COMMON COMPONENTS

Sprint Review

Still Relevant 💪

Agenda

01 Story of the sprint

02 Collaboration

03 Next Steps

Story of the Sprint

Adapting and working in the open

CoCo Sprint 3: Still Relevant

Uptake

0 days remaining | Complete sprint | Test View

Quick filters

TO DO	IN PROGRESS	IN REVIEW	TESTING	DONE

~~BOIL-59~~ DONE 1 sub-task TD: Remove all critical path tech debt from client / server /deployment

Other Issues 35 issues

TO DO

- Promote public wiki to lab teams — Marketing — 3 — COCO-108
- GitHub Actions: deploy to Dev environment when build succeeds (CD) — CI/CD — 5 — BOIL-58

IN PROGRESS

- Compile a list of cocos for library — CoCo Library — 1 — COCO-95
- Deploy WPS client and server to OS — CI/CD — 5 — BOIL-85

IN REVIEW

- Expand journey map for discovery experience — Research — 5 — COCO-83
- Publish user research summary — Research — 5 — COCO-77
- Determine how PRs in "booster" repos trigger CI/CD — CI/CD — 2 — BOIL-82
- Discuss OS config templates with WPS — CI/CD — 1 — BOIL-84
- Make ports use environments variables — CI/CD — 2 — BOIL-81

TESTING

DONE

- Map all lab teams to the journey stage they're at — Research — 1 — ~~COCO-107~~
- Publish list of opportunities from user research — Research — 5 — ~~COCO-78~~
- Create a PR and invite external feedback — MVP — 3 — ~~BOIL-15~~
- TD: Removing FlaskGroup declaration in manage.py stops the server app from working — Boilerplates — 5 — ~~BOIL-46~~
- TD: Don't run as root — MVP

Story of the Sprint

Sprint Goal

Uptake: Get a team to start using Launchpad

Prototype what a library of common components might look like

Engagement

Public PR

kendall 10:51 AM

Good morning folks! This message comes to you from the **Common Components Team (CoCo)** - in our first two sprints we've been working to build a product we're calling the 🚀 OpenShift Launchpad 🚀 which is designed to accelerate new teams onboarding to the lab by giving them the ability to quickly spin up a new project that is deployable on OpenShift. We have a public PR open and would like to invite feedback from anyone interested. Please come see what we've been up to! 🙂
https://github.com/bcgov/openshift-launchpad/pull/30

> github.com
> **Public PR by hitankar-freshworks · Pull Request #30 · bcgov/openshift-launchpad**

17 replies March 9, 2020 2:44 PM

2 4 1

you guys have been chatting with patrick.simonian and cvarjao about bcdk right? I'm sure there's some content in the···

If you're going to use the raw oc (and I'm curious to know why CoCo chose not go go with bcdk here) I'd really stron···

So far over the past year trying to roll out common services, I have found that generally people demand more than on···

Thanks everyone 😁 Just for some background, we also steered away from completing a full CI/CD implementation in···

Actually getting that input to validate that Launchpad is even a good idea in the first place is really important to us 🙂

package.json still has ISC license

Could the oc configs be yaml instead of json? Yaml would allow them to be commented, allowing suggestions like pat···

Python code runs in rhel, JavaScript runs in Debian, was that intentional?

Hey alec.wenzowski I'm going to add you as a reviewer to the PR so we can make sure your comments are addressed 🙂

patricksimonian reviewed 8 days ago — View changes

`deployment/client.bc.json`

```
64 +          "type": "Git"
65 +        },
66 +        "resources": {
67 +          "limits": {
```

patricksimonian 8 days ago

would like to see specific amount of build `requests` and a more fine tuned limit on cpu and memory. Has there been any measurement of how much resources are consumed during the client build?

wenzowski 2 days ago

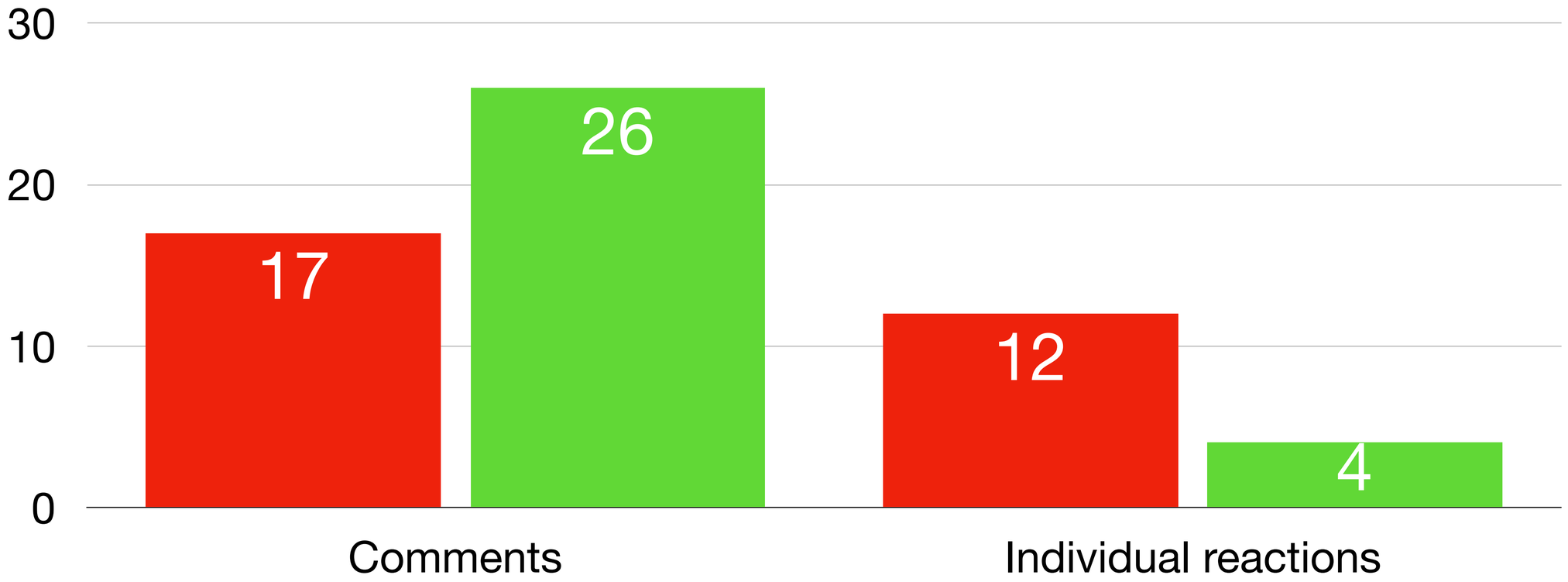
The process of tuning `requests` and `limits` itself is one I think we're looking for building competency in, beyond simply tuning this codebase. That is, we want new projects to learn how to be good shared-resource citizens, I think. Perhaps, to that end, we could put these `oc` templates in yaml format instead of json? That would allow us to have inline comments, like the salient **@patricksimonian** point above included directly. And this repo already has py so clearly the signficiant-whitespace tabs-vs-spaces point has already been provoked.

Reply...

Resolve conversation

Engagement

Public PR



User Research

Team Interviews

Invasive Species

Common Services Showcase

BC Gov Design System

Registries (Relationships)

Wildfire Predictive Services

Health Gateway

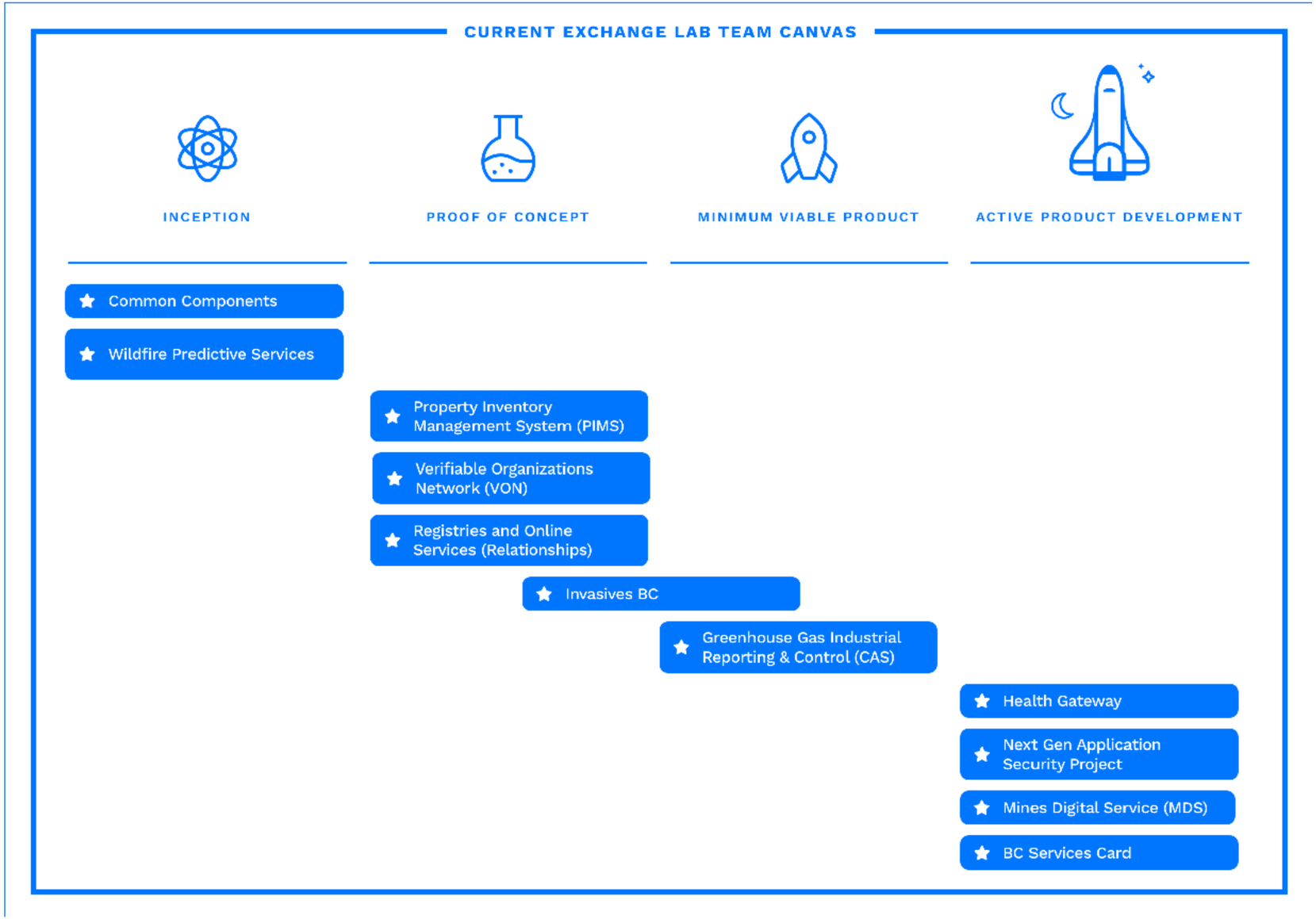
BC Developers Kit

Real Property Division PIMS

Mines Digital Services

User Research

User Journeys



Collaboration

The hustle for our first customer

Contribution Guidelines

Collaborating with Common Services Team

How to Contribute

Government employees, the public, and members of the private sector are encouraged to contribute to the repository by forking and submitting a pull request.

(If you are new to GitHub, you might start with a basic tutorial and check out a more detailed guide to pull requests.)

Pull requests will be evaluated by the repository guardians on a schedule and if deemed beneficial will be committed to the master branch.

All contributors retain the original copyright to their work, but by contributing to this project, you grant a world-wide, royalty-free, perpetual, irrevocable, non-exclusive, transferable license to all users under the terms of the license under which this project is distributed.

Adding Frontend and Backend Boosters

All boosters must be contained in public GitHub repositories. They must also meet the following requirements.

- Must contain a `Dockerfile` in the project root directory
- A backend booster must expose its API on port 5000
- A frontend booster must expose its web app on port 3000
- Backend boosters must abide by the Swagger definition

Please ensure that all boosters are able to be run without any modification to the code using standard Docker commands. Cloning a booster, navigating to the project root directory, and running `docker build -t booster . && docker run -it booster` must result in a functional product. Further, all frontend boosters should be decoupled from backend boosters and vice versa.

Once the above requirements have been met, you may request that the booster be added to the OpenShift Launchpad. This may be requested in the following ways.

- Create a GitHub issue explaining the request and include the booster repository

Contribution Guidelines

Swagger

Swagger Editor Supported by SMARTBEAR File ▾ Edit ▾ Generate Server ▾ Generate Client ▾

```yaml
swagger: '2.0'
info:
  description: This document outlines the API spec that OpenShift launchpad client and
    server modules must subscribe to. Please note that the API base URL must be filled in
    and will depend on the deployment of the application.
  version: '0.1.0'
  title: OpenShift Launchpad
  license:
    name: Apache 2.0
    url: https://github.com/bcgov/openshift-launchpad/blob/master/LICENSE
externalDocs:
  description: OpenShift Launchpad GitHub Repository
  url: https://github.com/bcgov/openshift-launchpad
host: '[YOUR_OPENSHIFT_ROUTE]'
basePath: /api
consumes:
- application/json
produces:
- application/json
schemes:
- https
- http
tags:
- name: foo
  description: Operations related to Foo instances
paths:
  /foo:
    get:
      tags: [foo]
      summary: List all Foo instances
      operationId: listAllFoos
      responses:
        200:
          description: Successful operation
          schema:
            $ref: '#/definitions/FooList'
    post:
      tags: [foo]
      summary: Create a new Foo instance
      operationId: createFoo
      parameters:
```

OpenShift Launchpad 0.1.0

[Base URL: [YOUR_OPENSHIFT_ROUTE]/api]
https://raw.githubusercontent.com/bcgov/openshift-launchpad/develop/swagger.yaml

This document outlines the API spec that OpenShift launchpad client and server modules must subscribe to. Please note that the API base URL must be filled in and will depend on the deployment of the application.

Apache 2.0

OpenShift Launchpad GitHub Repository

Schemes

HTTPS

foo Operations related to Foo instances

GET	/foo	List all Foo instances
POST	/foo	Create a new Foo instance
GET	/foo/{fooId}	Get Foo instance by ID
PUT	/foo/{fooId}	Update Foo instance by ID
DELETE	/foo/{fooId}	Deletes a pet

Pivot!

Agile Manifesto

We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.

Pivot!

Agile Manifesto

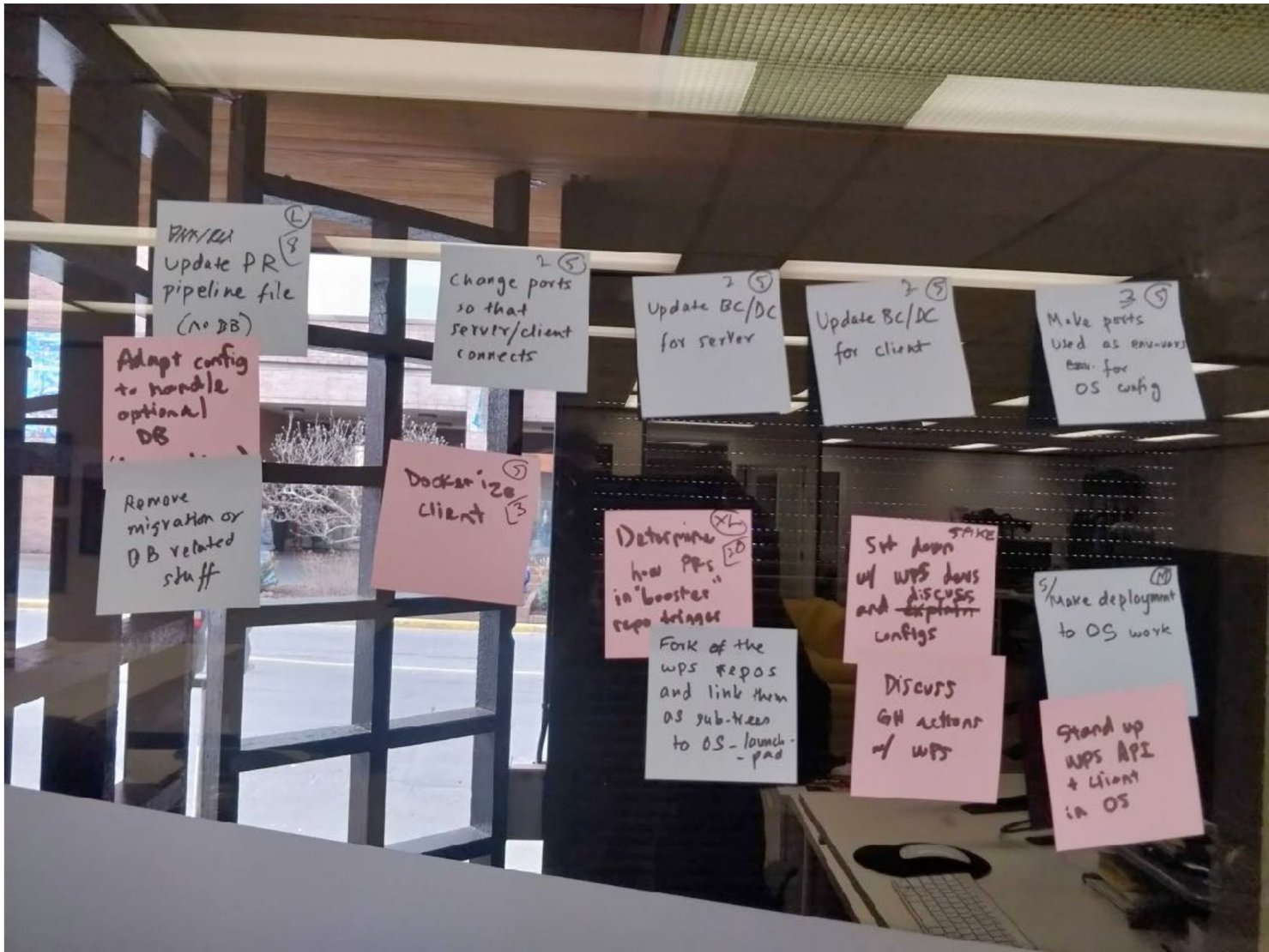
We are uncovering better ways of developing
software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on
the right, we value the items on the left more.

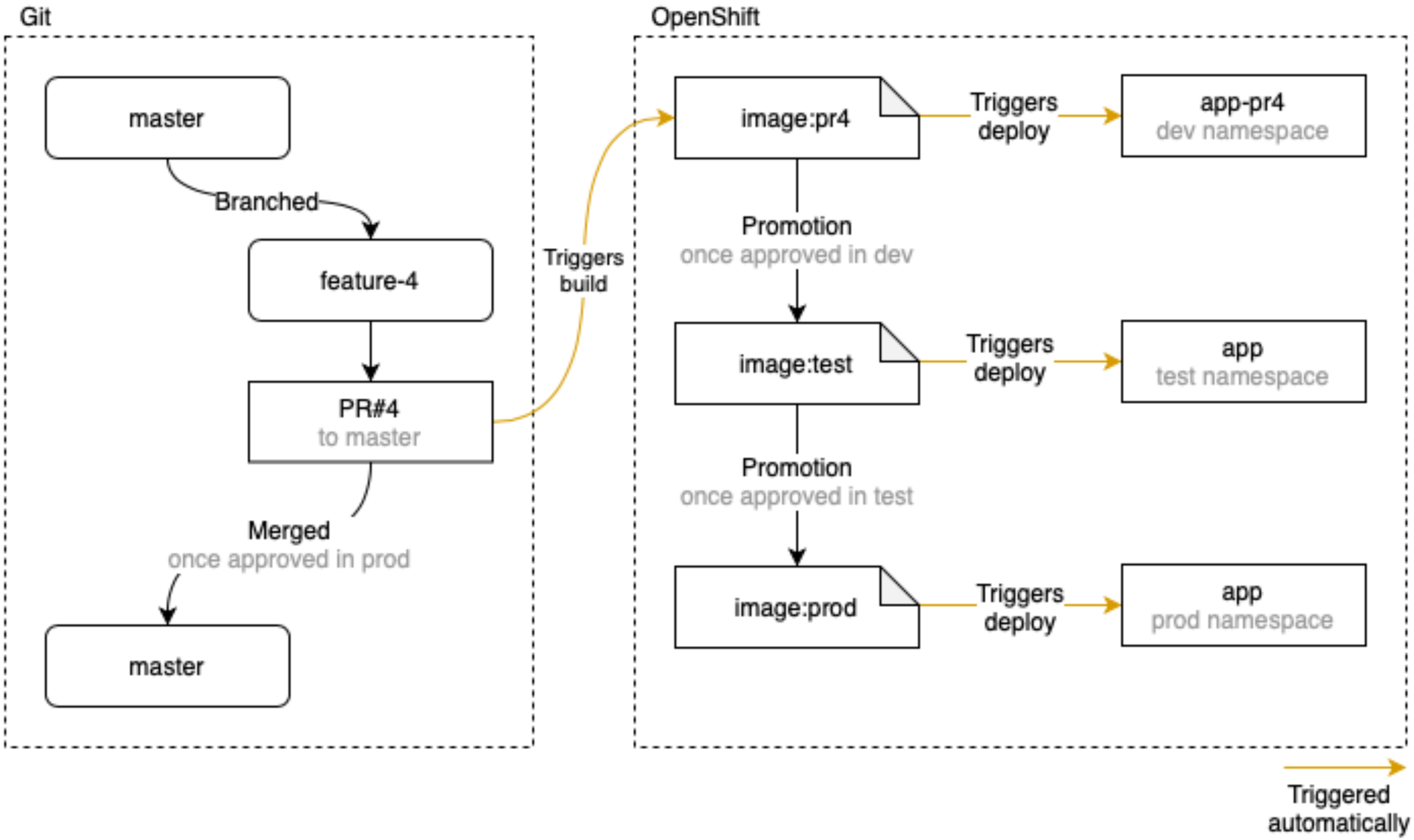
Pivot!

Collaboration with WPS



Pivot!

Collaboration with WPS



Pivot!

Collaboration with WPS

DEMO

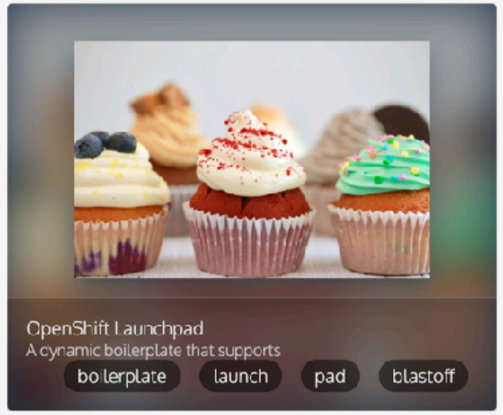
Next Steps

Betting Table and the future of Launchpad

Jim Lahey

Fullstack Developer

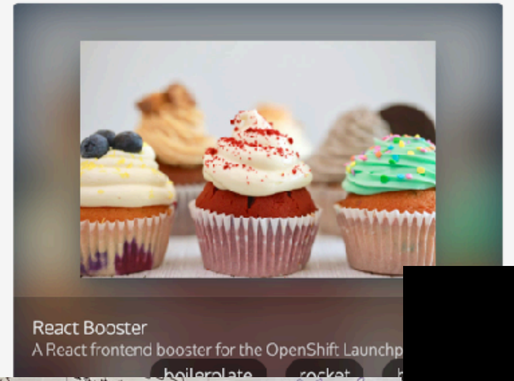
365	110	54
commits	stars	repositories

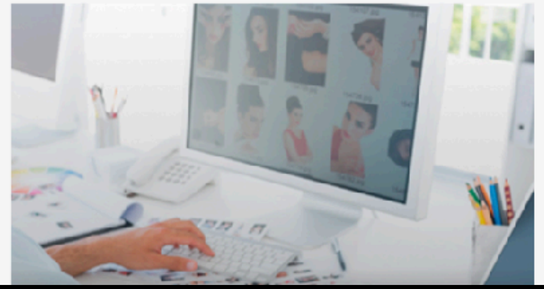


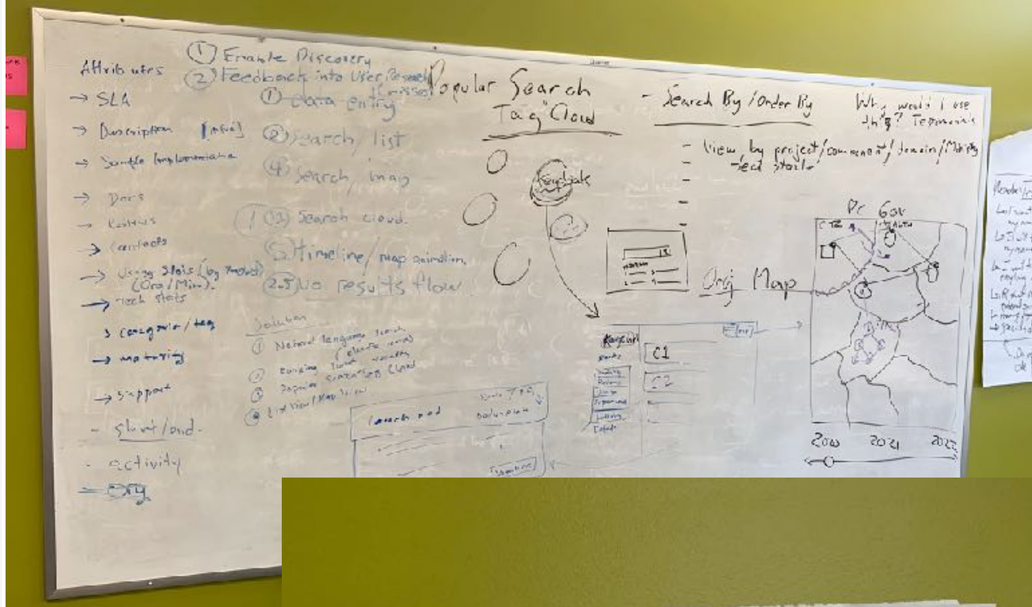
Randy Bobandy

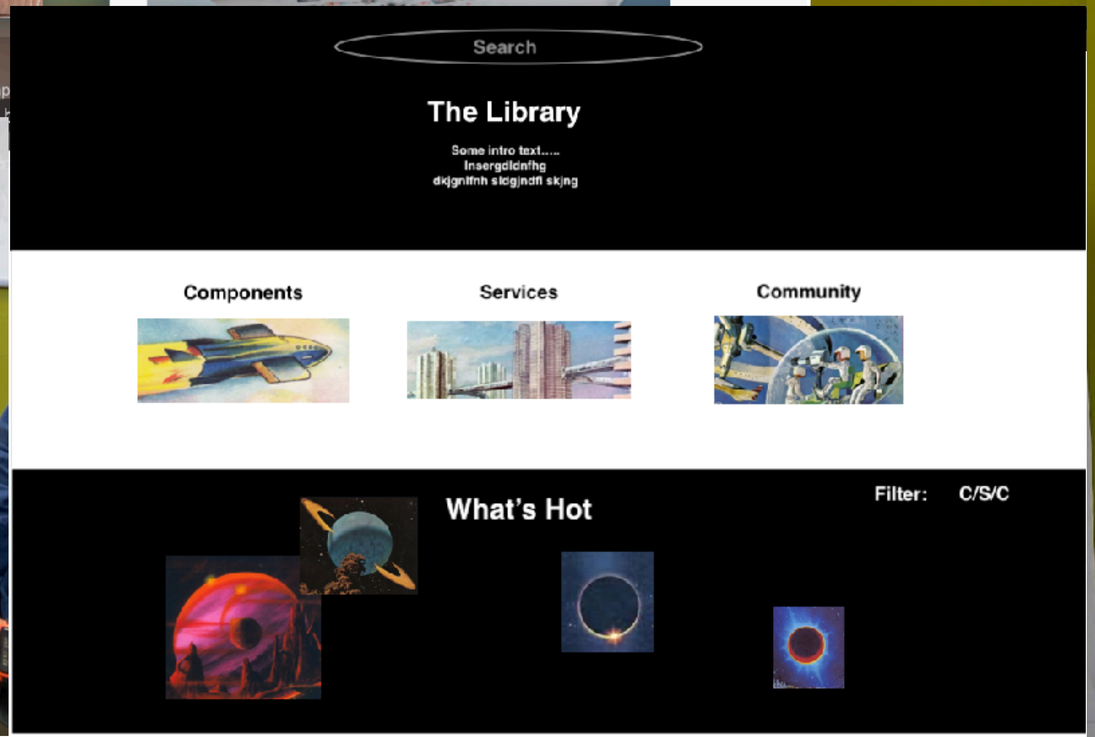
Fullstack Developer

365	110	54









Goals:

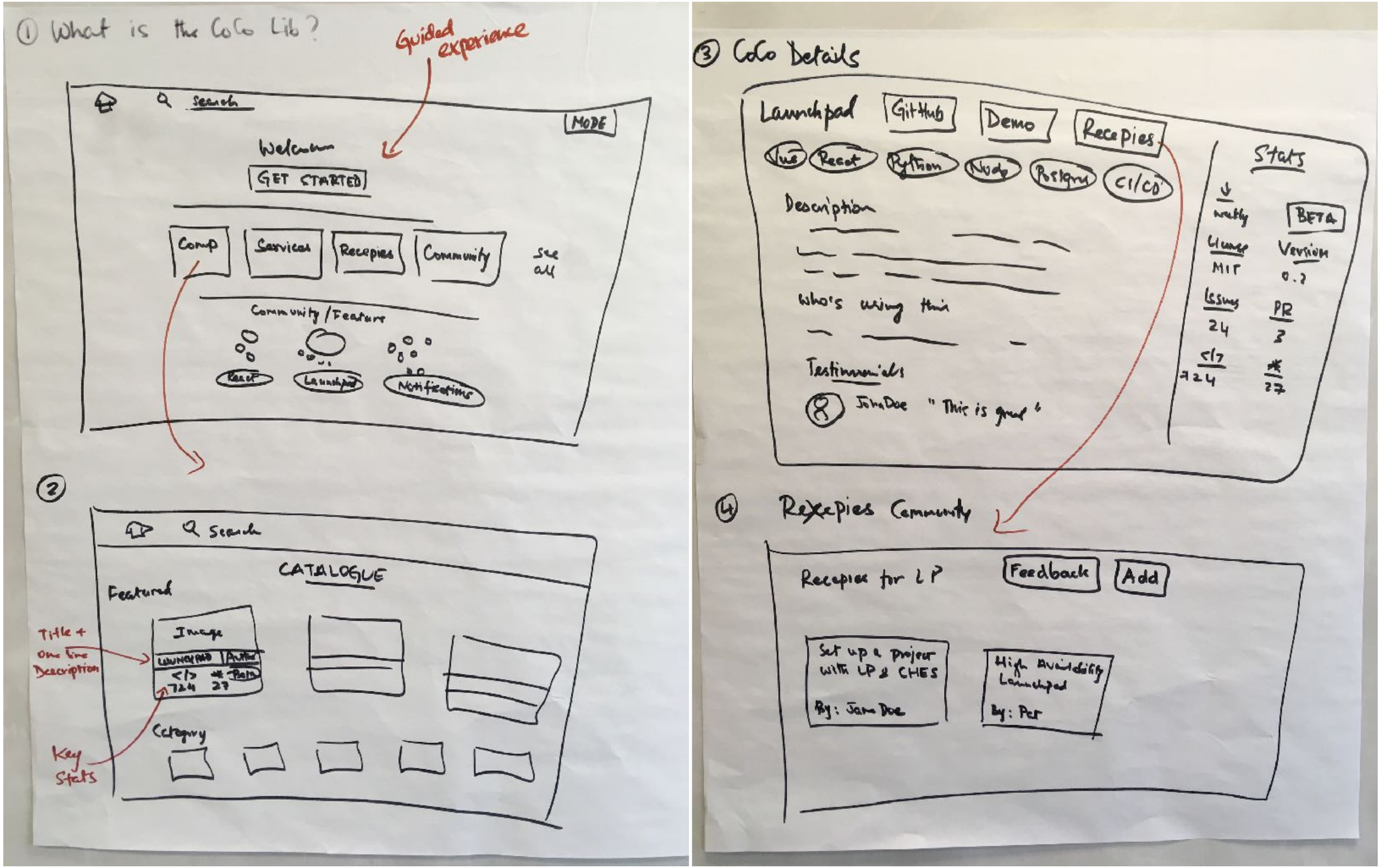
1. Enable Discovery
2. Feedback For User Research

Journey

~~1. DATA ENTRY~~

2. DEFAULT SEARCH VIEW
   ↳ NO RESULTS FEEDBACK
3. USAGE VISUALIZATION
4. SEARCH HISTORY TAG CLOUD

Bonus: USAGE TIMELINE VISUALIZATION


① What is the CoCo Lib?
Guided experience
Search
MODE
Welcome
GET STARTED
Comp
Services
Recepies
Community
see all
Community / Feature
React
Launchpad
Notifications
②
Search
CATALOGUE
Featured
Image
Title + One line Description
Key Stats
Category
③ CoCo Details
Launchpad
GitHub
Demo
Recepies
Vue
React
Python
Nuxt
Postgres
CI/CD
Description
Who's using this
Testimonials
John Doe "This is great"
Stats
Weekly
BETA
License
MIT
Version
0.2
Issues
24
PR
3
Stars
124
27
④ Recepies Community
Recepies for LP
Feedback
Add
Set up a project with LP & CHES
By: Jane Doe
High Availability Launchpad
By: Per

Betting Table

Order of Priorities

01 **CoCo Library** Catalogue and Community

02 **Notify** How good is it? Can we consume it?

03 **Launchpad** Maintenance and drawing uptake

Lessons-Help-Feedback

Contact Us!

#team-coco

#common-components

Exchange Lab

(3rd Floor)

jfournier@freshworks.io

kendall@freshworks.io

rachit@freshworks.io

Thank You

From #team-coco

BRITISH COLUMBIA

freshworks STUDIO

CGI