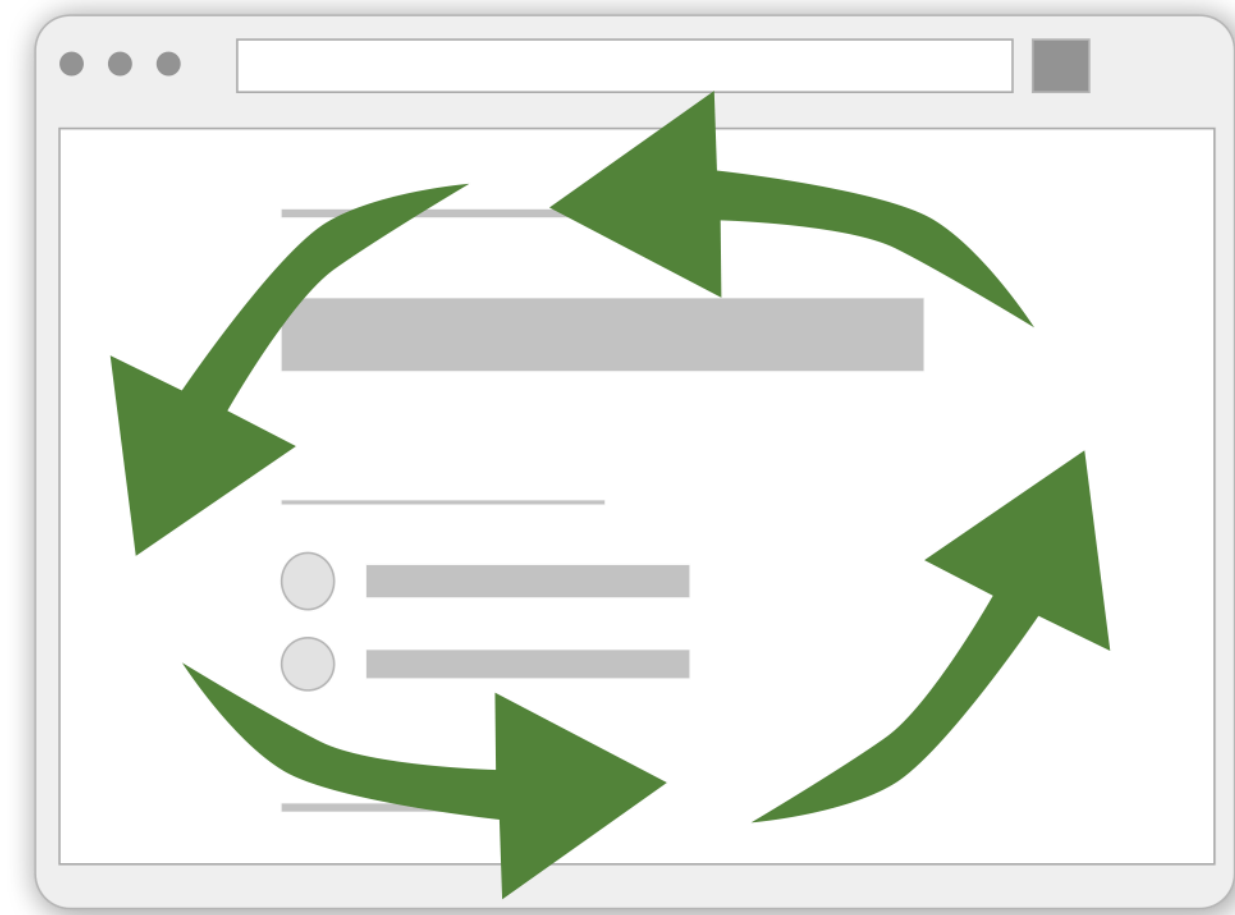
Incorporating UX into Agile Cycles

- Collaborative culture
- UX work ahead of development sprints
- Proactive UX professionals
- Institutionalized process to protect user-centered design

Lean UX

How to cope

- UX in quick agile cycles (MMF)
 - Constant Collaboration
- Deliverables >> *Slight Improvements*
- Based on **Hypotheses** and **Evidence**



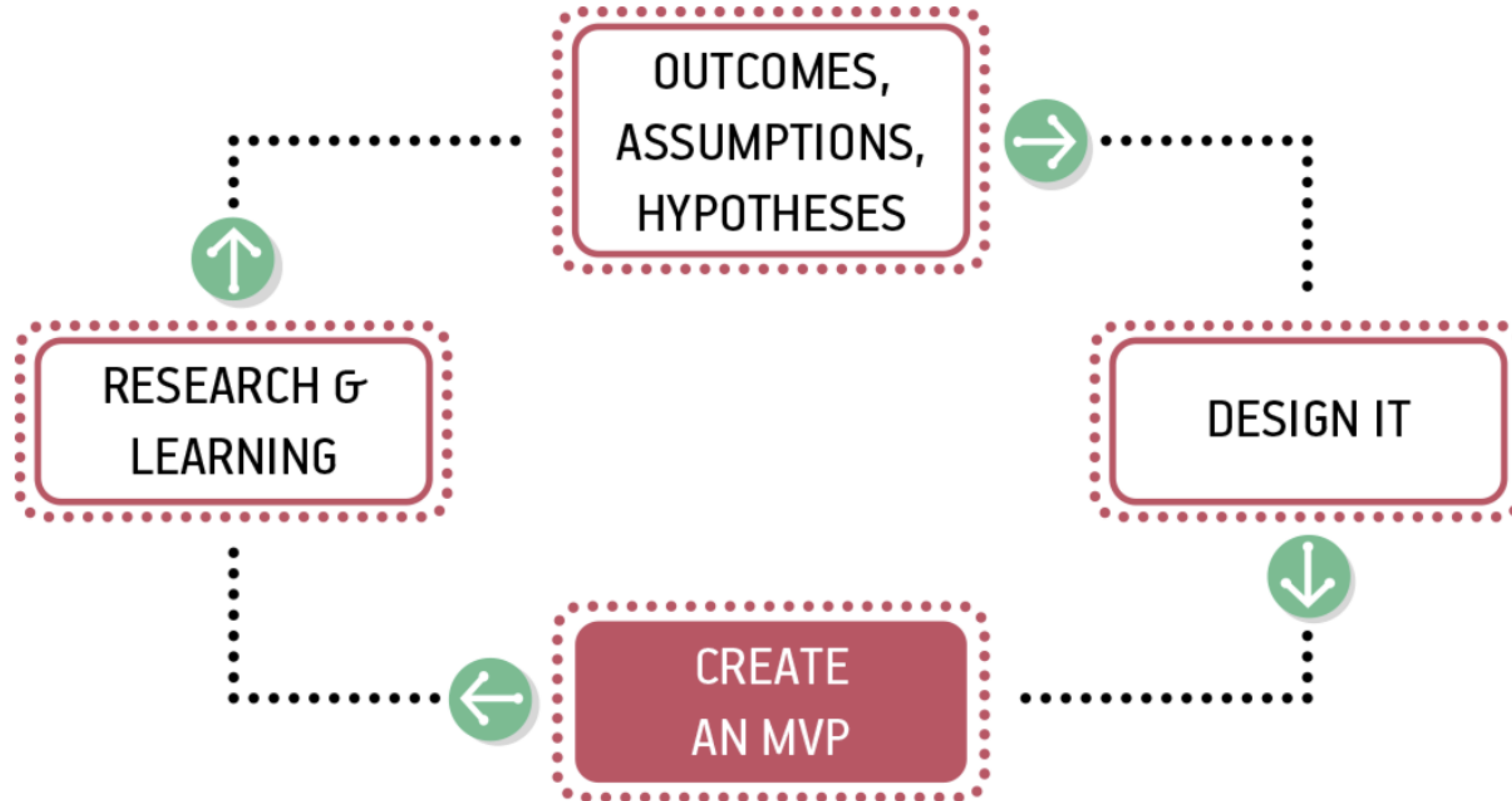
Lean UX

Principles

- Act as one team
- Solve the right problem
- Design collaboratively
- Exercise flexibility
- De-emphasize deliverables

Lean UX

Cycle



Lean UX

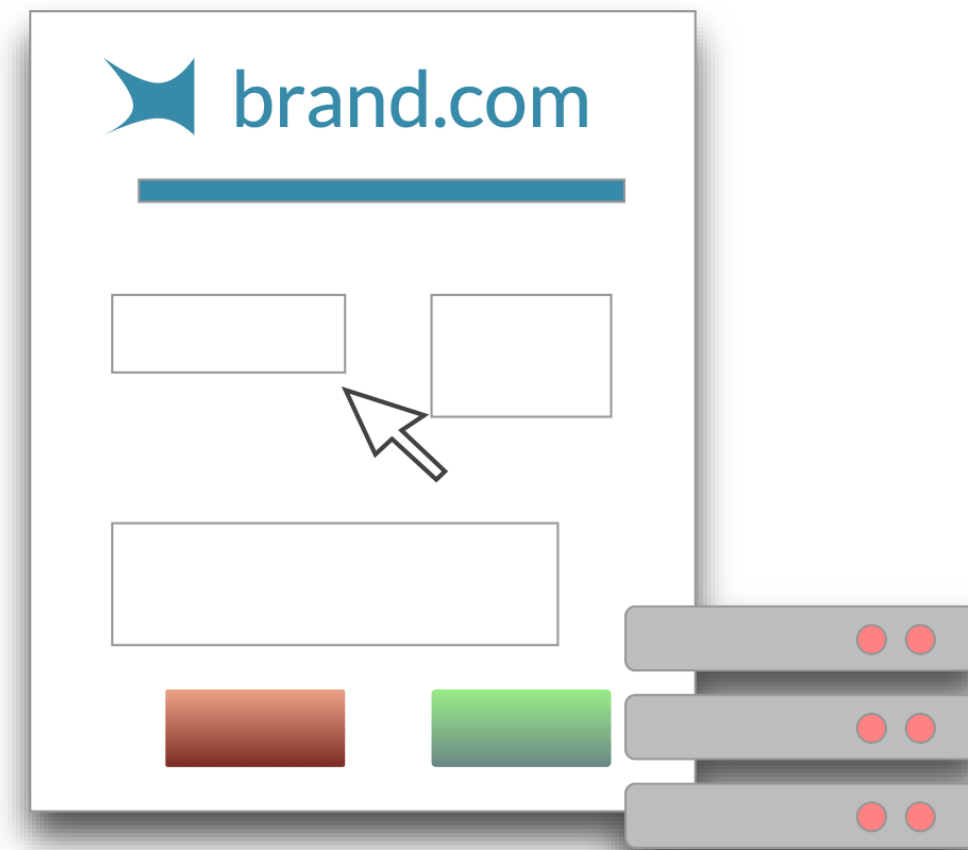
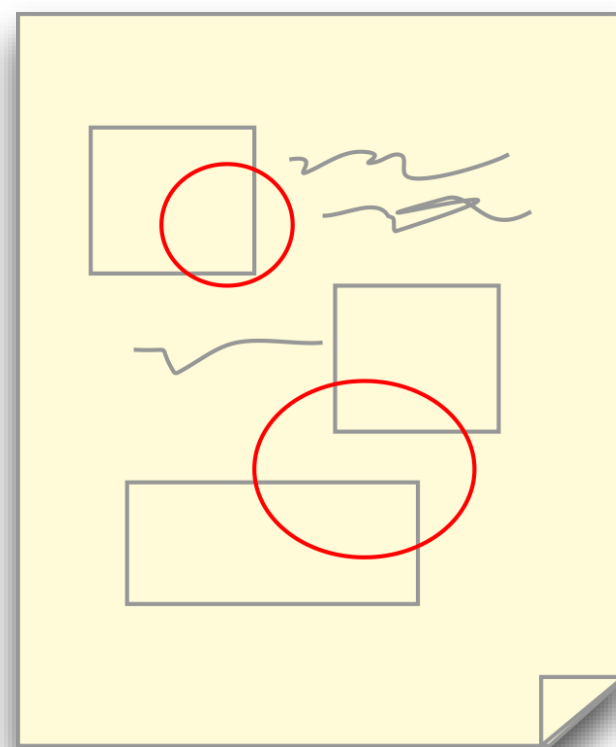
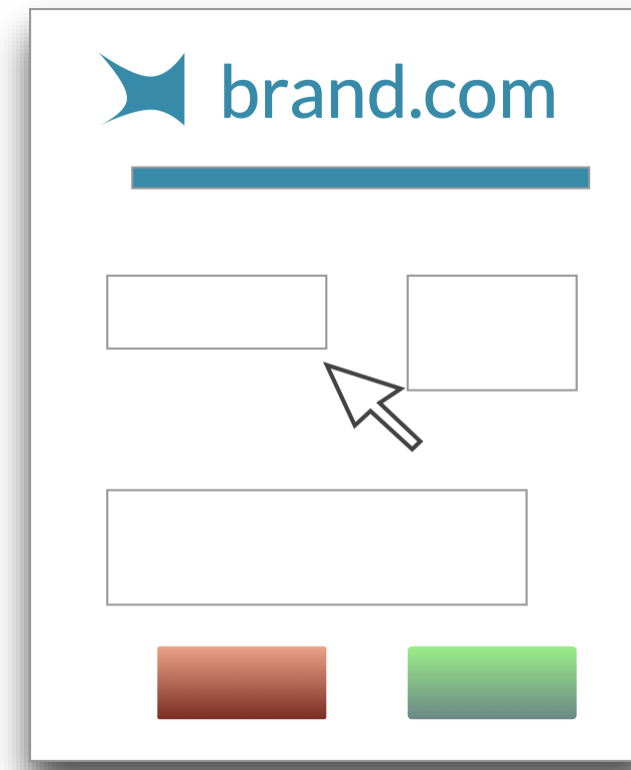
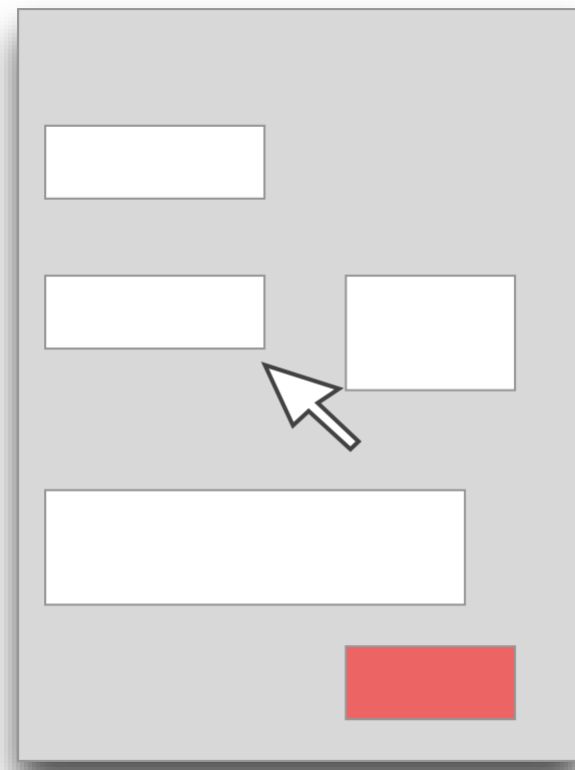
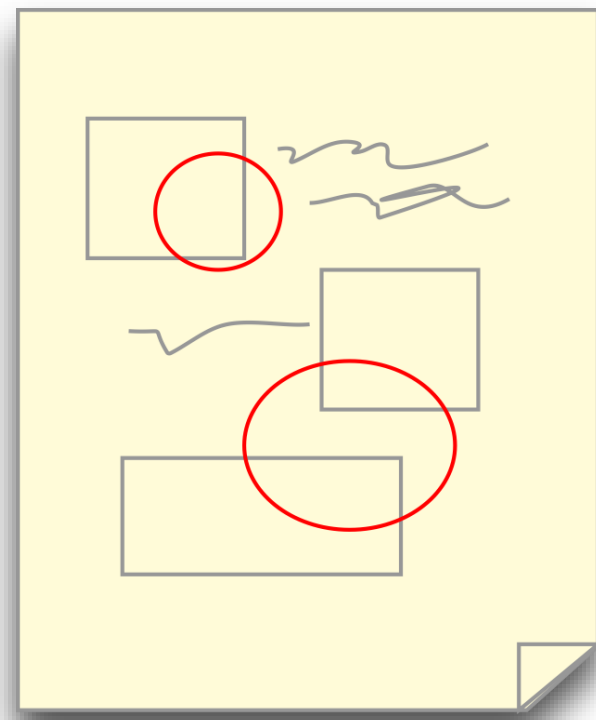
Humility Mindset

*“Lean UX has no time for heroes. The entire concept of design as a hypothesis immediately dethrones notions of heroism; as a designer, you must expect that **many of your ideas will fail in testing**. Heroes don’t admit failure. But Lean UX designers embrace it as part of the process.”*

Lean UX: Designing Great Products with Agile Teams

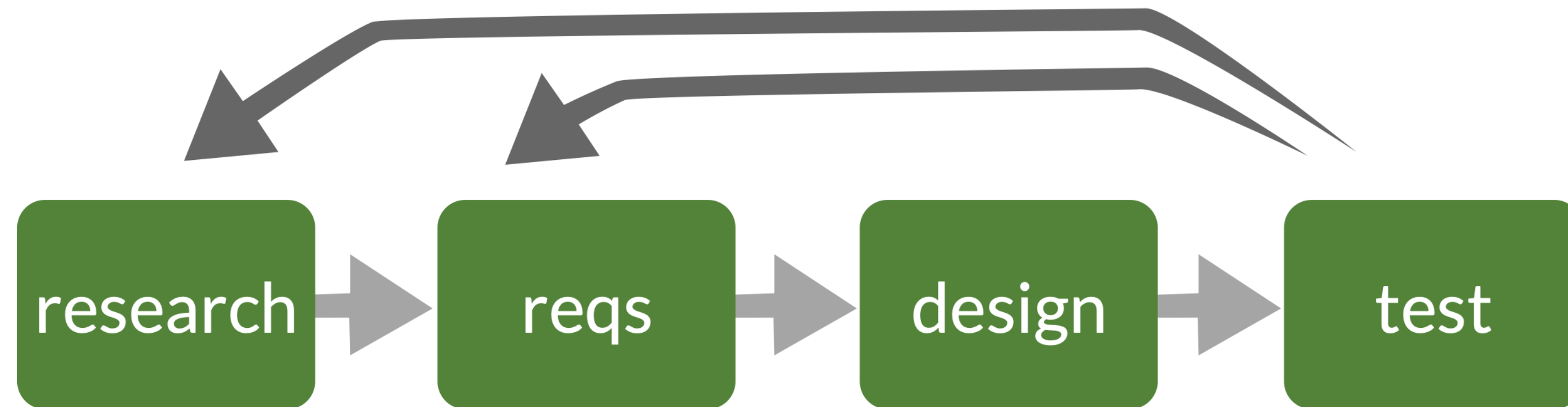
Lean UX

Prototyping for Requirements



UCD

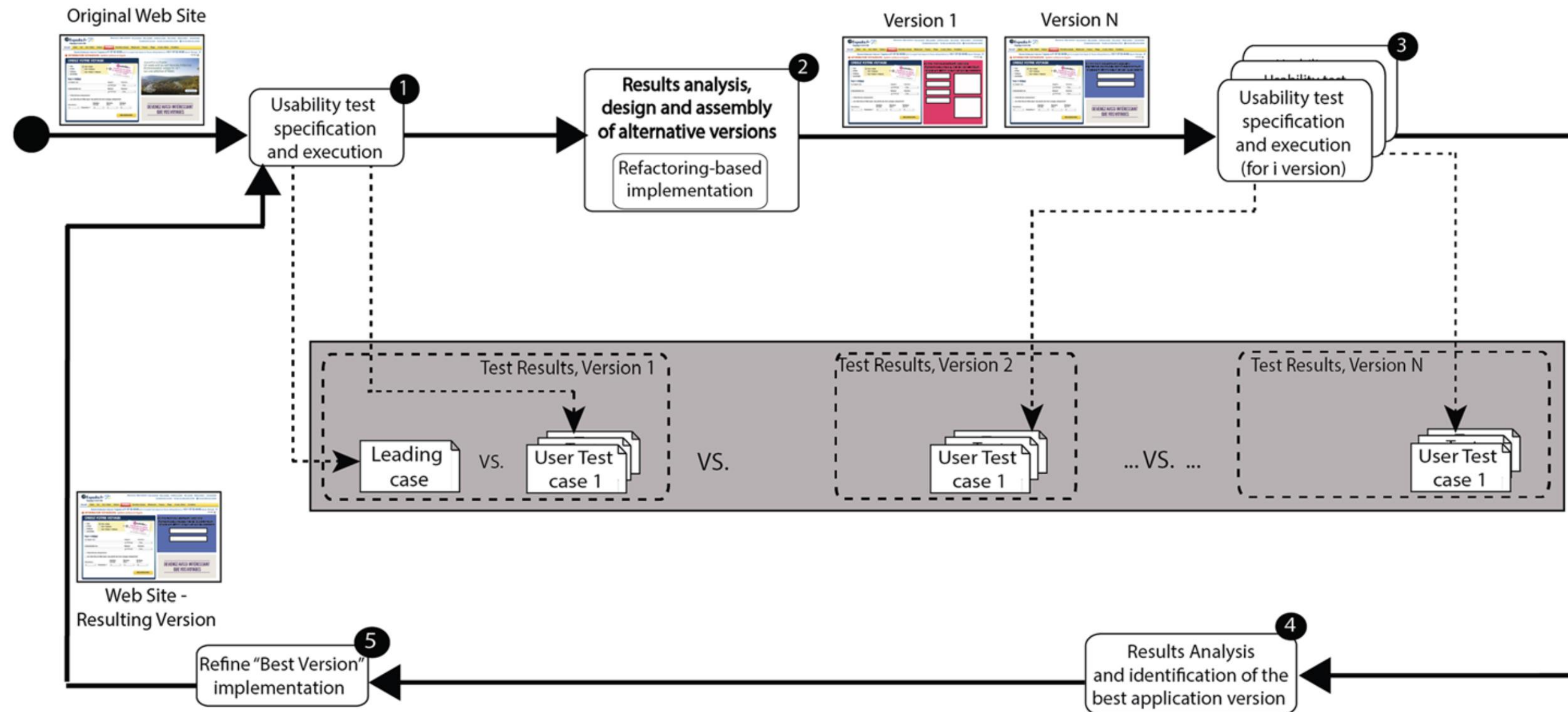
Dealing with (user) Test Results



overlapping UX in Agile

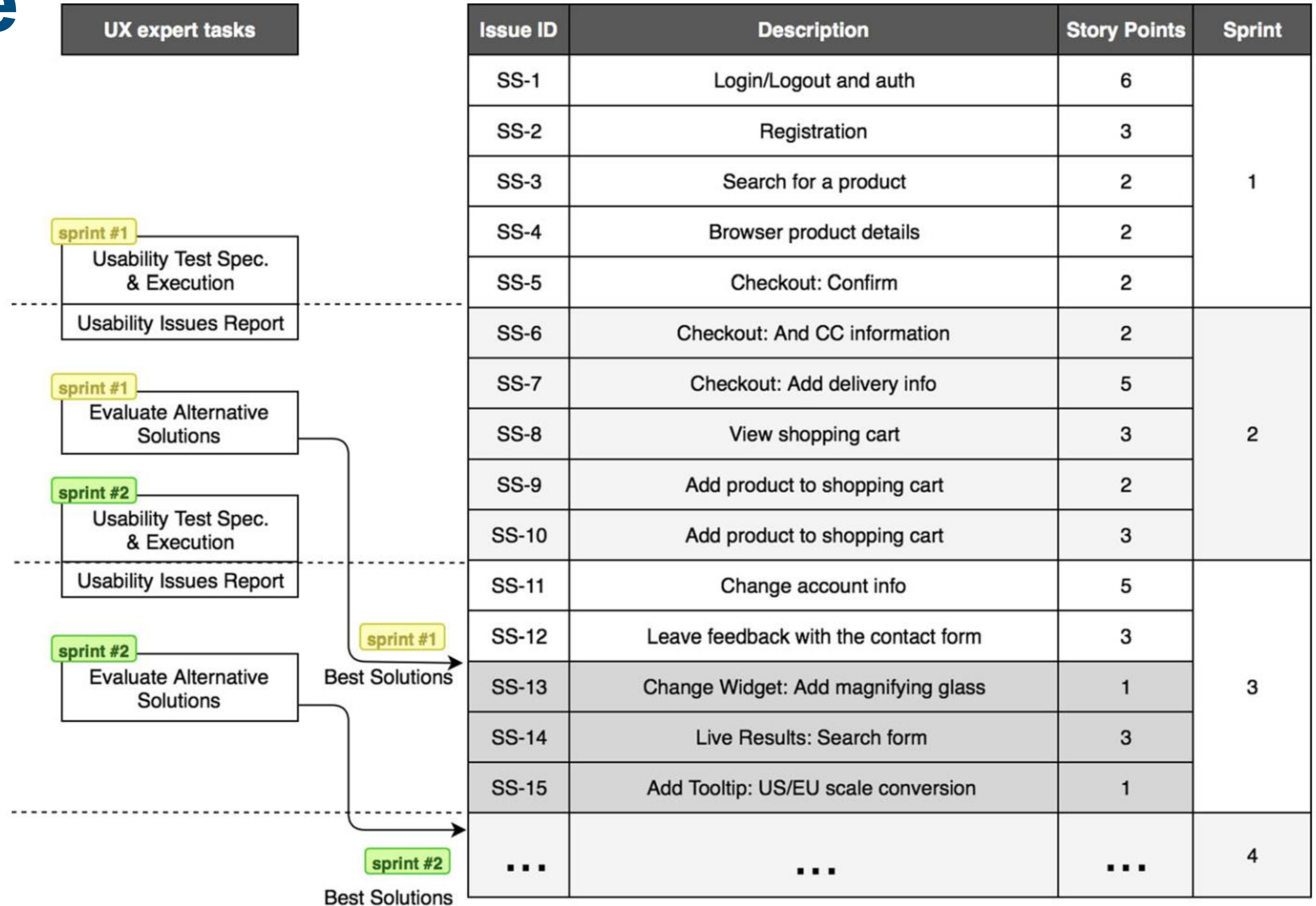
UX in Agile

Proposal



UX in Agile

Proposal

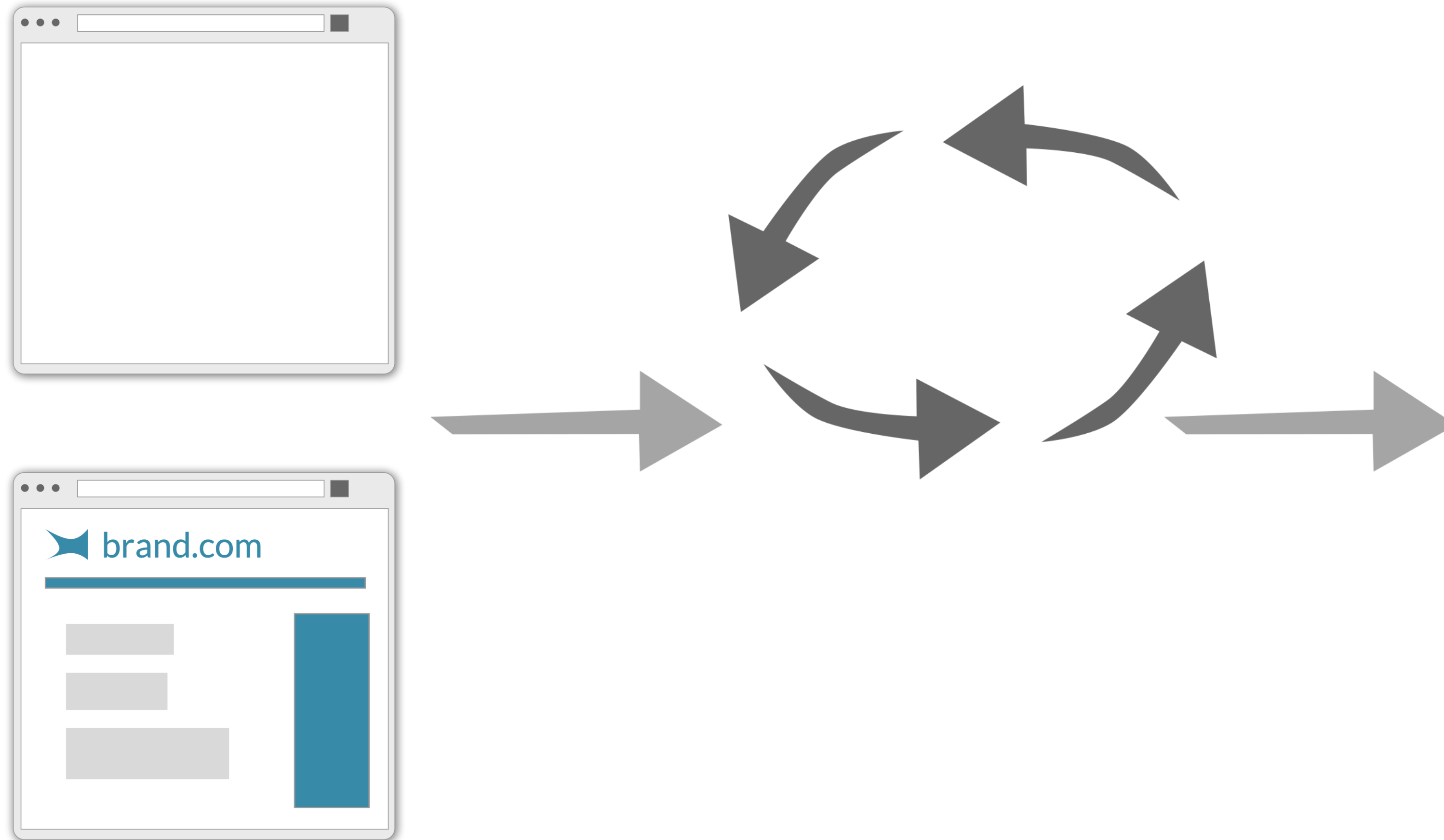


Evaluating UX



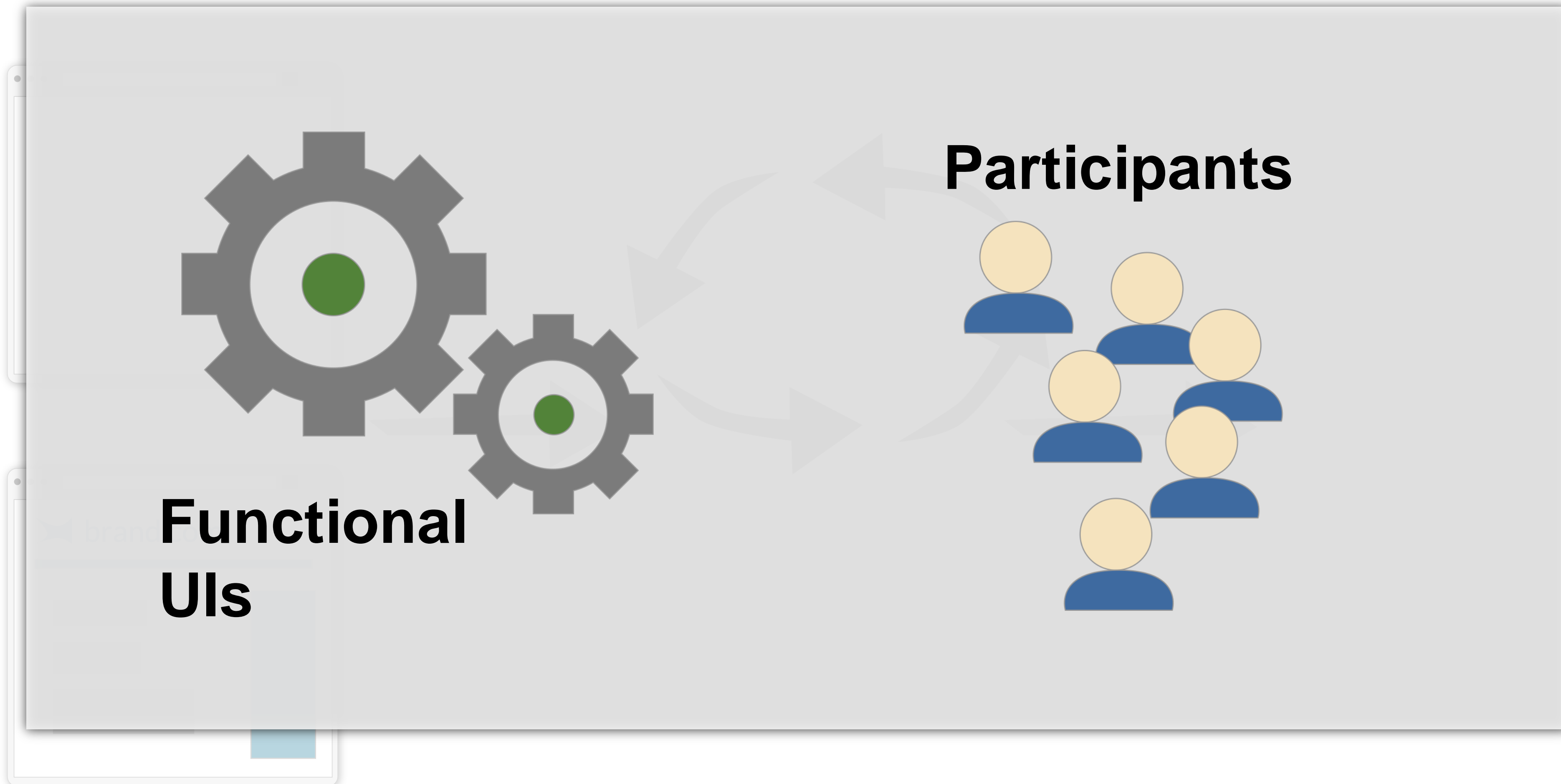
UX Diagnose

User Tests



UX Diagnose

User Tests



UX Diagnose

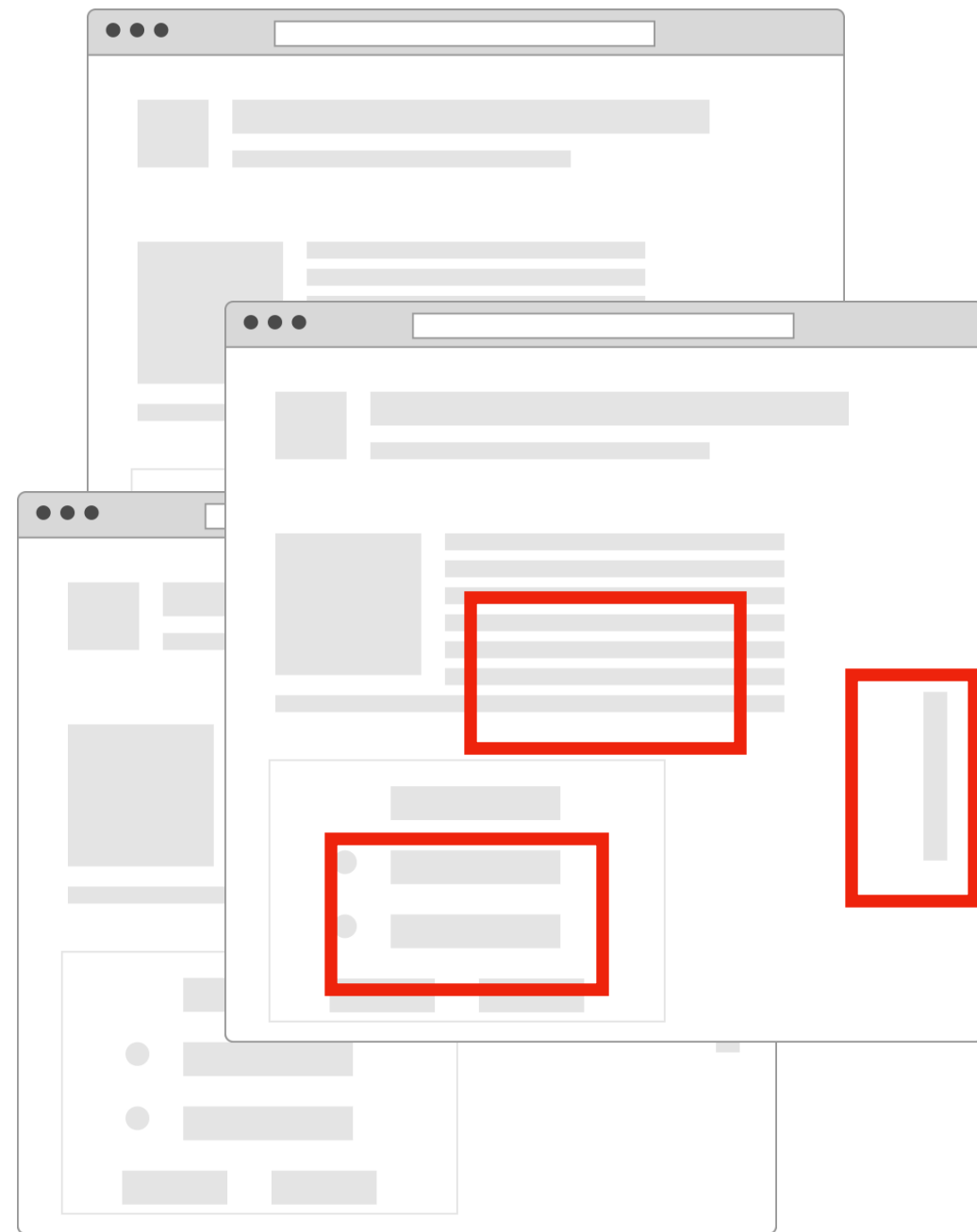
User Tests



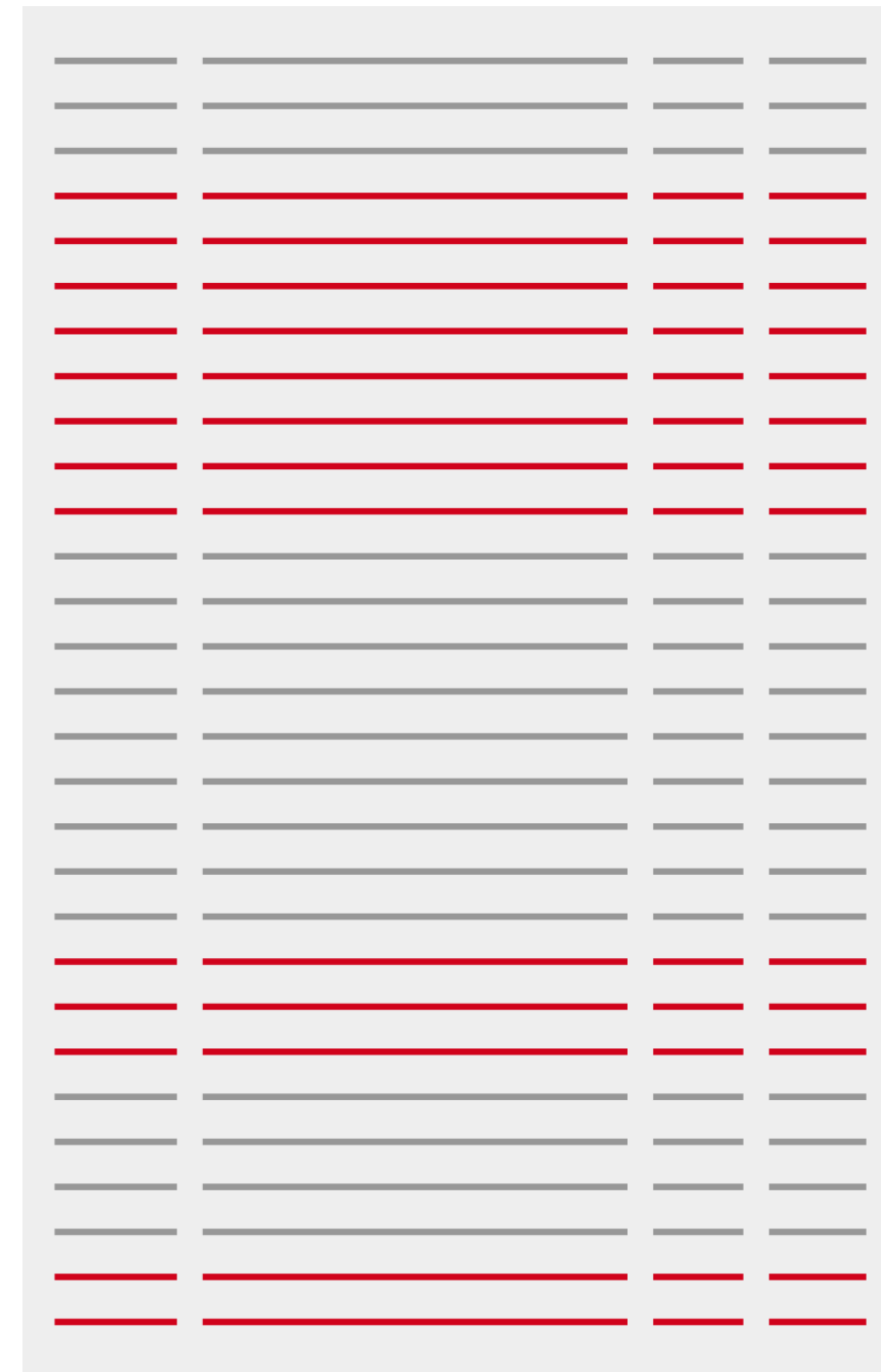
UX Diagnose Automation

Automation

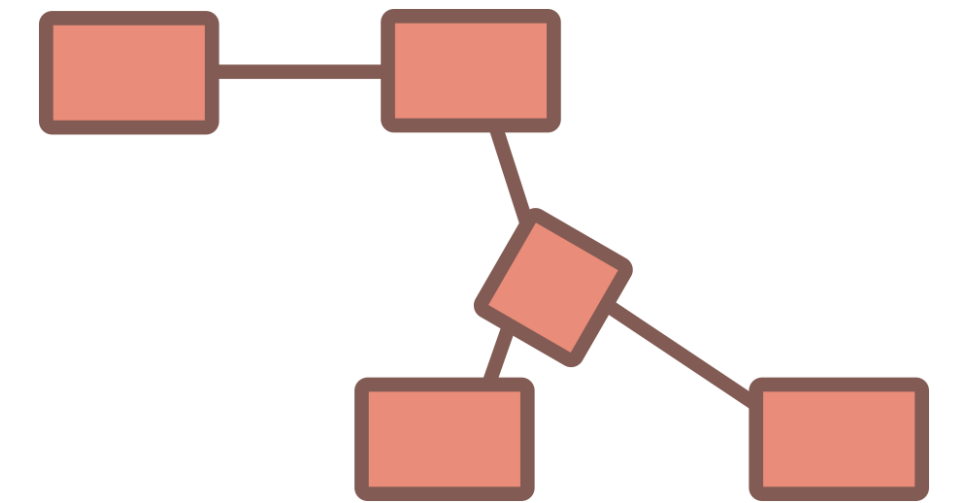
We don't *always* need participants



Static



Interaction-Based

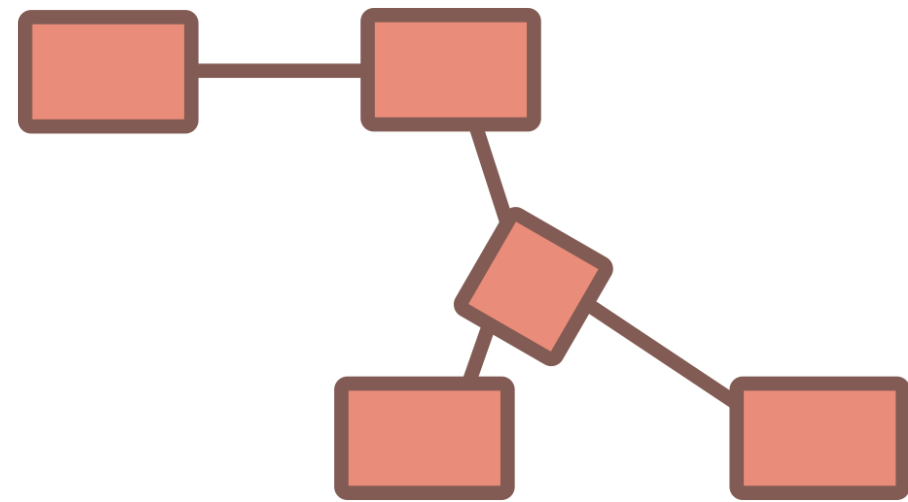


Heuristics



Metrics

Usability Smells



Heuristics

Personal Data

First Name

Last Name

Birthday

Phone Number

14452224503

Address

Zip / Postal Code

Country

Register

Fork me on GitHub

Kobold

Find usability issues on your Web App automatically.

Login

Login

Don't have an account?

Register

[* Usability Smells](#)[✎ Usability Refactorings](#)[⚡ Usability Events](#)[📄 Tracking Code](#)[⚙ Configuration](#)[📄 About](#)

Usability Smells

Found 3 smells.

[🔄 Recalculate](#)

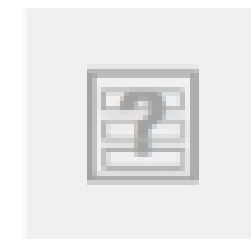


Scarce Search Results ⓘ

<http://shoestore.local/>

`/html/body/div/div[2]/div/div[3]/form`

[🚫 Ignore](#) [🔄 Reset stats](#)



Late Validation (server side) ⓘ

<http://shoestore.local/contact.php>

`/html/body/div[3]/div[2]/div/div/div[2]/form`

[🚫 Ignore](#) [🔄 Reset stats](#)



Unresponsive Element ⓘ

http://shoestore.local/search_products.php

`/html/body/div/div[2]/div/div[2]/img`

[🚫 Ignore](#) [🔄 Reset stats](#)

Developed at [LIFIA](#) - Universidad Nacional de La Plata, Argentina

Powered by [Pharo](#) and [Seaside](#)

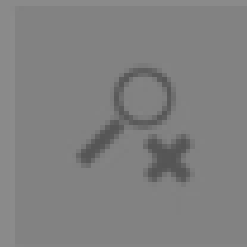
[GitHub](#)

* Usability Smells

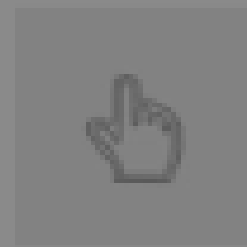
Usability Smells

Usability Smells

Found 3 smells.

[Recalculate](#)

Scarce Search Results

<http://shoestore.local/>[/html/body/div/div\[2\]](#)[Ignore](#) [Reset stats](#)

Unresponsive Element

<http://shoestore.local/search>[/html/body/div/div\[2\]](#)[Ignore](#) [Reset stats](#)

Scarce Search Results

X

At <http://shoestore.local/>This search form didn't bring results **66% of the times** (308 out of 205 search attempts).[Live view](#)

Top 5 unsuccessful search queries

Position	Search query	Times searched
1	bags	101
2	eliot boots	101
3	bootz	3

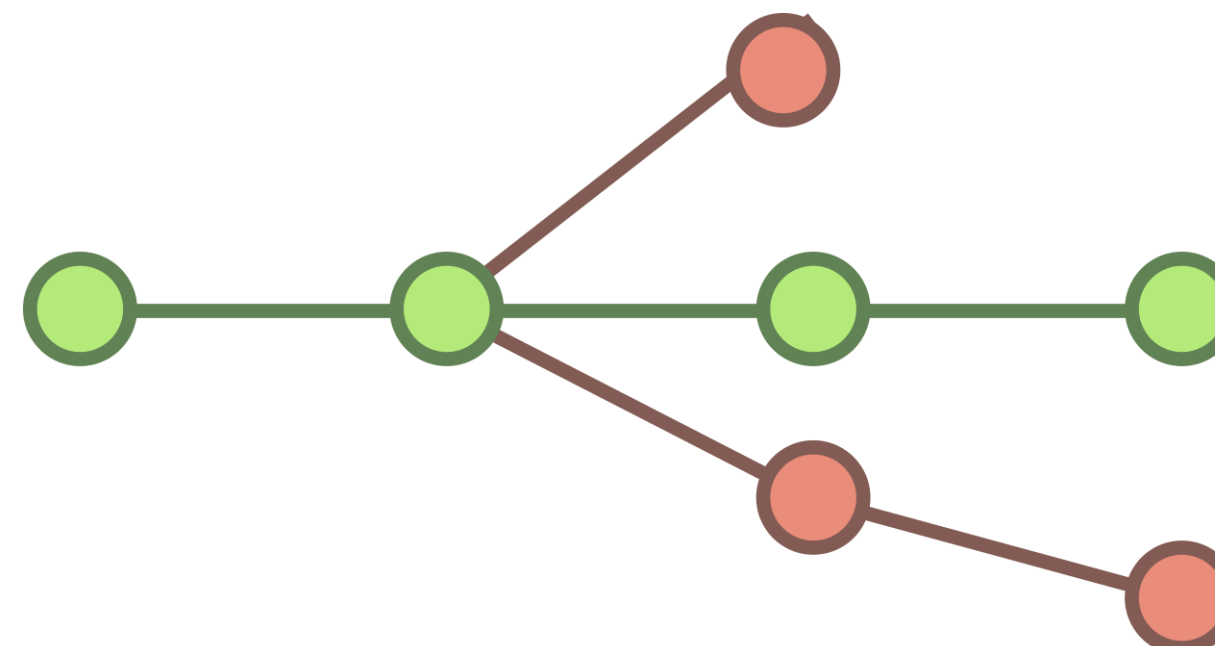
Recommended Refactorings

Add Autocomplete by providing AC features, users will have an overview of what's available before hitting 'Search'. [Learn more...](#)

Close

Metric-based Analysis

Automated UX issues detection



Metric-based Analysis

Cognitive Effort Metric



Personal Data

First Name

James Marshall

Last Name

Hendrix

Birthday

10/27/1942

Phone Number

Address

Zip / Postal Code

Country

Register

Personal Data

First Name

James Marshall

Last Name

Hendrix

Birthday

27 11 1942

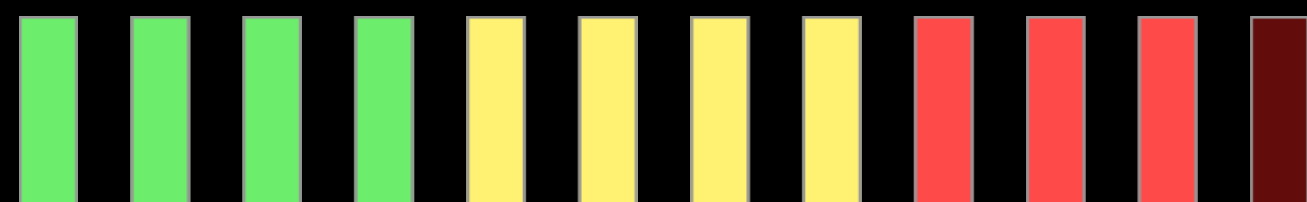
Phone Number

Address

Zip / Postal Code

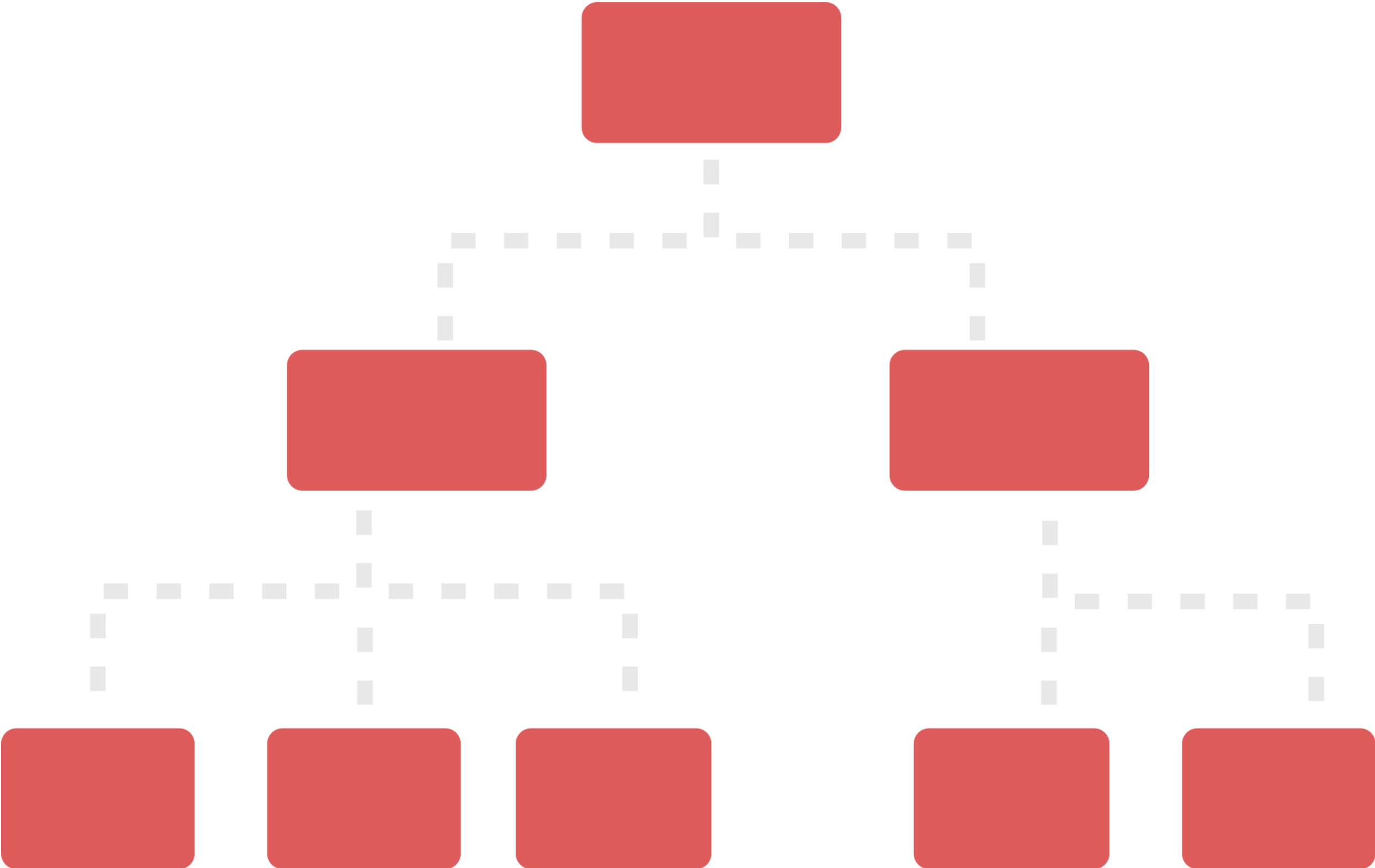
Country

Register



<< USER EFFORT >>





Evaluating UX

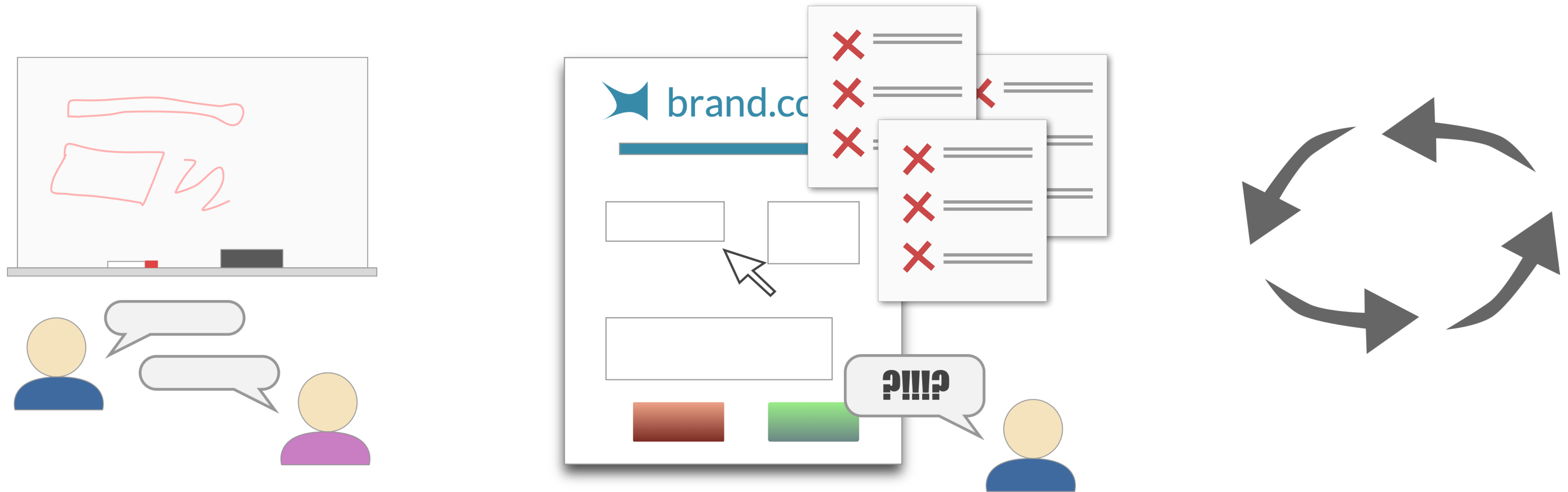


Improving UX



UX Problems

Dealing with test results



UX Problems

Dealing with test results

Personal Data

First Name

Last Name

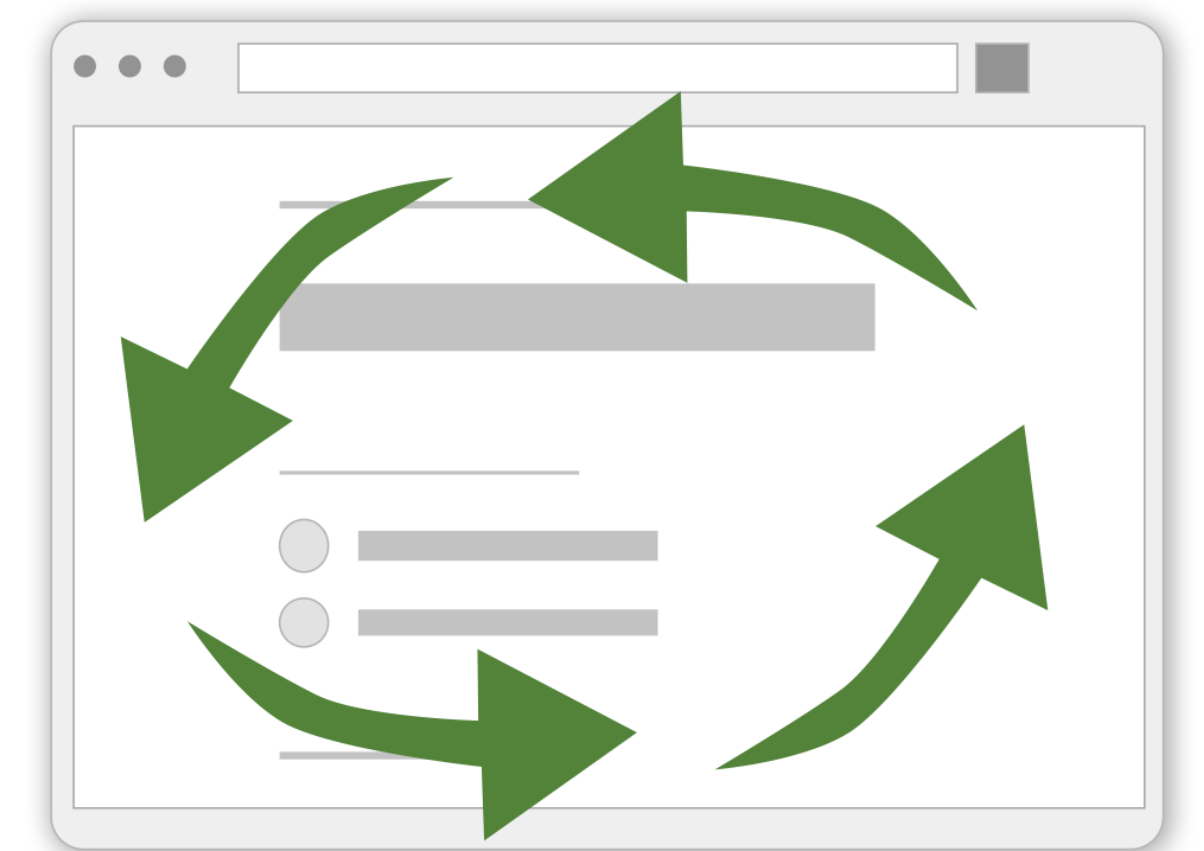
Birthday
Birth Day Day Month Year

Phone Number
14452224503

Address

Country
Zip / ar |
Argentina
Armenia
Aruba

Register



Lean / Agile UX

* Usability Smells

Usabi

Usability Smells

Found 5 smells.

Recalculate



No Validation

http://clientes.belugas.com

//*[@id="register"]

Ignore Reset stats



Free Input for Limited V

http://clientes.belugas.com

//*[@id="country"]

Ignore Reset stats



Undescriptive Element

http://clientes.belugas.com

/html/body/div/div[2]/

Ignore Reset stats

At http://clientes.belugas.com.ar/testsite/register.php

On element `//*[@id="register"]`Failed 97% of the submissions, over a total of 103 with **none** validation

Live view

Birthday

dd/mm/yyyy

Country

Country

City

City

Email address

Enter email

Repeat email address

Enter email

Password

Password

Recommended Refactorings

Add Form Validation provide validation on the client, right after each field, so users don't have to hit "Submit" to find out there's a missing or invalid datum.

Apply now

Usability/UX Refactoring

Personal Data

First Name

Last Name

Birthday

Day ↓ Month ↓ Year ↓

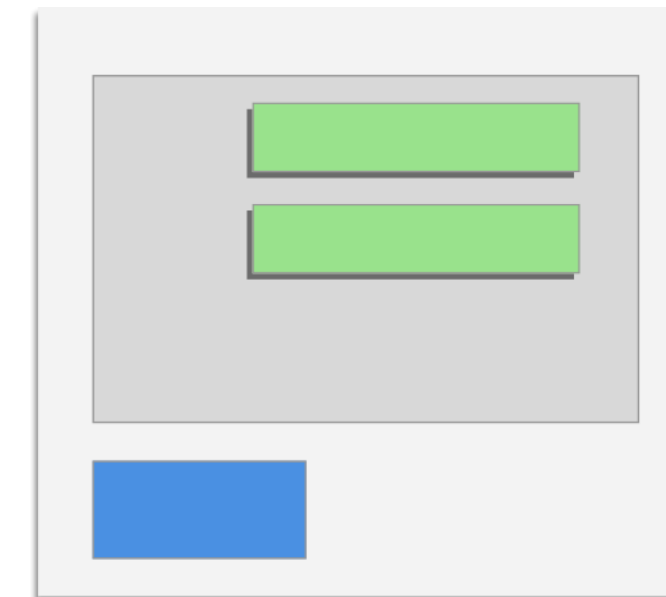
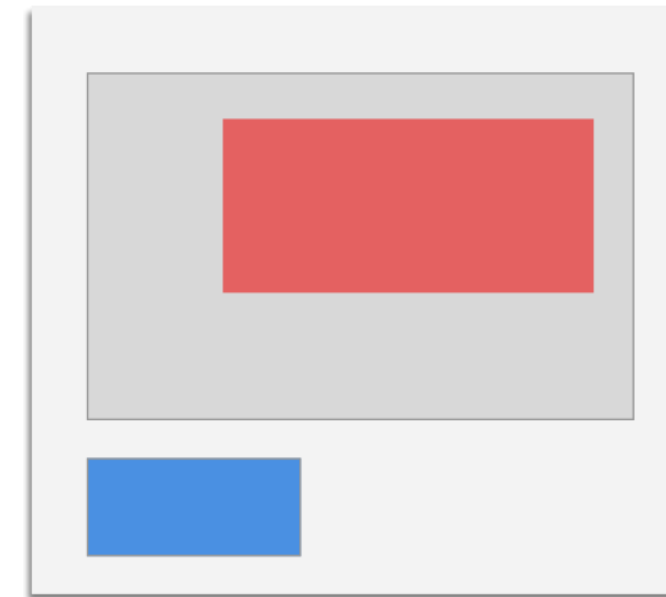
Phone Number
14452224503

Address

Zip / Postal Code

Country

Register



Usability Smells

Found 8 smells.

Recalculate Reveal 1 Ignored

No Validation
[http://clientes.belugas.com.ar/clientsite/register.php](#)
 /**[@id="register"]
 Ignore Reset stats

Late Validation (server)
[http://clientes.belugas.com.ar/clientsite/register.php](#)
 /**[@id="tinyform"]
 Ignore Reset stats

Free Input for Limited Values
[http://clientes.belugas.com.ar/clientsite/register.php](#)
 /**[@id="country"]
 Ignore Reset stats

Undescriptive Element
[http://clientes.belugas.com.ar/clientsite/register.php](#)
 /html/body/div/div[2]
 Ignore Reset stats

Free Input for Limited Values

At <http://clientes.belugas.com.ar/clientsite/register.php>

On element `/**[@id="country"]`

Live view

Country

City

Recurrent values



Value	Times entered
Argentina	101
Brasil	201
Other	0
Uruguay	103
Total	405

Recommended Refactorings

Add Autocomplete by providing Autocompletion with the most popular values, users won't have to type the full values.

Apply now

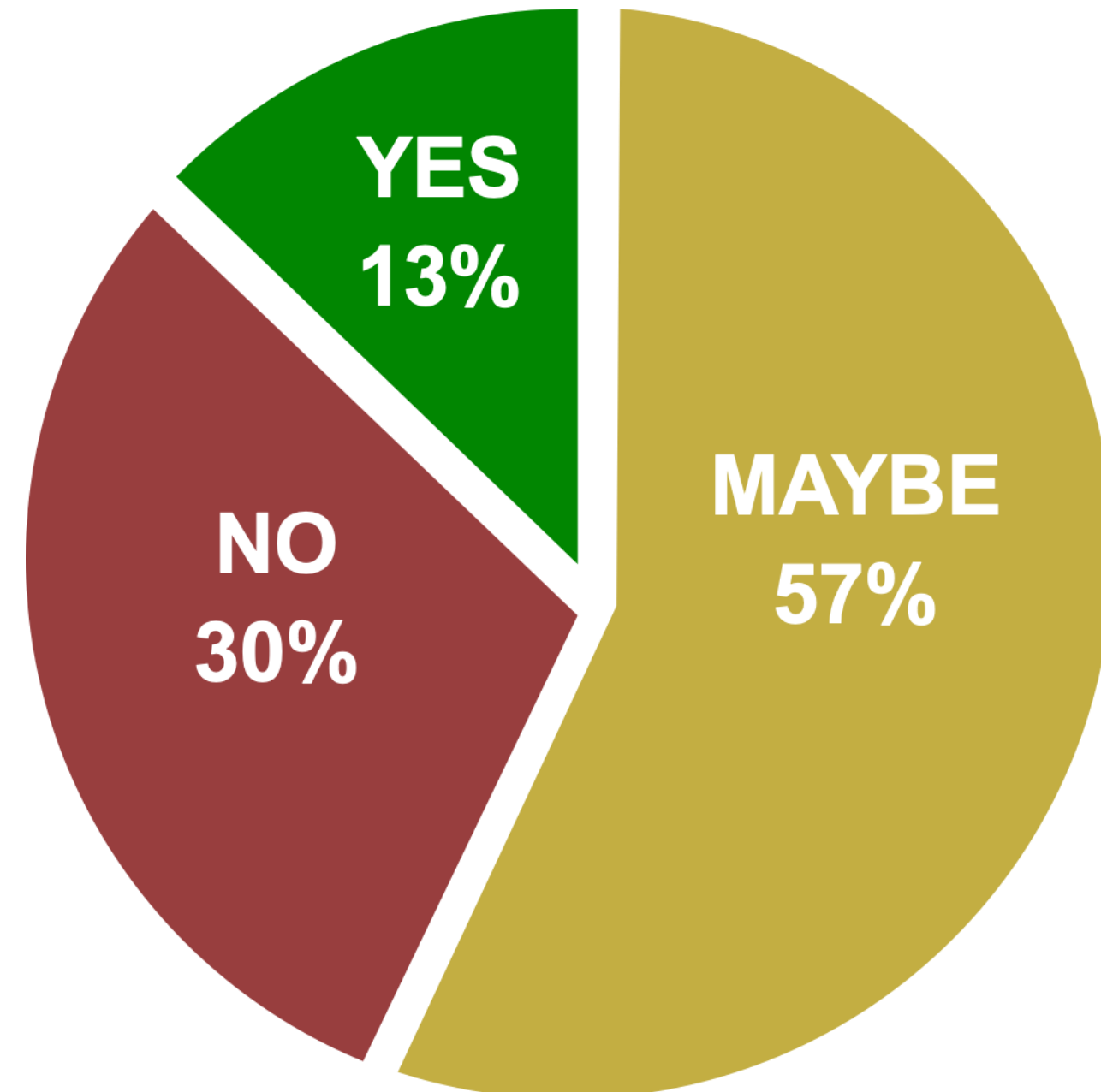
InputIntoRadios Turn a regular text field into a set of radio buttons with an "other" option.

Apply now

UX Refactoring Adoption

Survey

“Would you allow an external service to alter your application’s UI?”



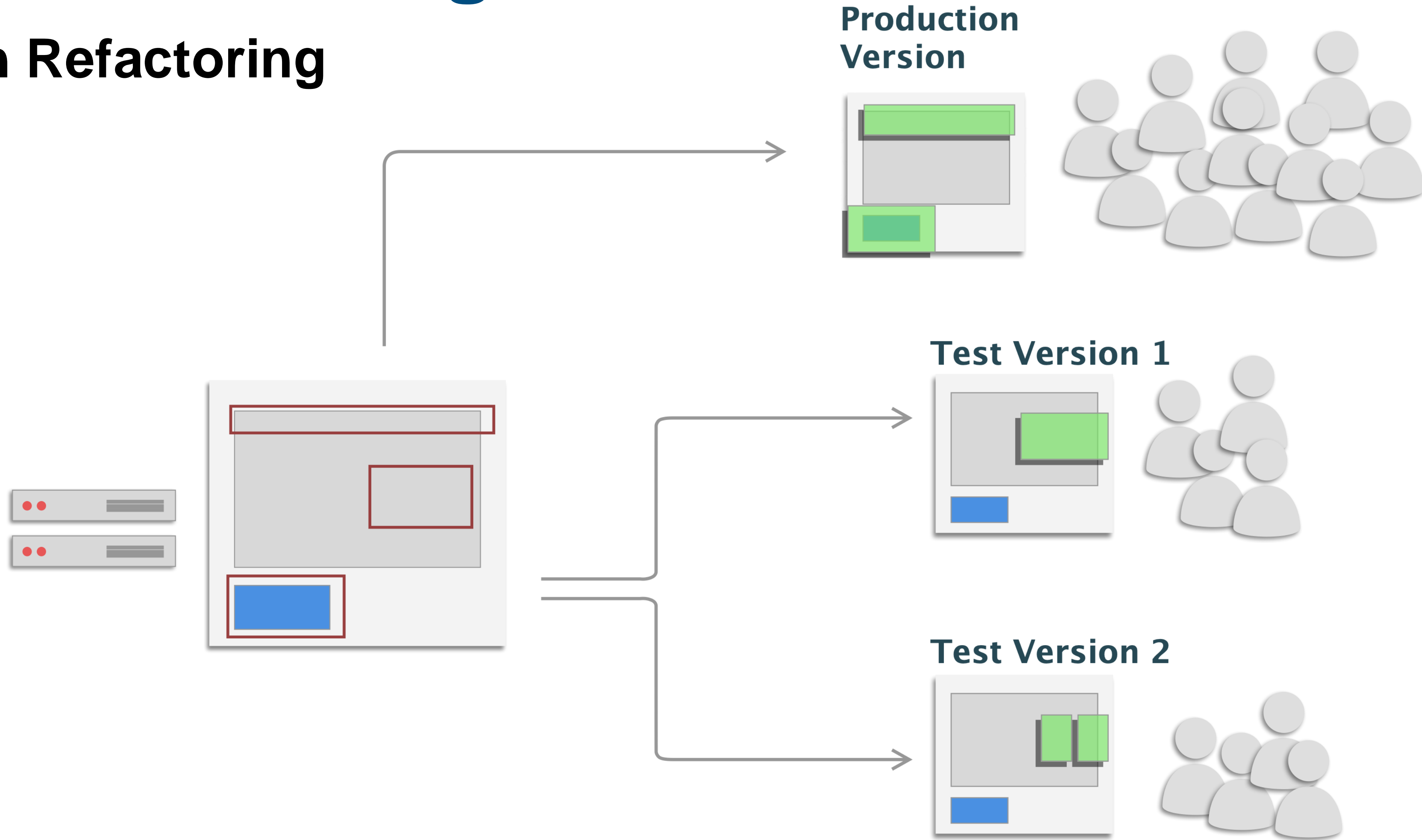
I must test before deploy (65%)

I must ask the client first (47%)

Trust issues (47%)

Live Versioning

With Refactoring



UX Painter

End User Programming

My Account

Buscar en Google o escribir una URL

Personal Information

Name

Gender

Birthdate

Address

City

Country

Select

UPDATE

My Account

Buscar en Google o escribir una URL

Personal Information

Name

Gender

Male

Female

Other

Birthdate

June 2021

Su	Mo	Tu	We	Th	Fr	Sa
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30			

Country

Select

UPDATE

My Account

Buscar en Google o escribir una URL

Personal Information

Name

Gender

Male

Birthdate

Día Mes Año

Address

City

Country

arg

Argentina

UPDATE

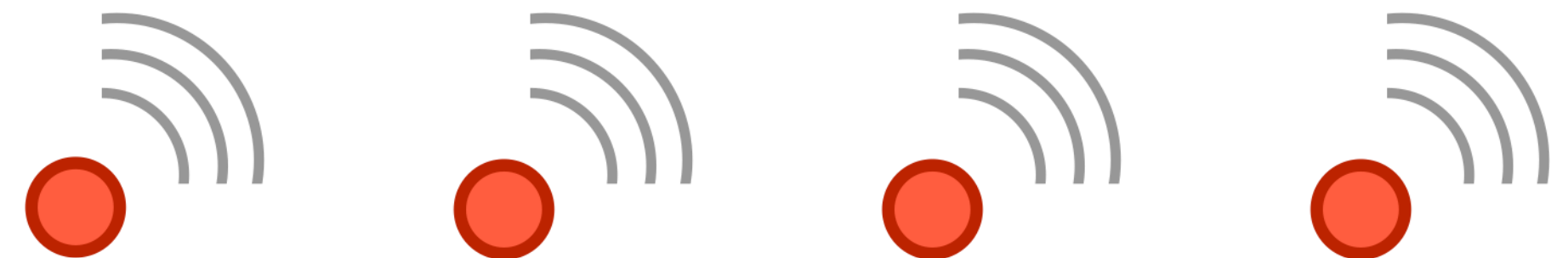
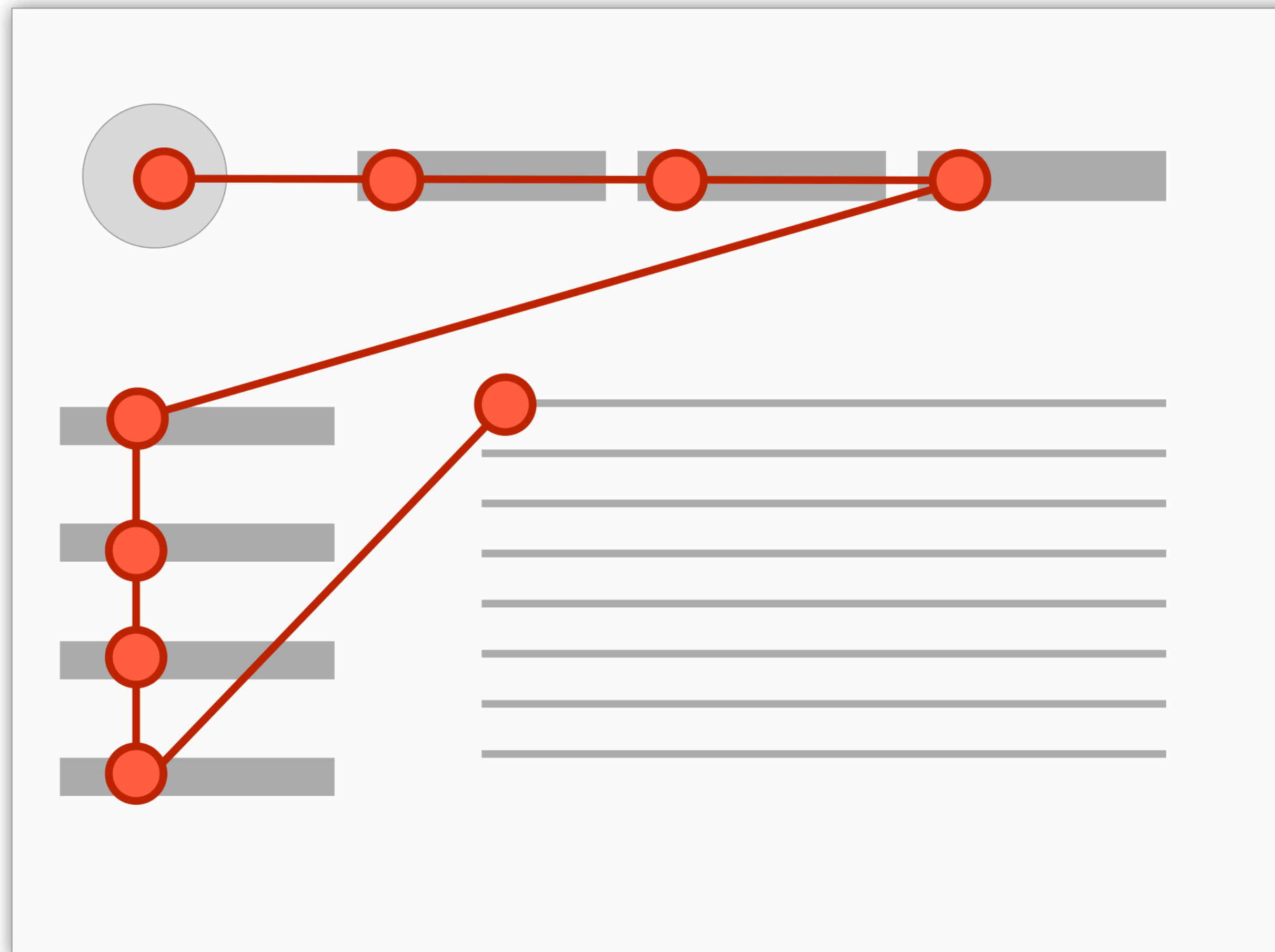
Refactoring for Accessibility



Screen Readers

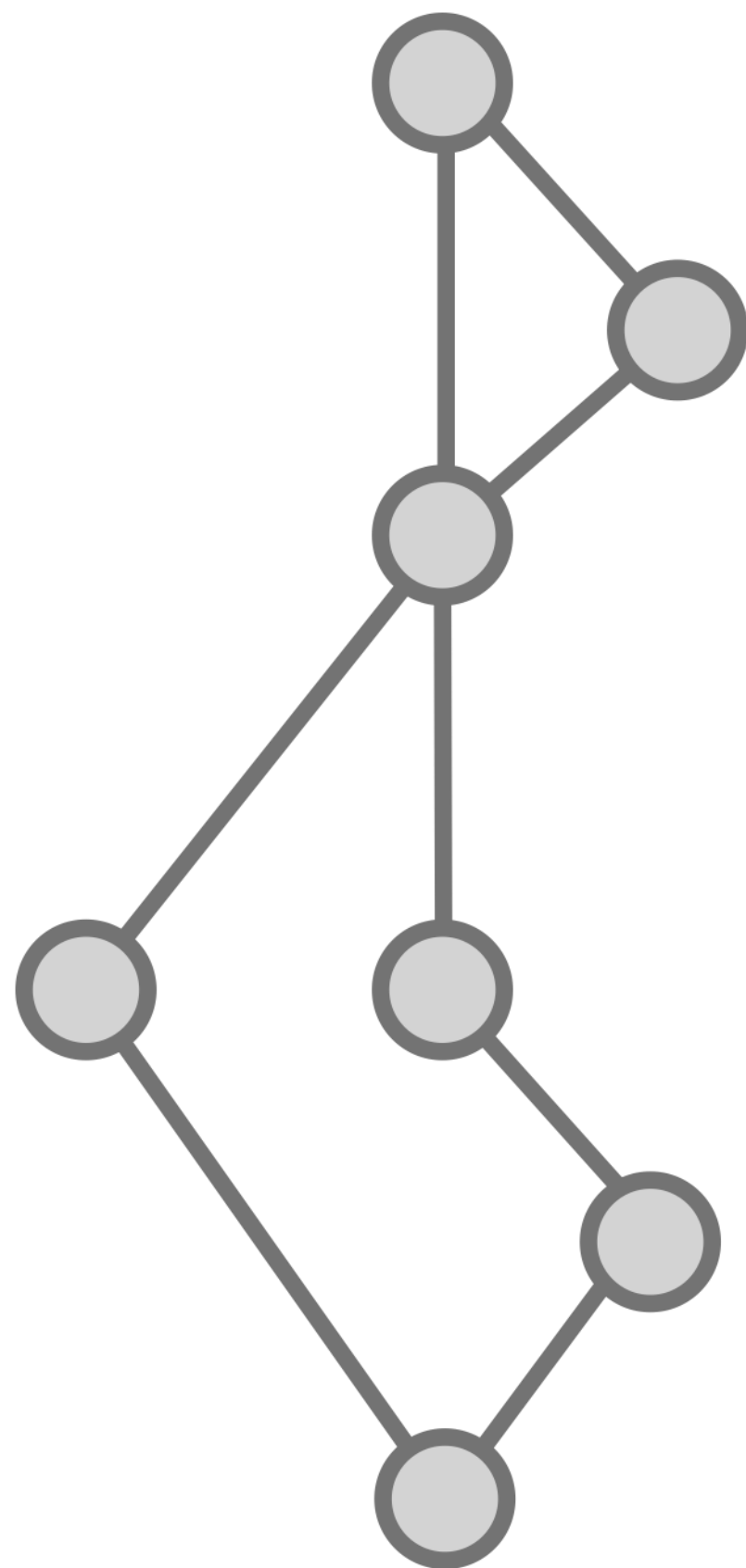
Screen Readers

Linear Workflow

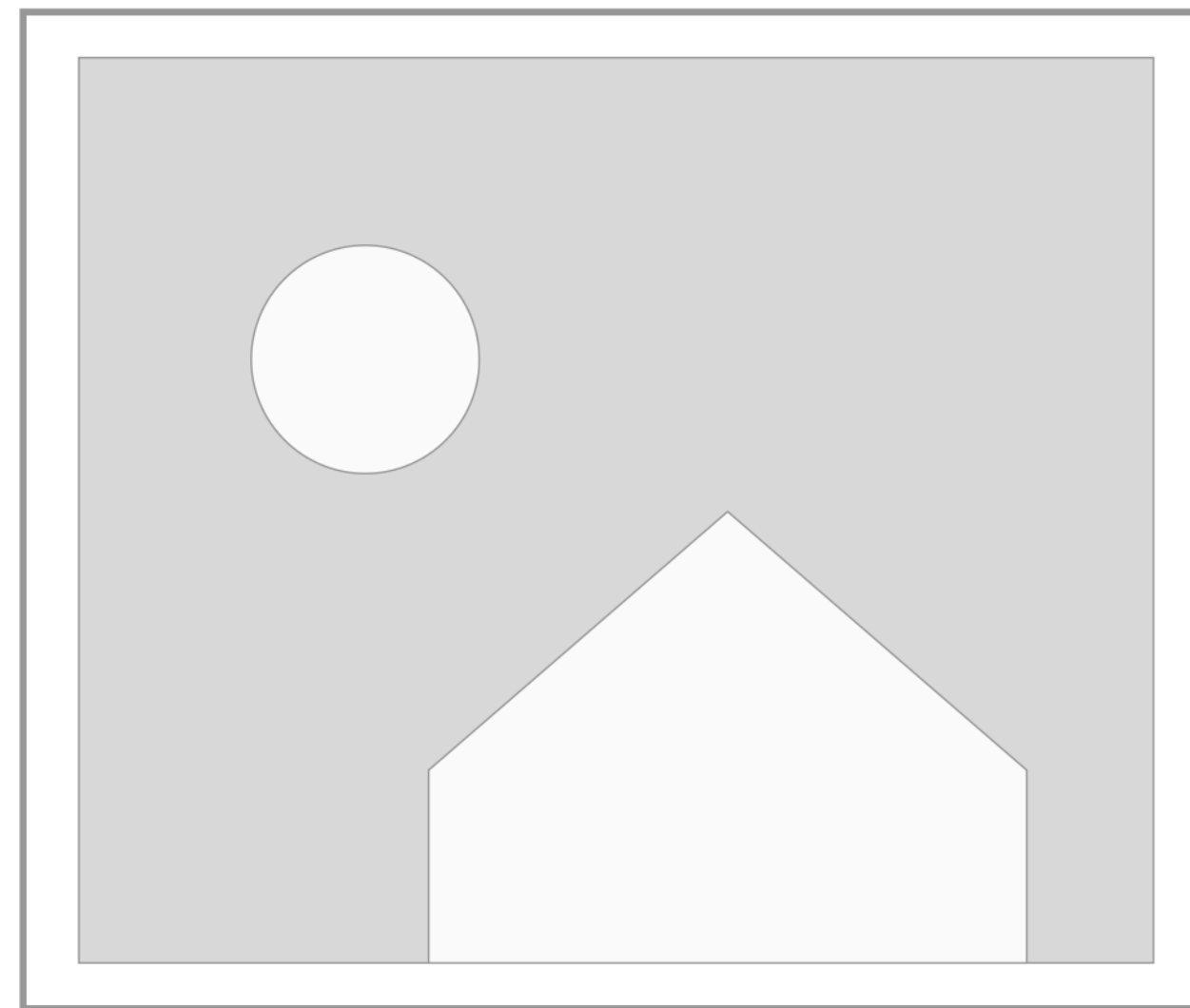


Screen Readers

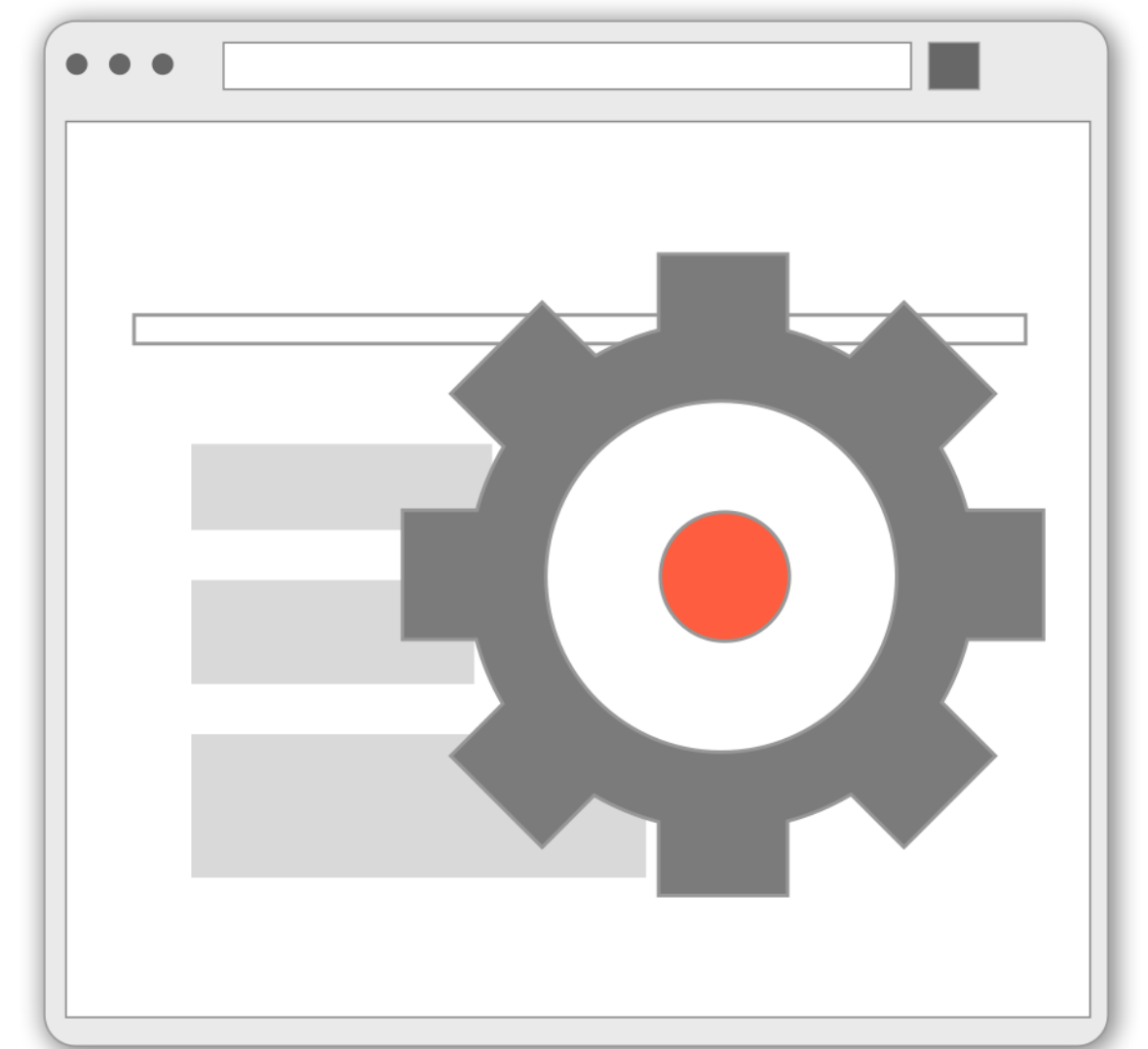
Linear Workflow



Complex Paths



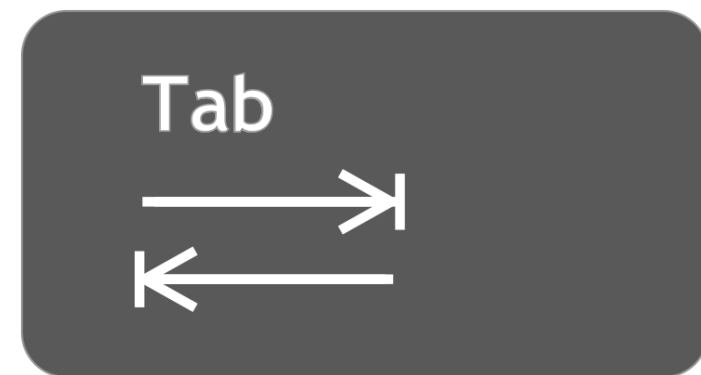
Metadata Issues



RIA / SPA Interaction

Accessibility Smells

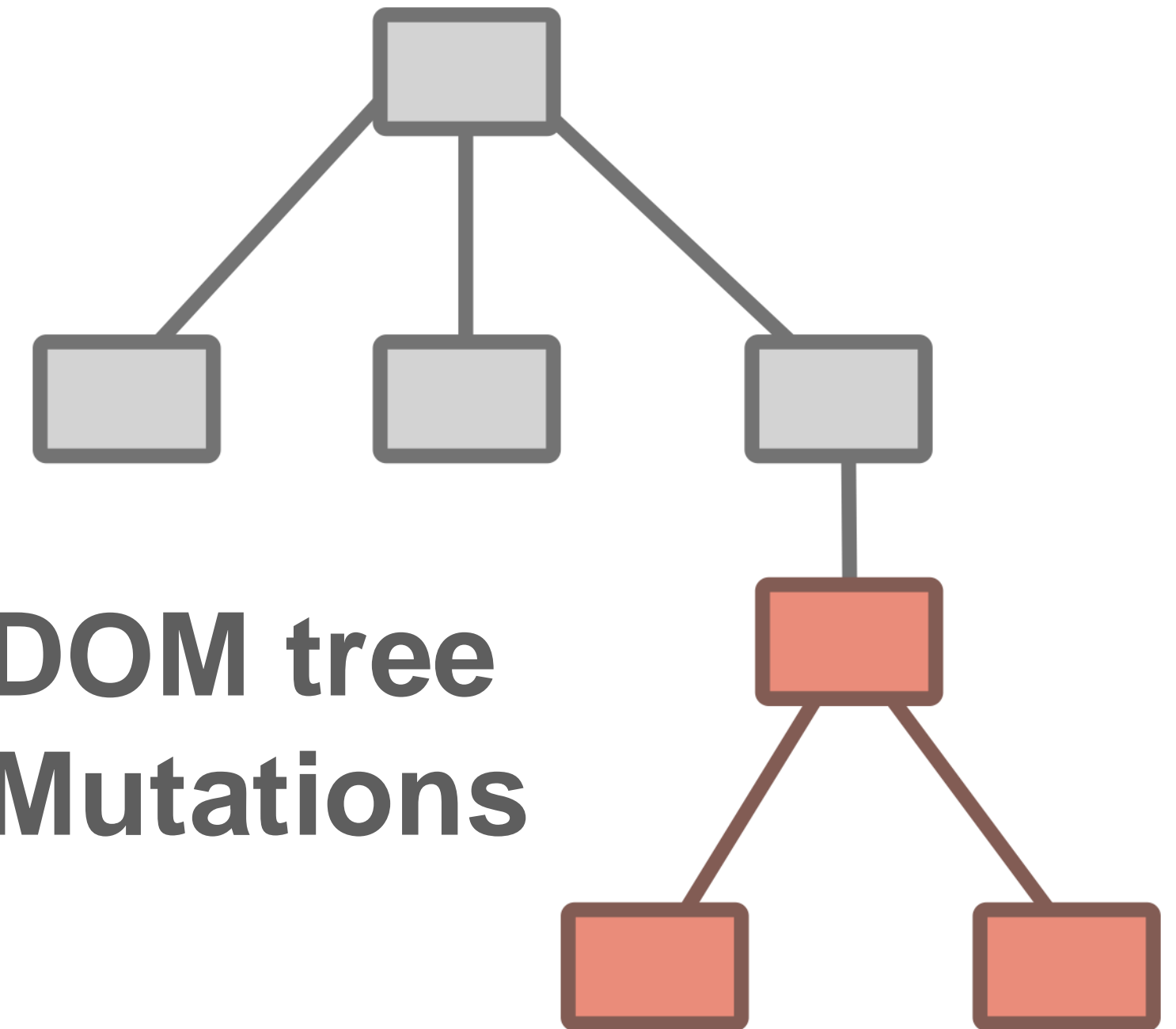
For Screen Readers



**SR-specific
interaction**



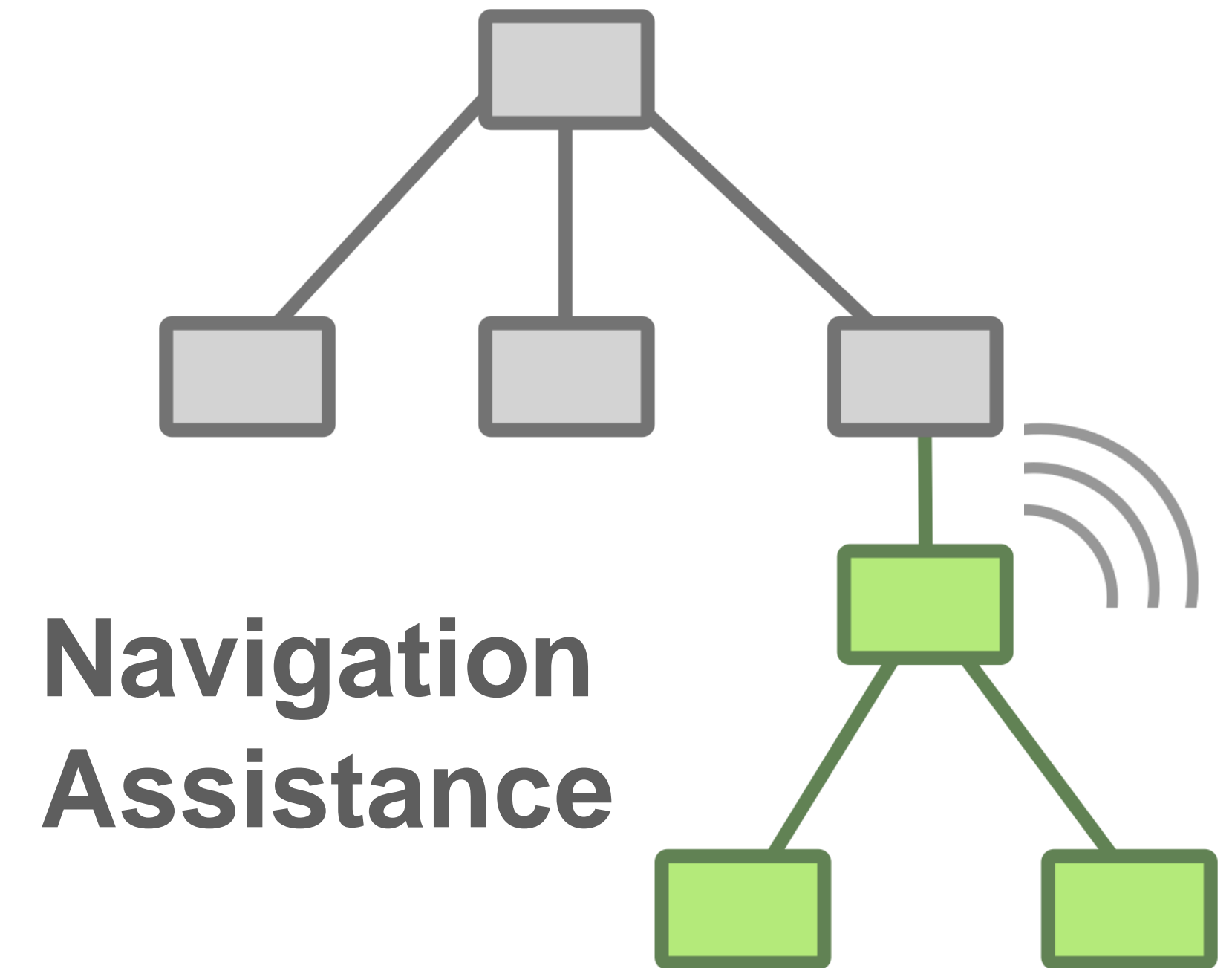
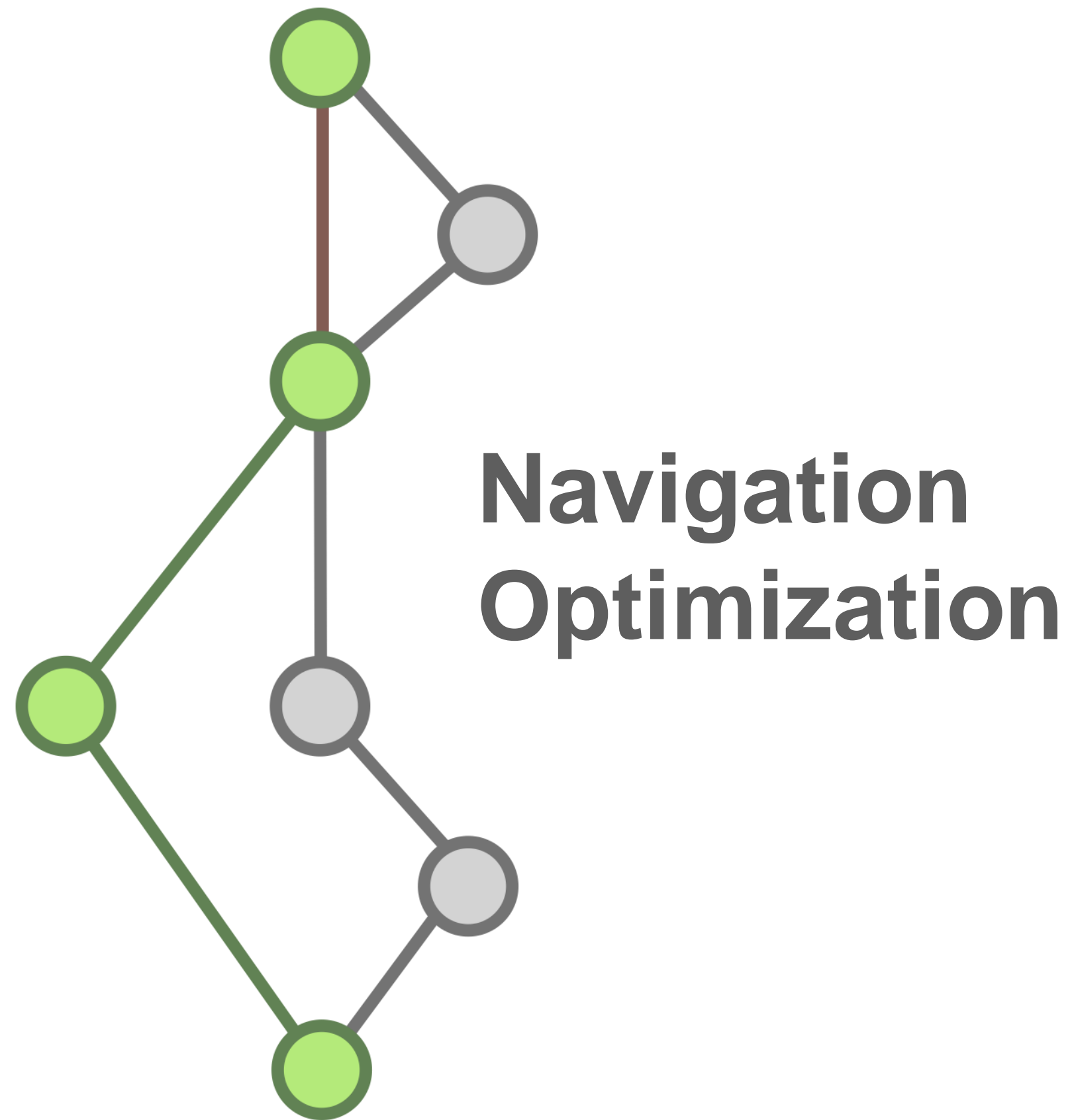
**Erratic
Navigation**



**DOM tree
Mutations**

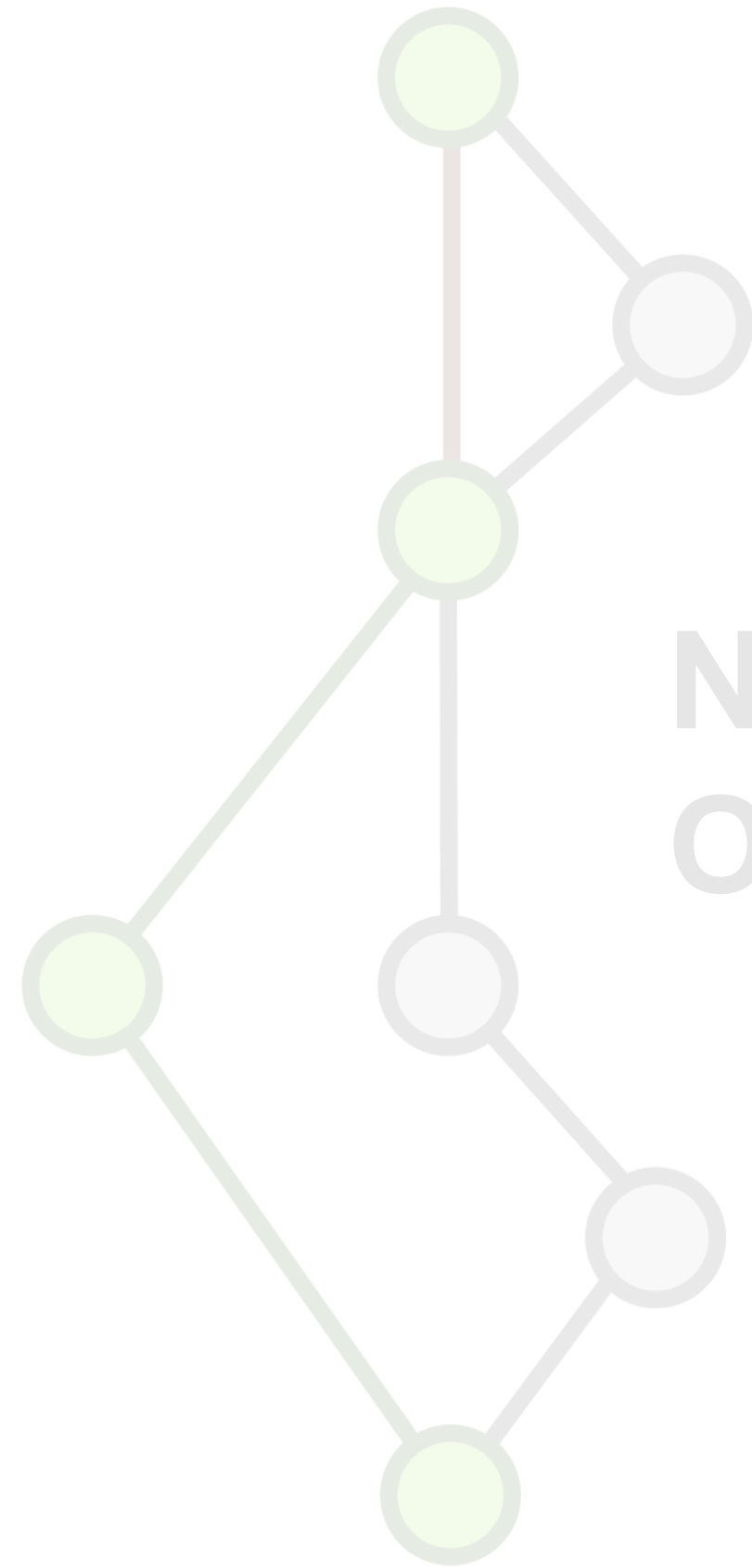
Refactoring for Accessibility

Screen Reader Flows



Refactoring for Accessibility

Screen Reader Flows



Navigatio
Optimiza

Feature: Evolving Web Applications



Personalized Web Accessibility using Client-Side Refactoring

According to W3C accessibility standards, most Web applications are neither accessible nor usable for people with disabilities. Developers often solve this problem by building separate accessible applications, but these are seldom usable and typically offer less functionality than the original. Another common solution is to maintain a single application, but create an accessible view by applying on-the-fly transformations to each requested page — a solution that rarely suits all audiences. A third solution is described here: let users improve Web accessibility in their client browsers through interface refactorings, which offer many customized, accessible views of a single application.

Alejandra Garrido,
Sergio Firmenich,
Gustavo Rossi,
and Julián Grigera
Universidad Nacional
de La Plata, Argentina

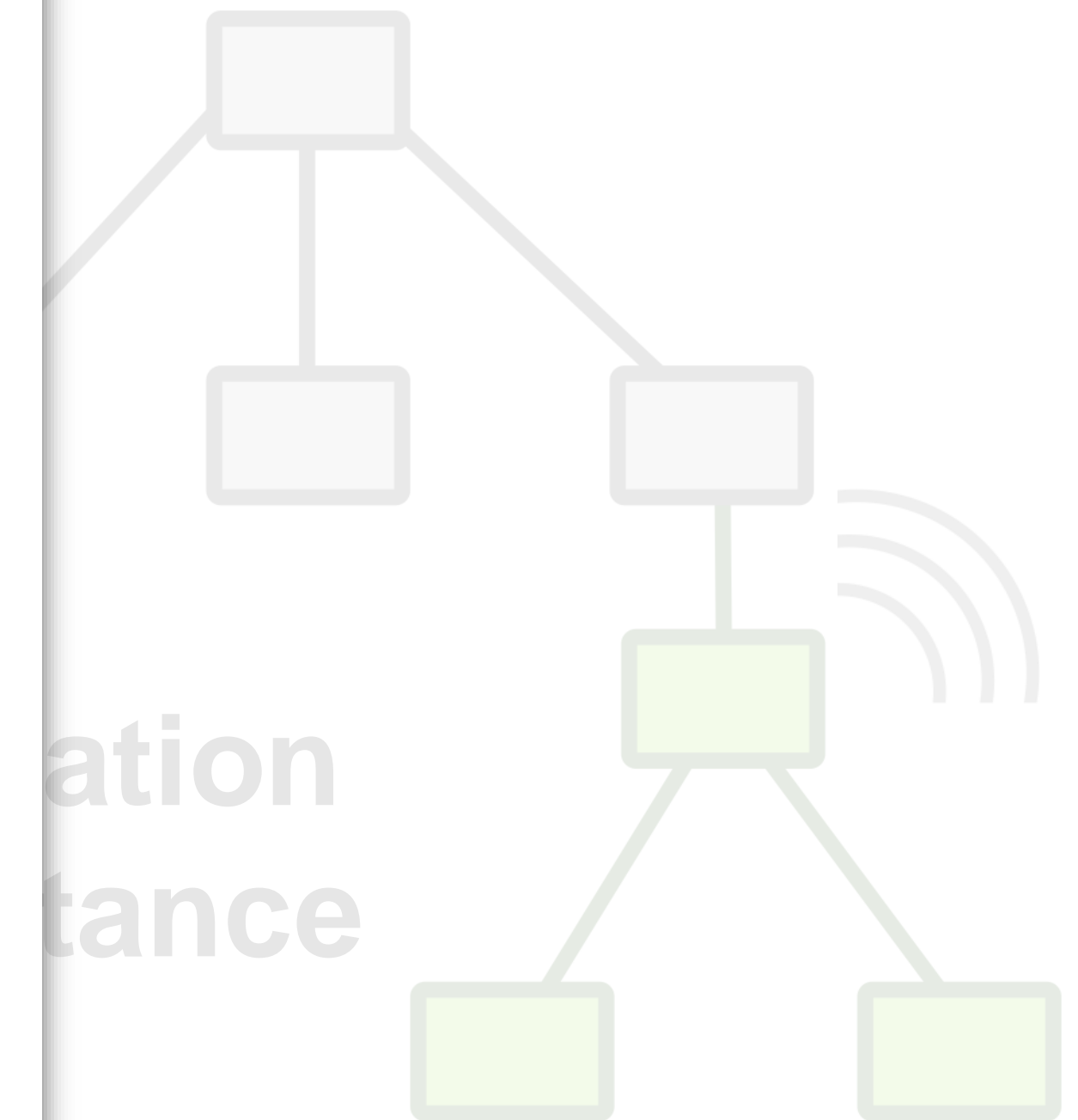
Nuria Medina-Medina
Universidad de Granada, Spain

Ivana Harari
Universidad Nacional

Refactoring was originally conceived as a technique to improve software's internal qualities — such as understandability and maintainability — while preserving semantics.¹ In prior work, we adapted the refactoring approach to improve a Web application's external attributes, such as usability.² These Web refactorings consist of small navigation or interface transformations that enhance perceivable concepts of Web applications, such

different contexts, “one for all” is barely feasible.

When applying refactoring to improve internal qualities, developers decide which transformations to apply and where, because they're the ones benefiting from the improvement. As Brian Foote and Joseph Yoder put it, “Who better to resolve the forces impinging upon each design issue as it arises, as the person who is going to have to live with these decisions?”³ Moreover, differ-



ation
tance