

**Incrementally shifting
toward a modular
experience**

How is our approach changing?

We're transforming...

FROM

Big contract

Single vendor

Big releases: R1 & R2

Existing R2 plans

Extensive, rigid requirements specifying detailed solutions

Big design up front

TO

Smaller, modular contracts

Multiple vendors/dev teams

Frequent releases

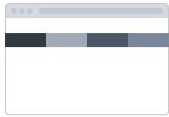
New roadmap built around prioritized user needs

Flexible requirements specifying desired outcomes

Light global design prescription; greater design freedom within workflows

How?

A



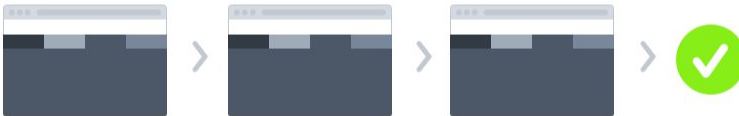
B



C



D



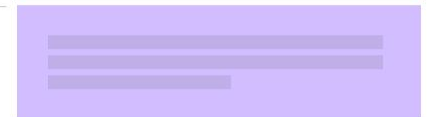
E



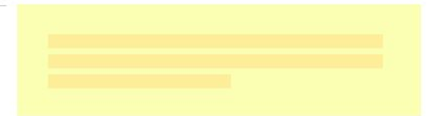
1



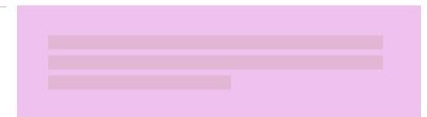
2



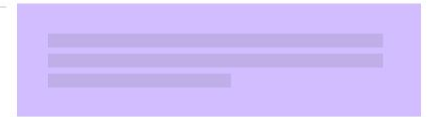
3



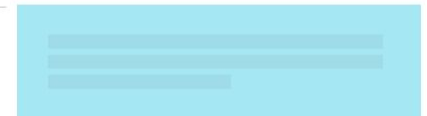
4



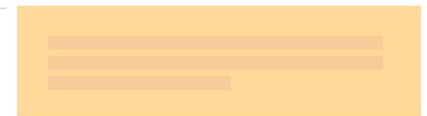
5



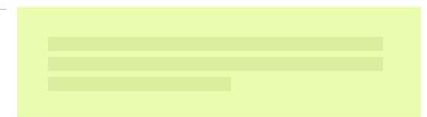
6



7



8

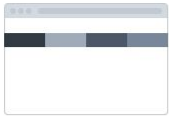


9



Here is an example legacy system.
We're going to transform it incrementally.

A



B



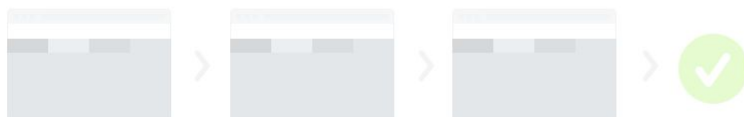
C



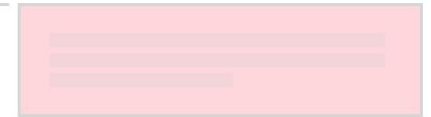
D



E



1



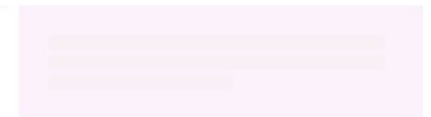
2



3



4



5



6



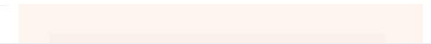
7



8



9



We've got a **homepage** or start screen.

A



B



C



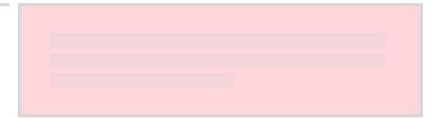
D



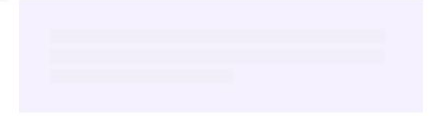
E



1



2



3



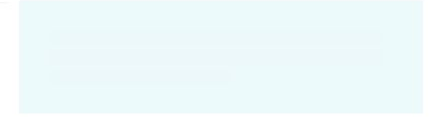
4



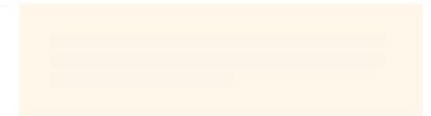
5



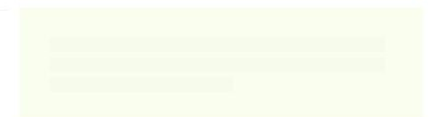
6



7



8



9



We've got four **tasks**, or complete series of steps that a user needs to take to accomplish a piece of work.

D

Person search

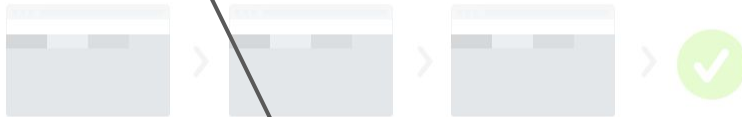
User wants to find a specific person in order to work their case.



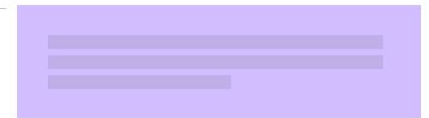
D



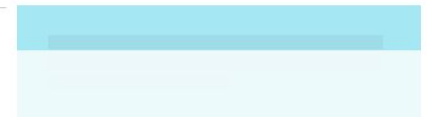
E



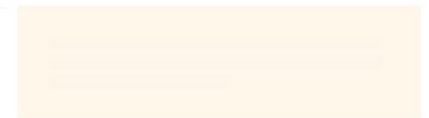
5



6



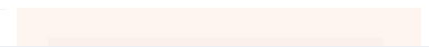
7



8



9



For example, **searching for a person** could be a task, where a user looks for a specific person to work their case.

A



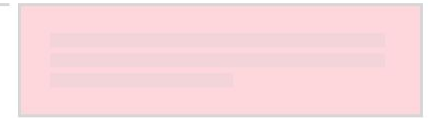
B



C



1



2



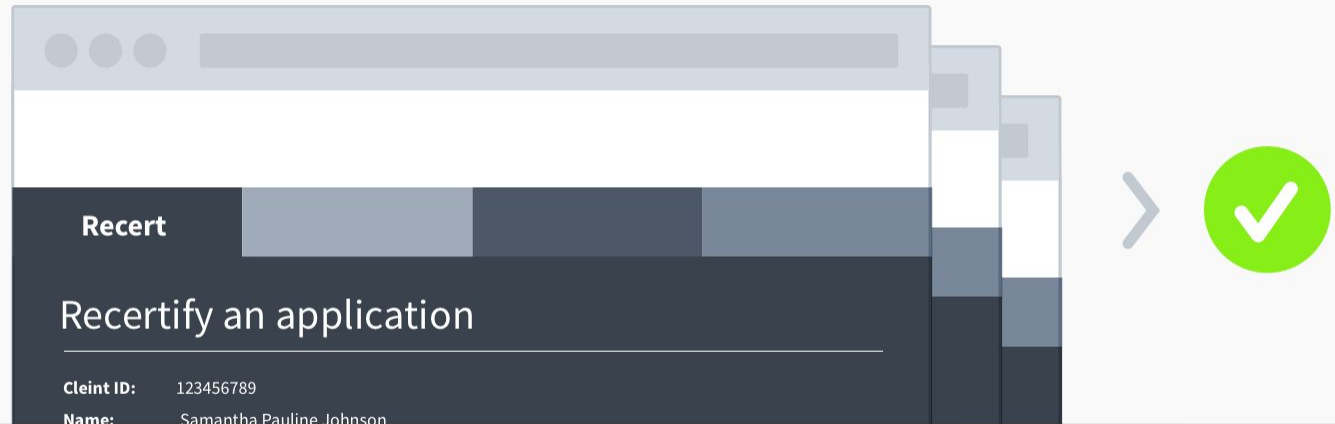
3



4

**B****Recertify an application**

User wants to review current data and make a determination on recertification.



Or **recertifying eligibility** could be a task. Just remember that it's a full journey from start to finish to accomplish a piece of work.

A



B



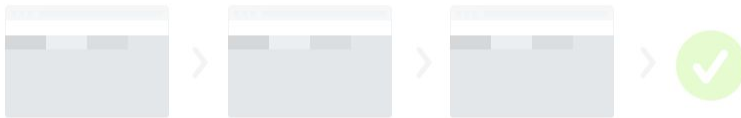
C



D



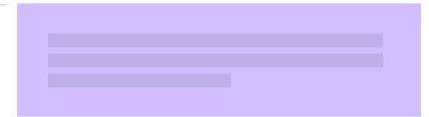
E



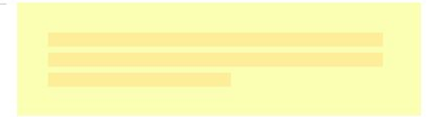
1



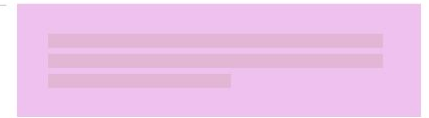
2



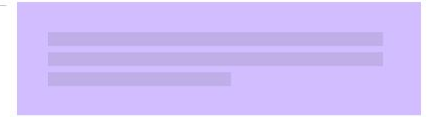
3



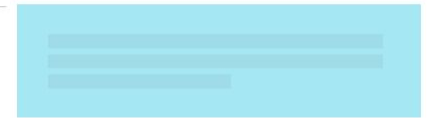
4



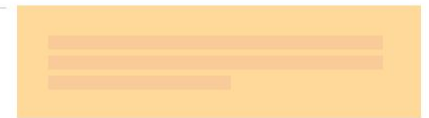
5



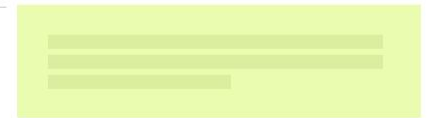
6



7



8



9



We have prioritized work, but that can always change as we continuously prioritize.

A



B



C



D



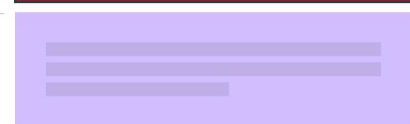
E



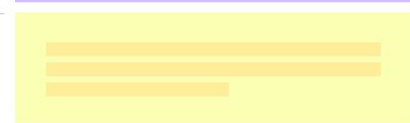
1



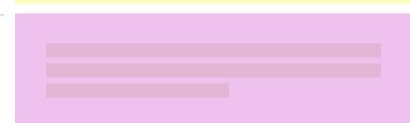
2



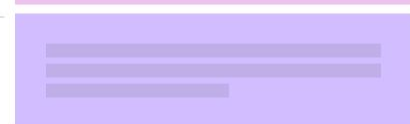
3



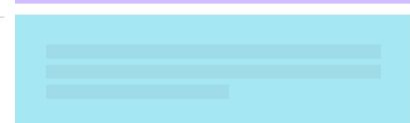
4



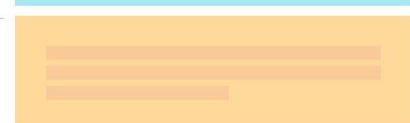
5



6



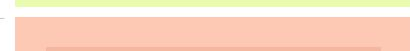
7



8



9



This red box represents a new task/module/procurement and is in the top priority slot.
We'll work on that first.

A



B



C



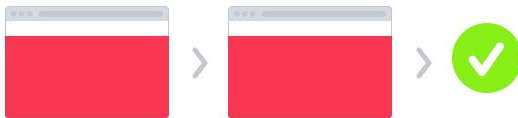
D



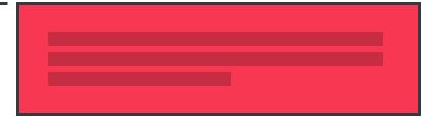
E



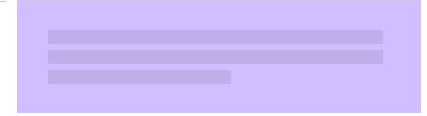
E2



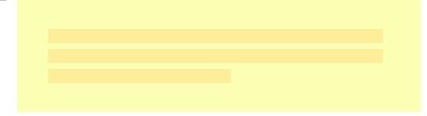
1



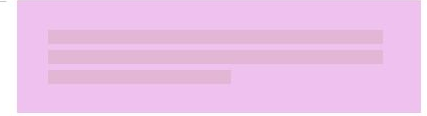
2



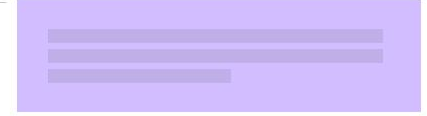
3



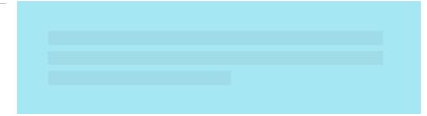
4



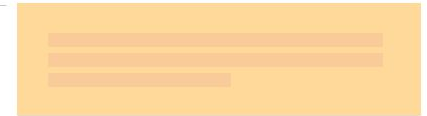
5



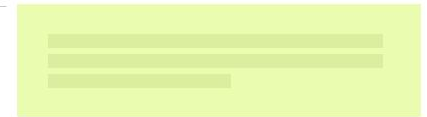
6



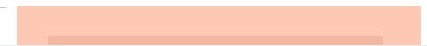
7



8

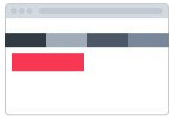


9



We introduce the new module to our system. We'll call it E2. It's an alternate experience that focused on the same task as the legacy E task. We launch it alongside the existing workflow.

A



B



C



D



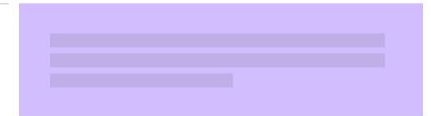
E2



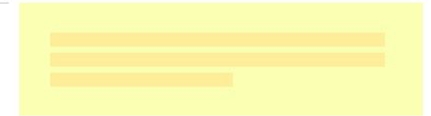
1



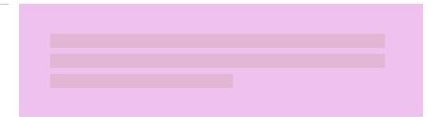
2



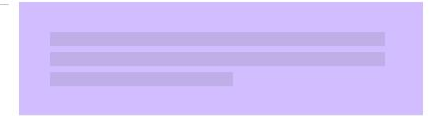
3



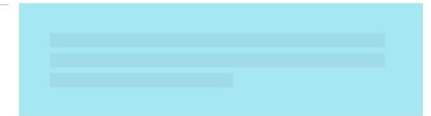
4



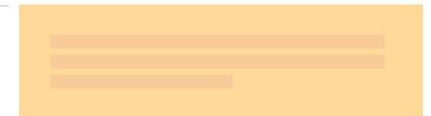
5



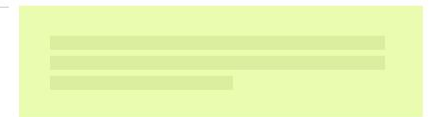
6



7



8



9



Once E2 has been adopted, the legacy E experience can be hidden.



With E2 behind us, we have a new top priority.
We introduce the D2 module, a new experience for the legacy D task.

A



B



C



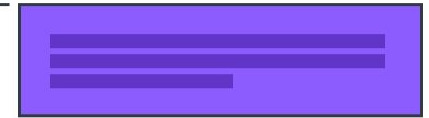
D2



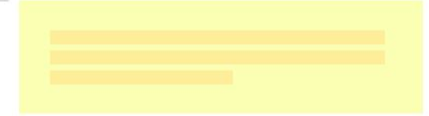
E2



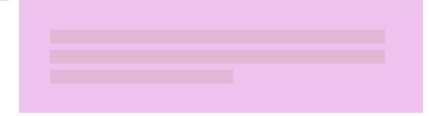
1



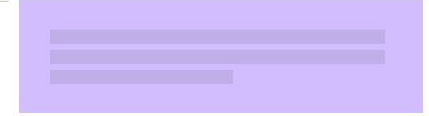
2



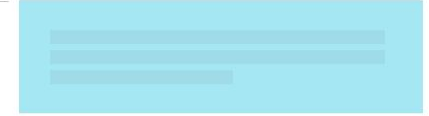
3



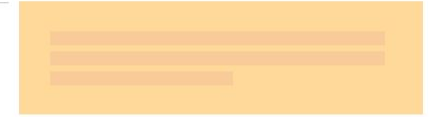
4



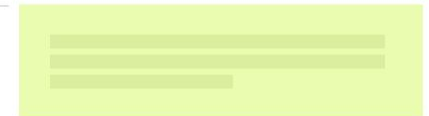
5



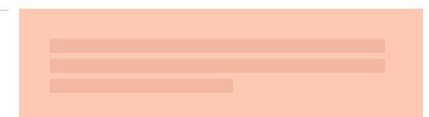
6



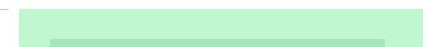
7



8

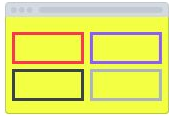


9



With adoption of D2, we hide the legacy D experience.
Note how we're shifting responsibilities from the old system to
the newly developed code.

A2



B



C



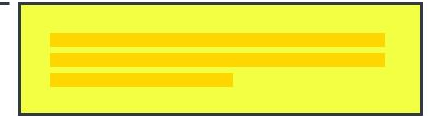
D2



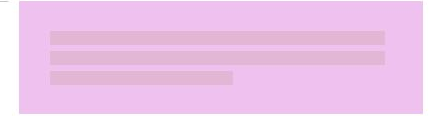
E2



1



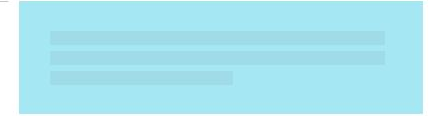
2



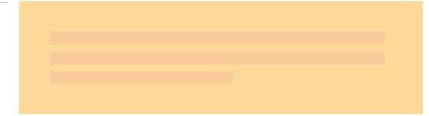
3



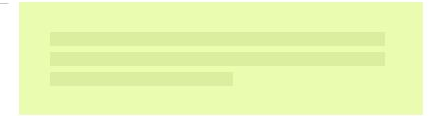
4



5



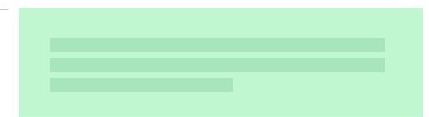
6



7



8



9



Our new top priority is focused on a new homepage, so we focus there.
The new homepage module provides navigation to both legacy and new experiences.

A2



B



C



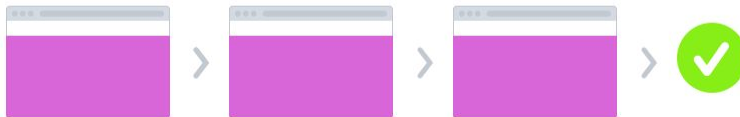
D2



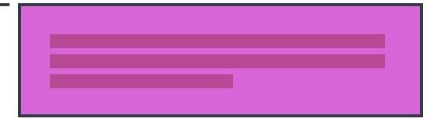
E2



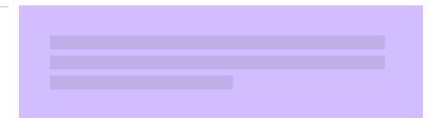
F



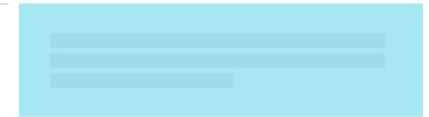
1



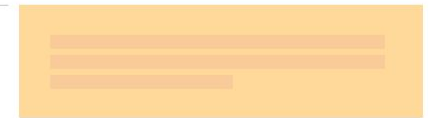
2



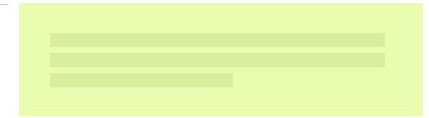
3



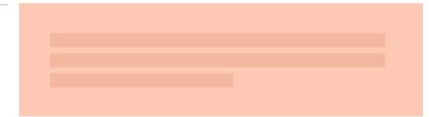
4



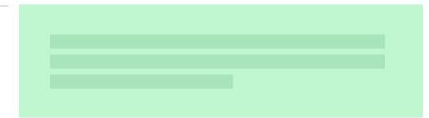
5



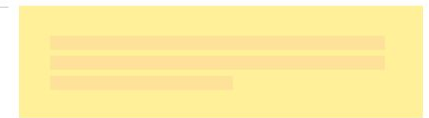
6



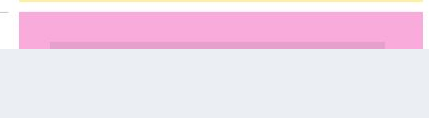
7



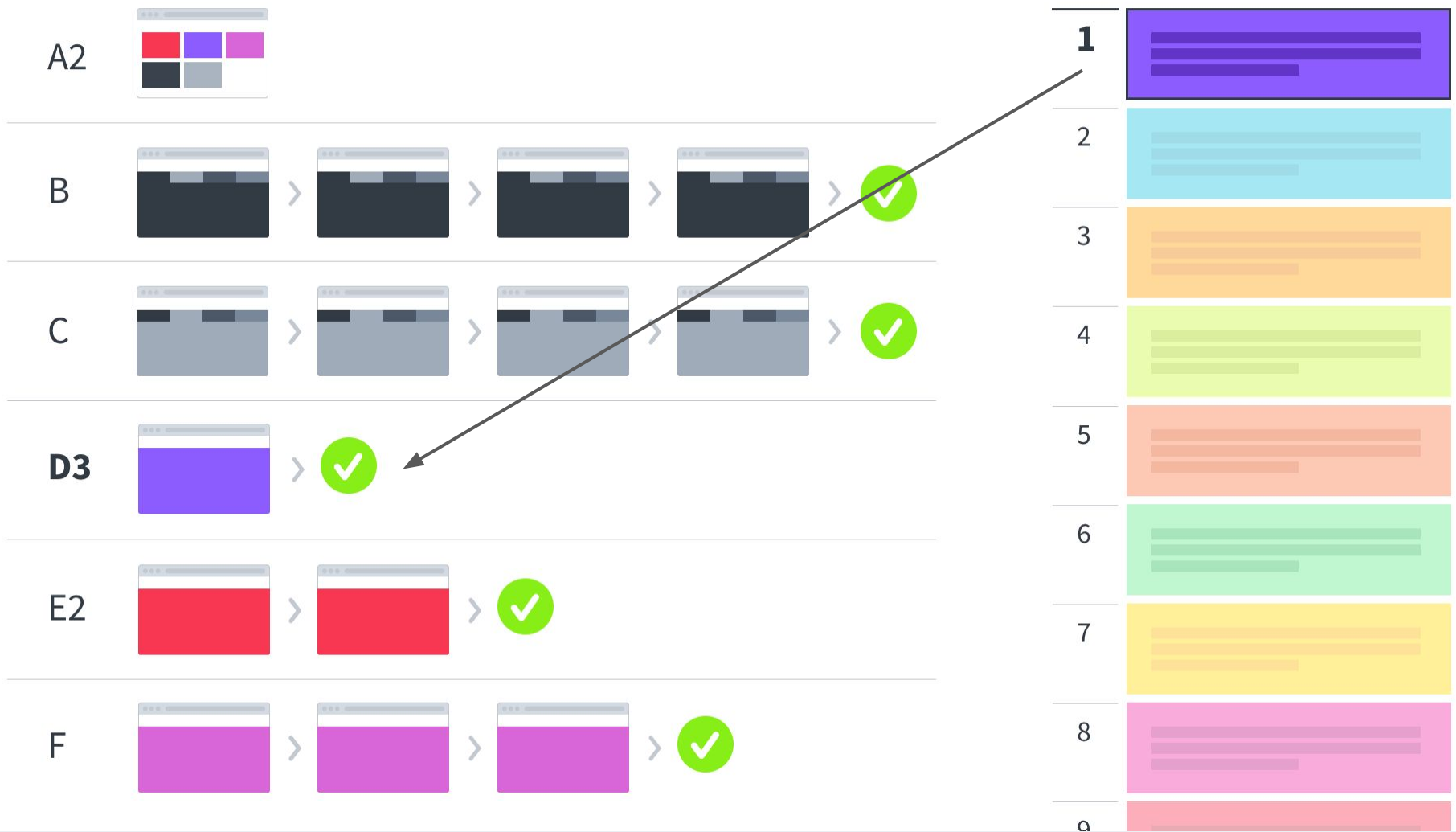
8



9



Now we're adding a new module that addresses something the legacy system never addressed. It's the highest priority, so we're doing that ahead of transforming other existing legacy workflows.



Let's say we do some testing and figure out we can improve the recently updated D module, and it's the top priority, so create a new D3 version.

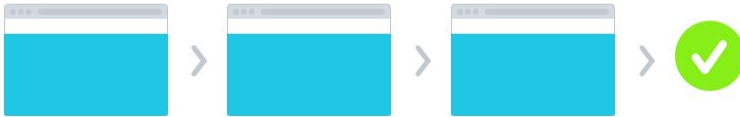
A2



B



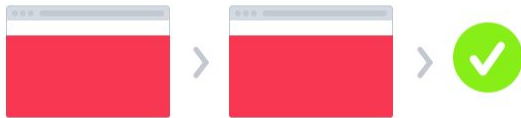
C2



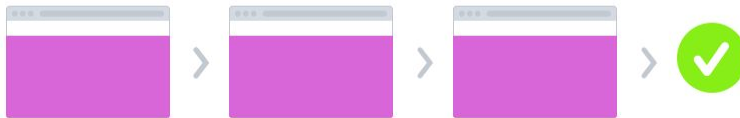
D3



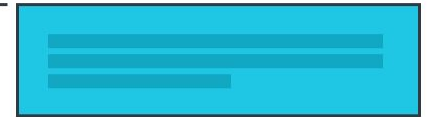
E2



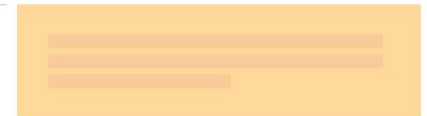
F



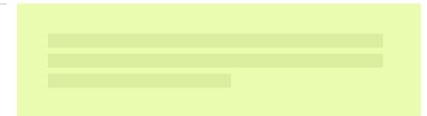
1



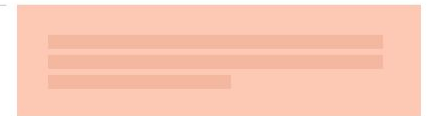
2



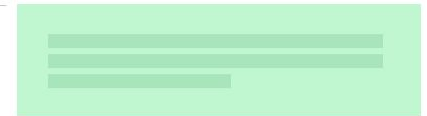
3



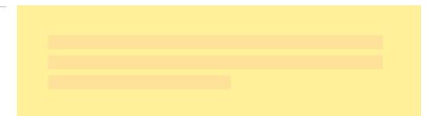
4



5



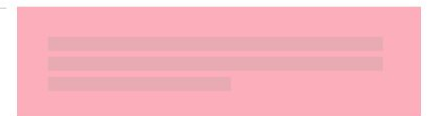
6



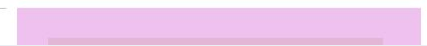
7



8



9



Now we're replacing the legacy C task with the new C2 module.
If the legacy B experience is functioning acceptably, we can just leave it...

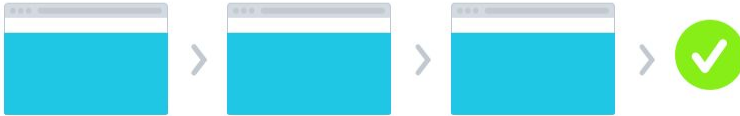
A2



B2



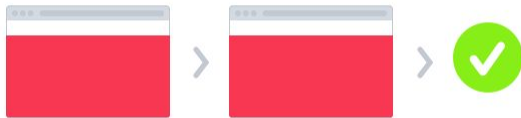
C2



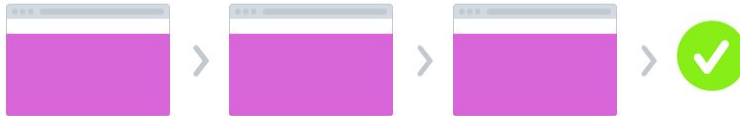
D3



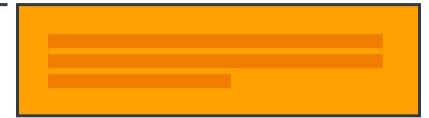
E2



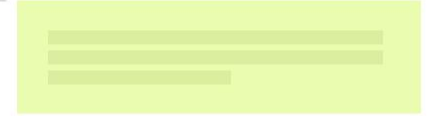
F



1



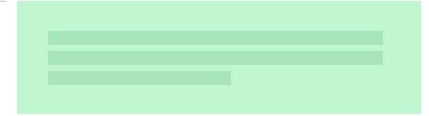
2



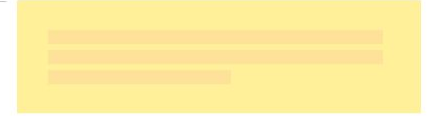
3



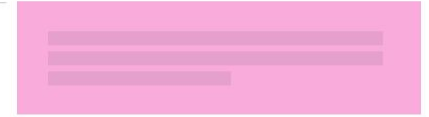
4



5



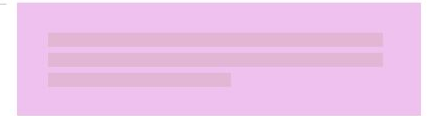
6



7



8



9

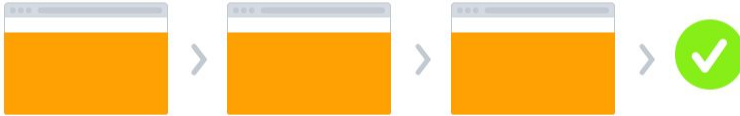


...until updating it becomes the top priority.

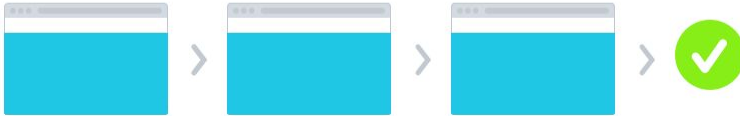
A2



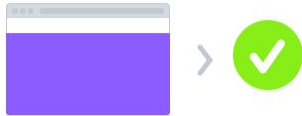
B2



C2



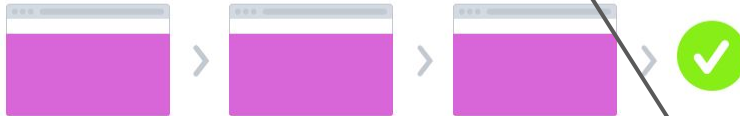
D3



E2



F



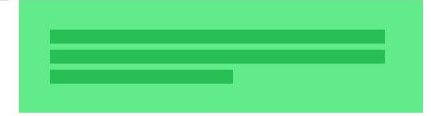
1



2



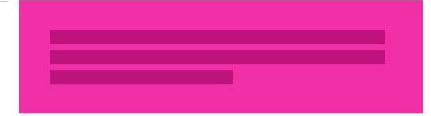
3



4



5



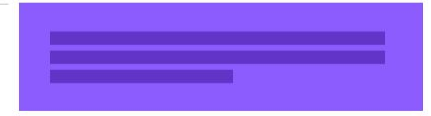
6



7



8



9



After launching a number of modules, we have a system that's been iteratively transformed. We maintain a roadmap and continue making incremental improvements.

**How can we design a modular system
to be built by multiple vendors?**

A2



B2



C2



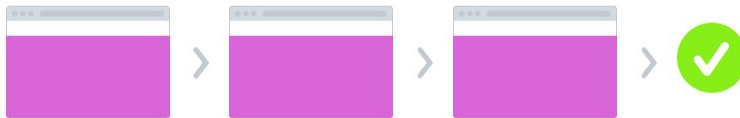
D3



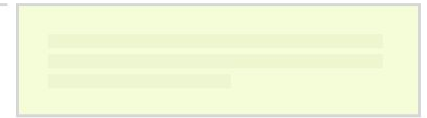
E2



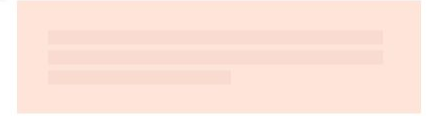
F



1



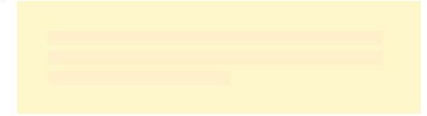
2



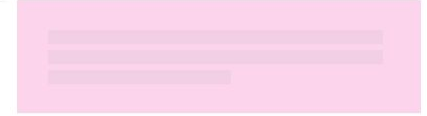
3



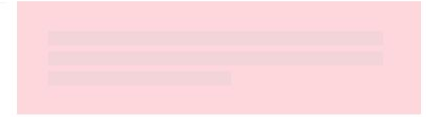
4



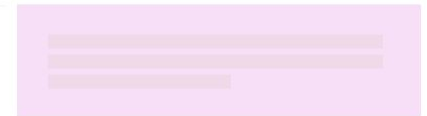
5



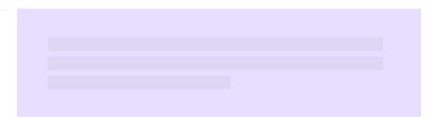
6



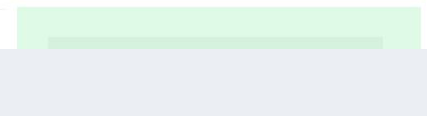
7



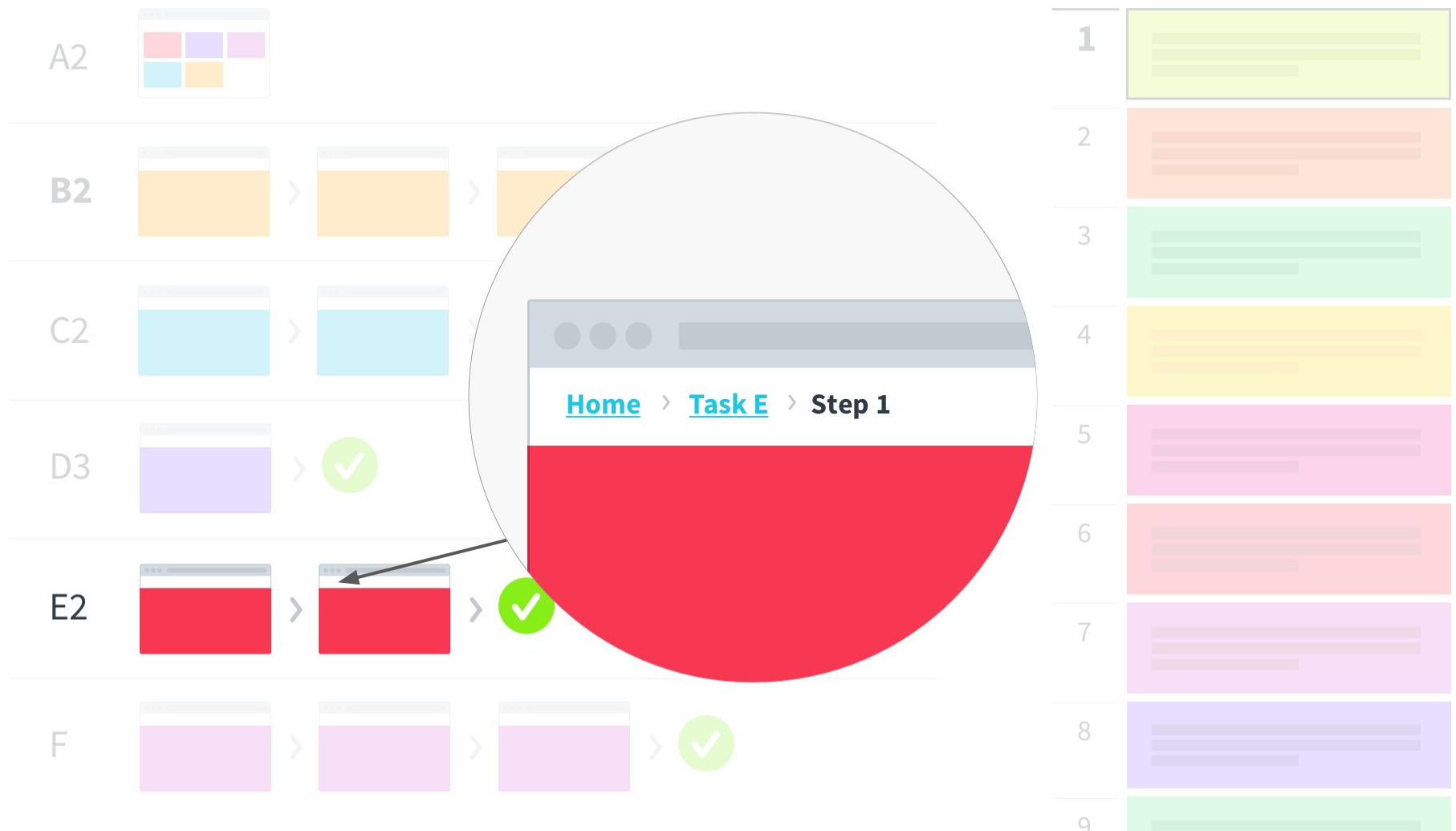
8



9



With the assumption that multiple vendor teams will work on different parts of the system, likely at the same time, we have to set some guidelines to keep the experience simple.



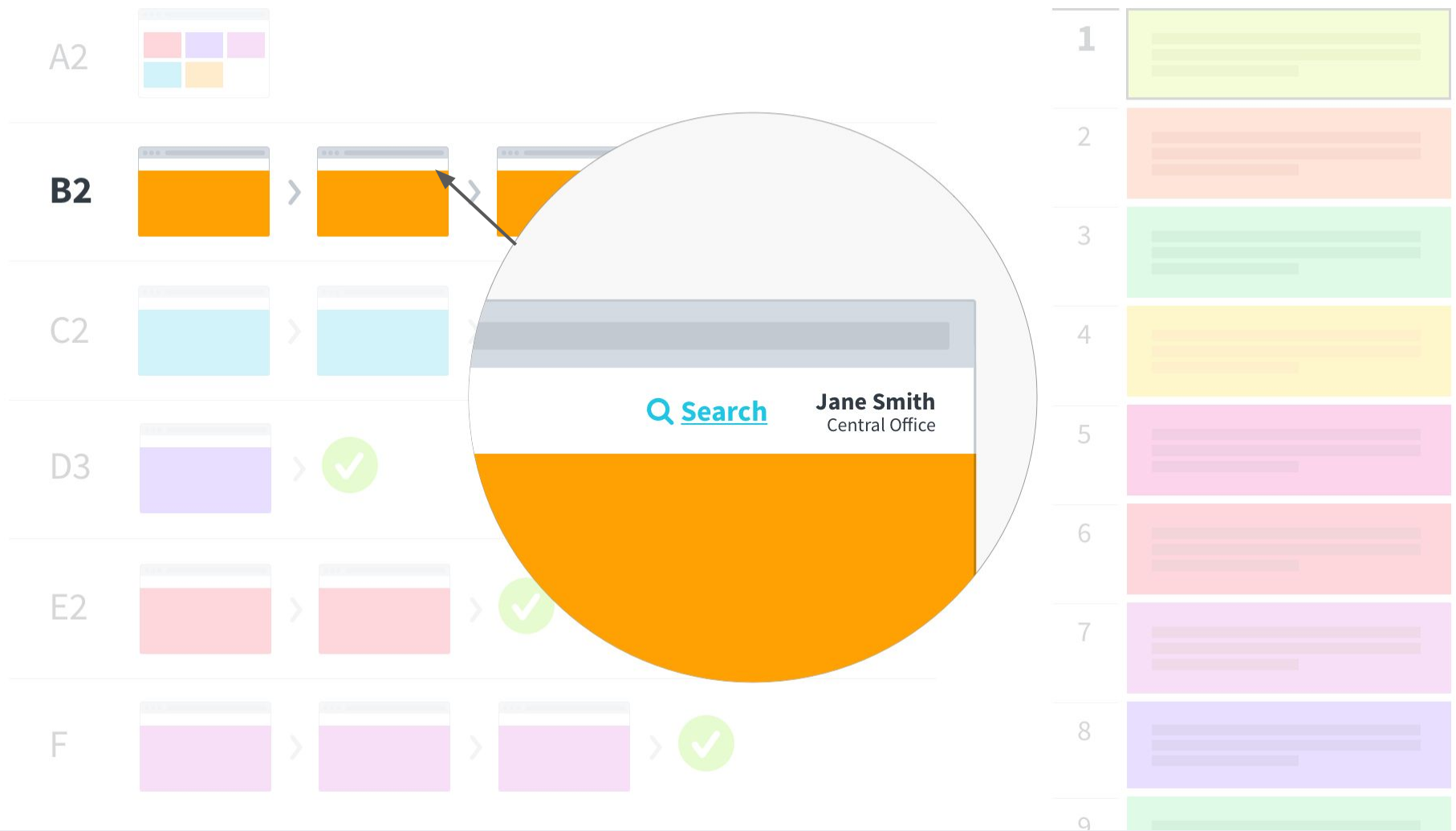
Navigation is always difficult. Once set, it can be difficult to change. Users get used to it. Stakeholders fight over it. We can avoid the issue by adopting a “hub-and-spoke” approach to global navigation. Teams building a new module don’t need to think about global nav, just link back home.



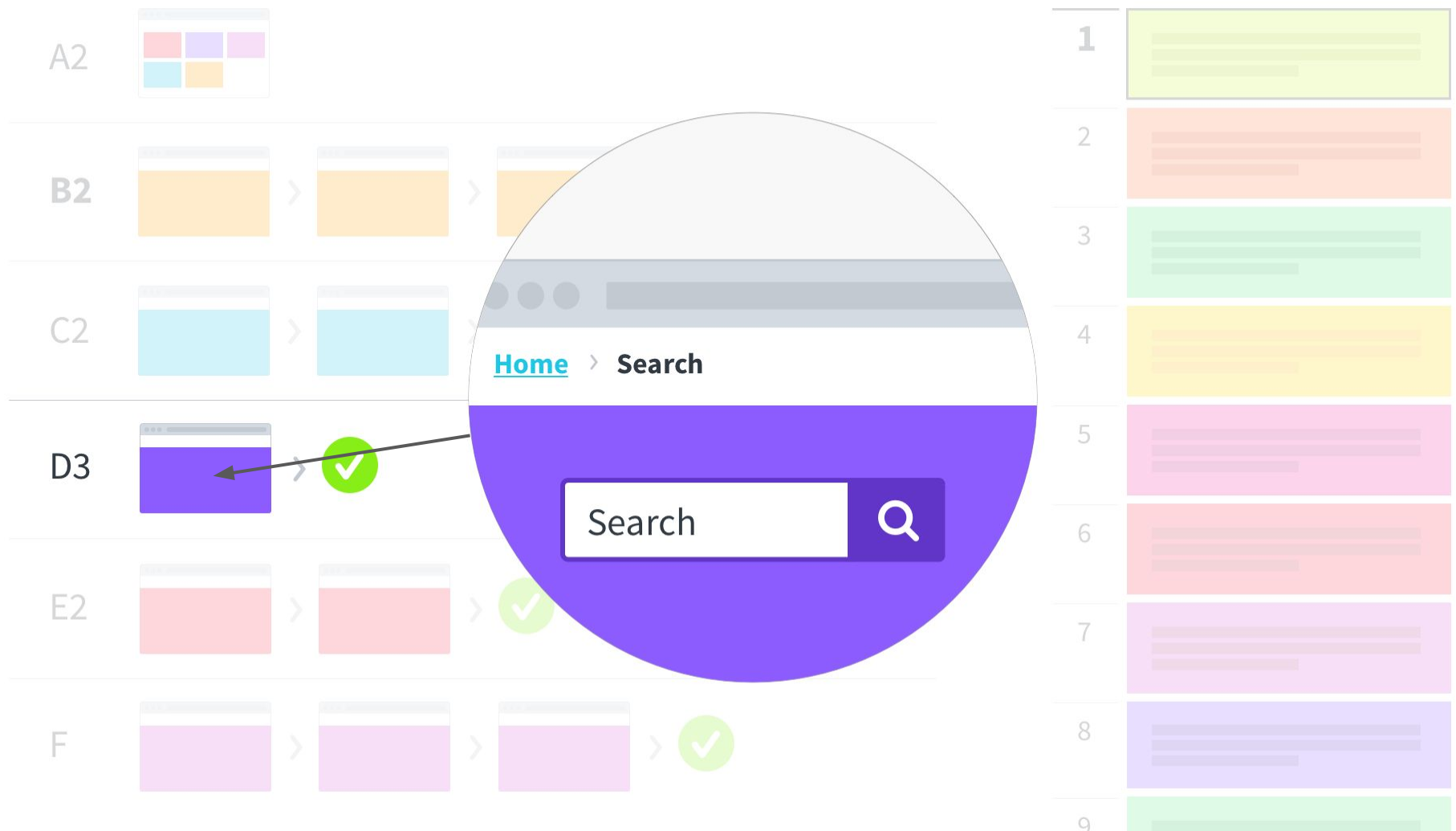
The homepage can facilitate global navigation, aggregating links to anything necessary up front.



With all navigation in a single place, outside of the various workflows, we can treat the homepage as a product itself and iterate over time. We also set the expectation that the page will continually improve based on feedback.



Each module needs to be aware of the user's identity.
Search is a first class means of navigation along with the homepage, so it's accessible in each workflow.



Search, which was our first buy, is a module itself.

Can I see a more concrete example?

View a demonstration website that illustrates these patterns. <https://federalist-proxy.app.cloud.gov/site/18f/ak-modular-experience/portal.html>