Frequent
Small
Good
Decoupled

FSGD

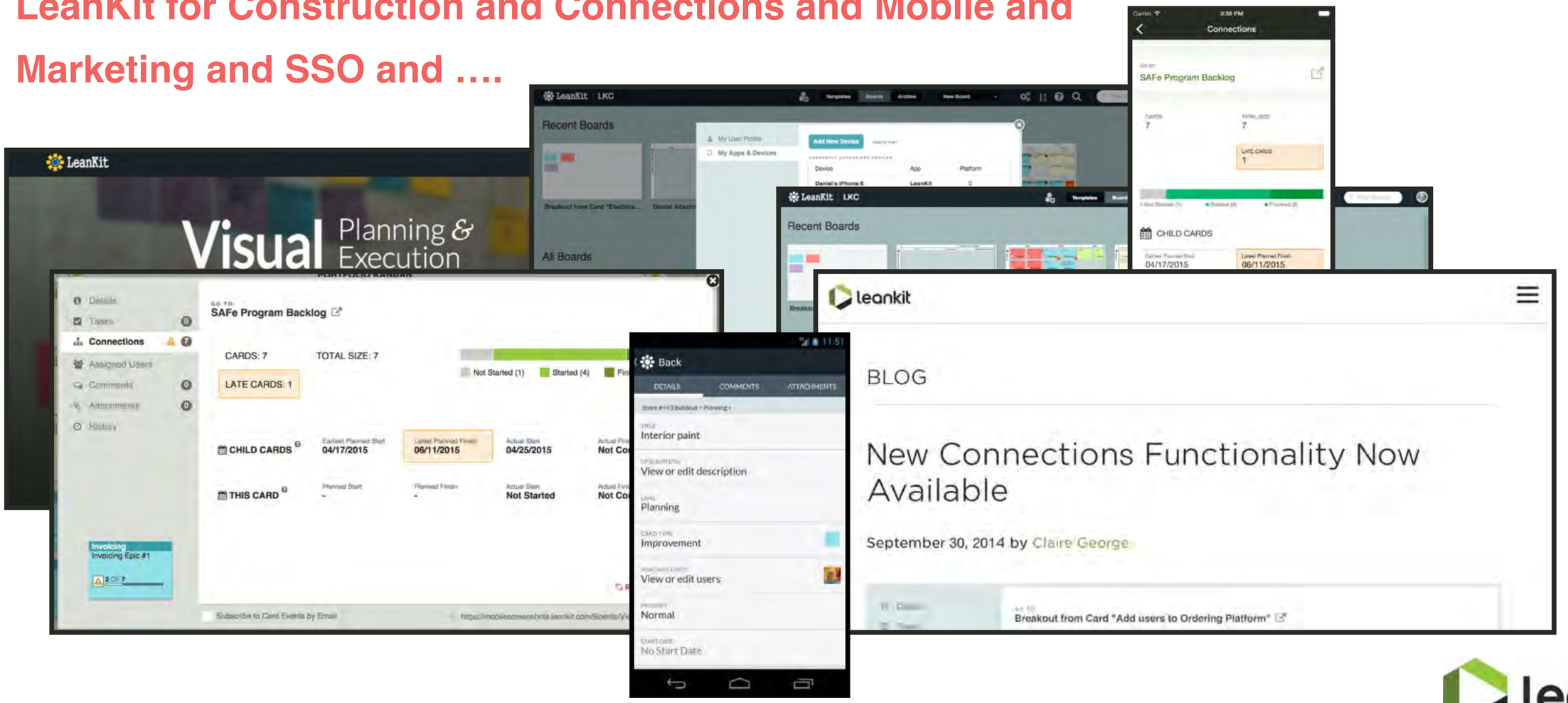
(Fizz Good)

what **Fizz Good**
and
what **doesn’t**

leankit

Hurray! Never Again

LeanKit for Construction and Connections and Mobile and Marketing and SSO and ….



leankit

We love Lean, Kanban, Agile, DevOps

AND

leankit

Frequent
Small
Good
Decoupled

FSGD

(Fizz Good)

Frequent

Small

Good

Decoupled

Annually

Quarterly

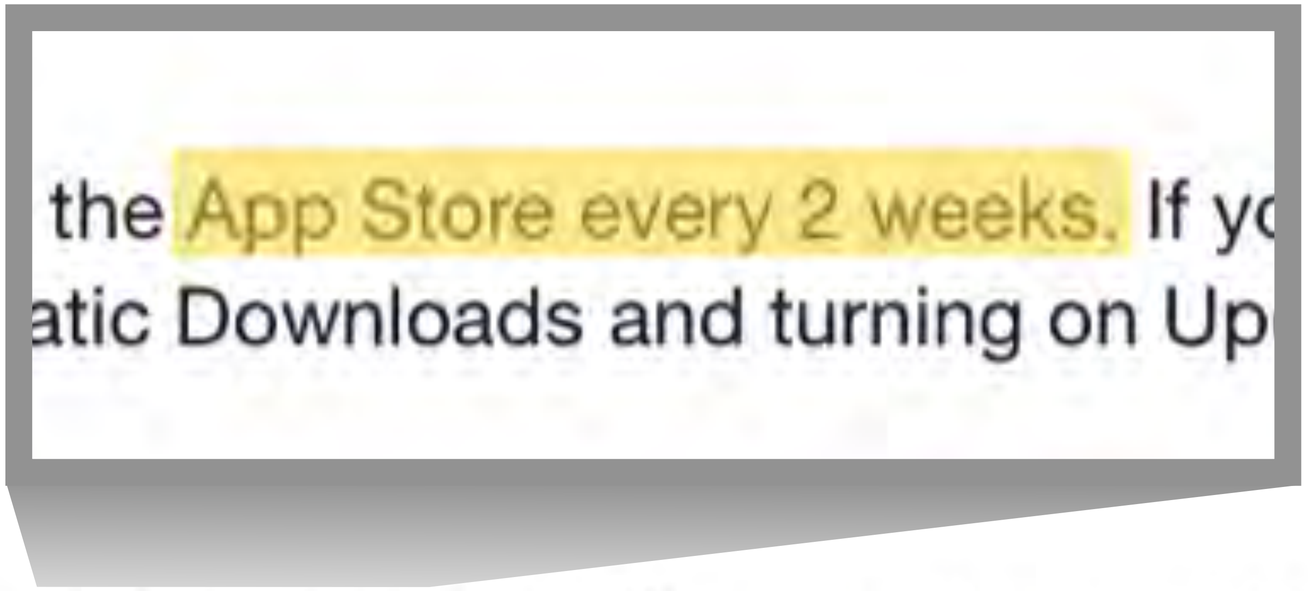
Sporadic

leankit

Release Frequently for ...

- Feedback
- Visibility into the work
- Opportunity to reprioritize
  - Maybe you're done
  - Critical issues
  - Unplanned work
  - Changing business realities

leankit



Version 16.0 Posted Oct 15, 2014

Thanks for using Facebook! To make our app better for you, we bring updates to the App Store every 2 weeks. If you have iOS7 or up, you can update the app automatically (without checking back here) by going to Settings > iTunes & App Store > Automatic Downloads and turning on Updates.

Every update of our Facebook app includes improvements for speed and reliability. As other new features become available, we'll highlight those for you in the app.

Version 15.0 Posted Sep 29, 2014

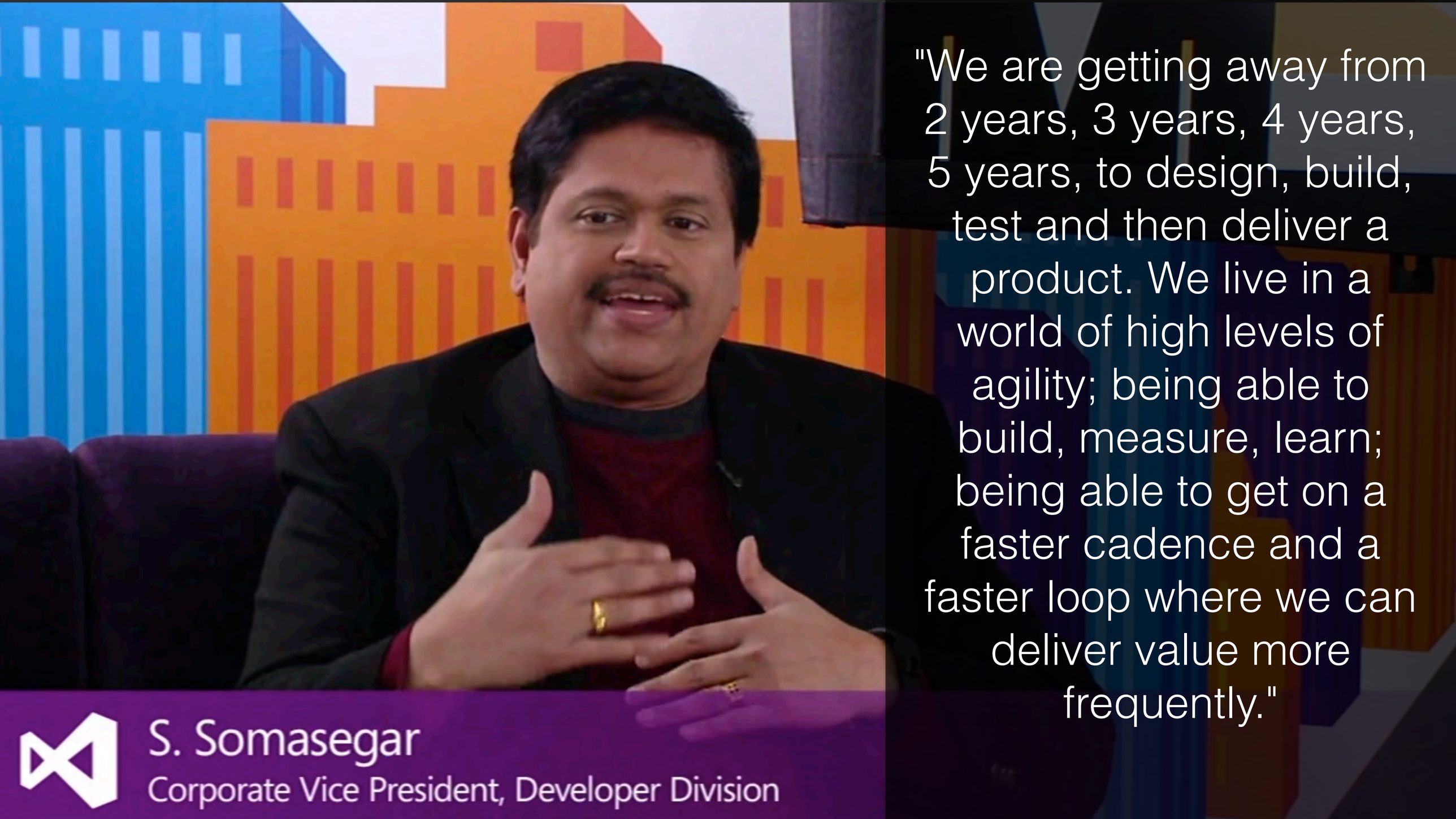
Thanks for using Facebook! To make our app better for you, we bring updates to the App Store every 4 weeks. If you have iOS7 or up, you can update the app automatically (without checking back here) by going to Settings > iTunes & App Store > Automatic Downloads and turning on Updates.

Every update of our Facebook app includes improvements for speed and reliability. As other new features become available, we'll highlight those for you in the app.

Version 14.9 Posted Sep 17, 2014

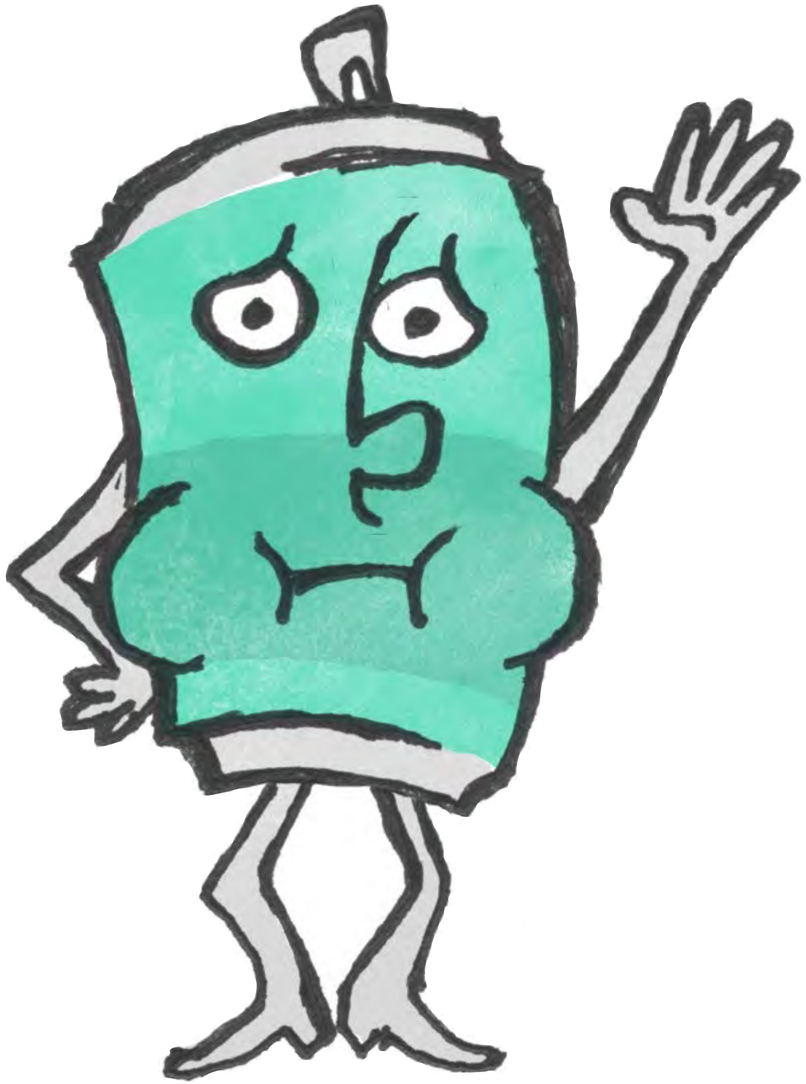
Thanks for using Facebook! To make our app better for you, we bring updates to the App Store every 4 weeks. If you have iOS7 or up, you can update the app automatically (without checking back here) by going to Settings > iTunes & App Store > Automatic Downloads and turning on Updates.

Every update of our Facebook app includes improvements for speed and reliability. As other new features become available, we'll highlight those for you in the app.



"We are getting away from 2 years, 3 years, 4 years, 5 years, to design, build, test and then deliver a product. We live in a world of high levels of agility; being able to build, measure, learn; being able to get on a faster cadence and a faster loop where we can deliver value more frequently."

… and we don’t just mean to the customer

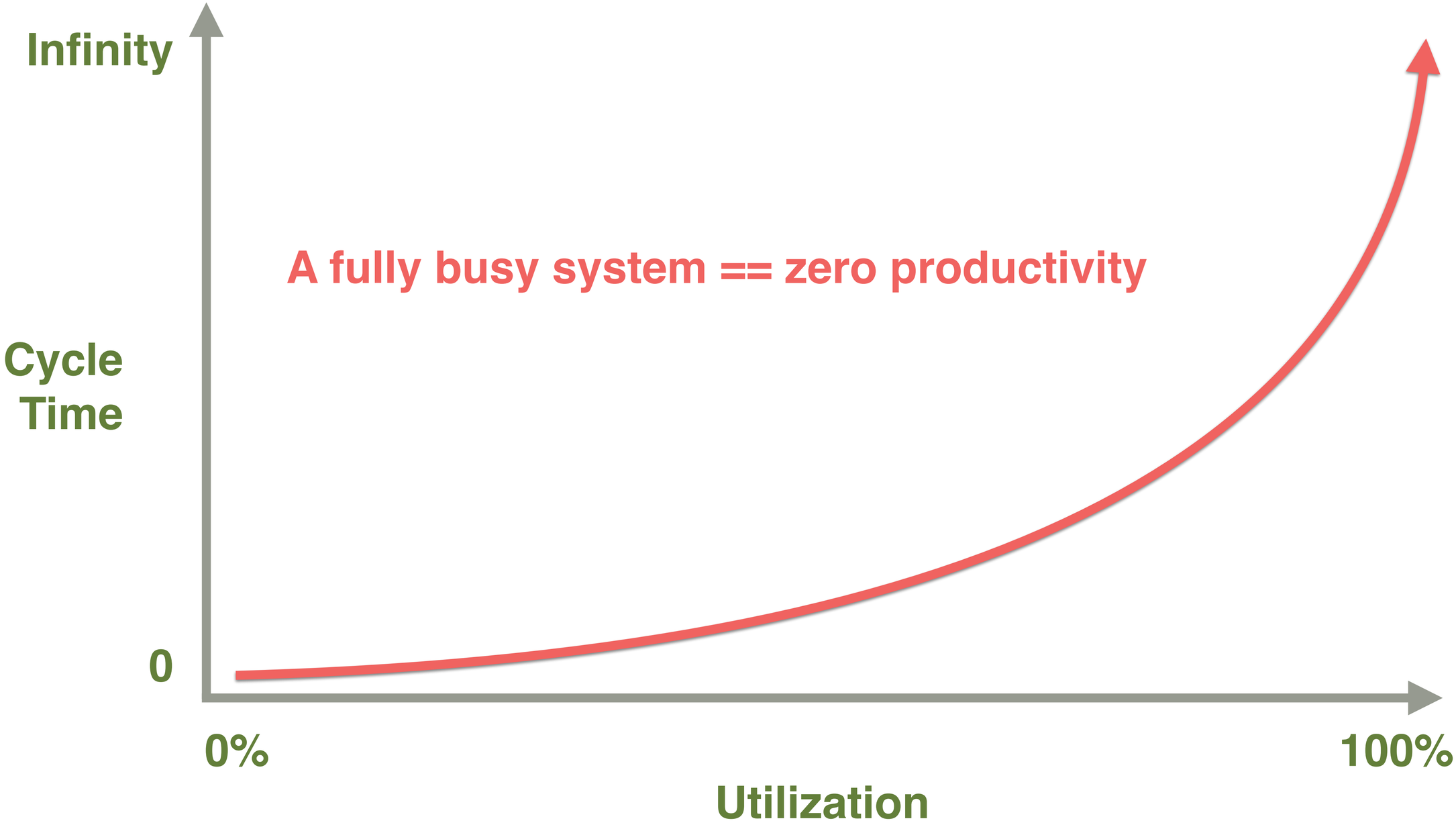


leankit

because math

(not the Agile Manifesto)

leankit



All credit and praise rightfully belongs to to Don Reinertsen

leankit

A little bit of reduction goes a long way

WIP Limit	1	2	5	10	20	Infinite
Average Cycle Time	1.0	1.5	2.8	4.6	7.2	10
Time in Queue	0	0.5	1.8	3.6	6.2	9
Utilization Percent	47	63	79	85	89	90
Slack Time	53	37	21	15	11	10
Blocking Percent	47	30	13	5	1	0

All credit and praise rightfully belongs to Don Reinertsen

leankit

Let's assume the starting point is 90% utilization, 10 day cycle time & no WIP limits

WIP Limit	1	2	5	10	20	Infinite
Average Cycle Time	1.0	1.5	2.8	4.6	7.2	10
Time in Queue	0	0.5	1.8	3.6	6.2	9
Utilization Percent	47	63	79	85	89	90
Slack Time	53	37	21	15	11	10
Blocking Percent	47	30	13	5	1	0

All credit and praise rightfully belongs to Don Reinertsen

leankit

A WIP limit of twice current average makes us 28% faster in return for 1% slack

WIP Limit	1	2	5	10	20	Infinite
Average Cycle Time	1.0	1.5	2.8	4.6	**7.2**	10
Time in Queue	0	0.5	1.8	3.6	**6.2**	9
Utilization Percent	47	63	79	85	**89**	90
Slack Time	53	37	21	15	**11**	10
Blocking Percent	47	30	13	5	**1**	0

All credit and praise rightfully belongs to Don Reinertsen

leankit

A WIP limit equal to current average makes us 54% faster in return for 5% slack

WIP Limit	1	2	5	10	20	Infinite
Average Cycle Time	1.0	1.5	2.8	**4.6**	7.2	10
Time in Queue	0	0.5	1.8	**3.6**	6.2	9
Utilization Percent	47	63	79	**85**	89	90
Slack Time	53	37	21	**15**	11	10
Blocking Percent	47	30	13	**5**	1	0

All credit and praise rightfully belongs to Don Reinertsen

leankit

It’s tough to limit WIP with variable batch sizes

leankit

Frequent

Small

Good

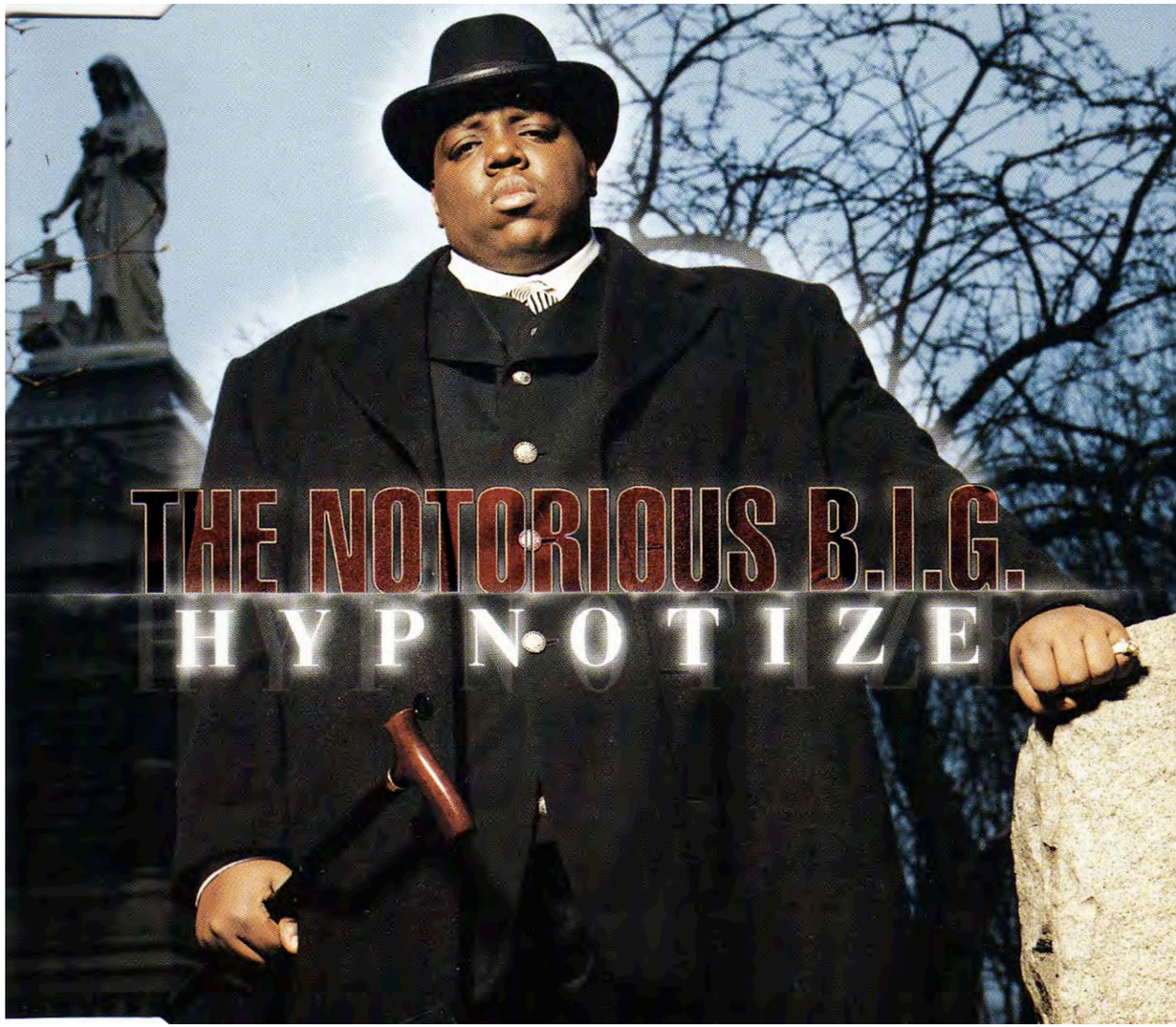
Decoupled

Big

Some things are just naturally big?

leankit

Some things are just notoriously big



leankit

Why?

Why?

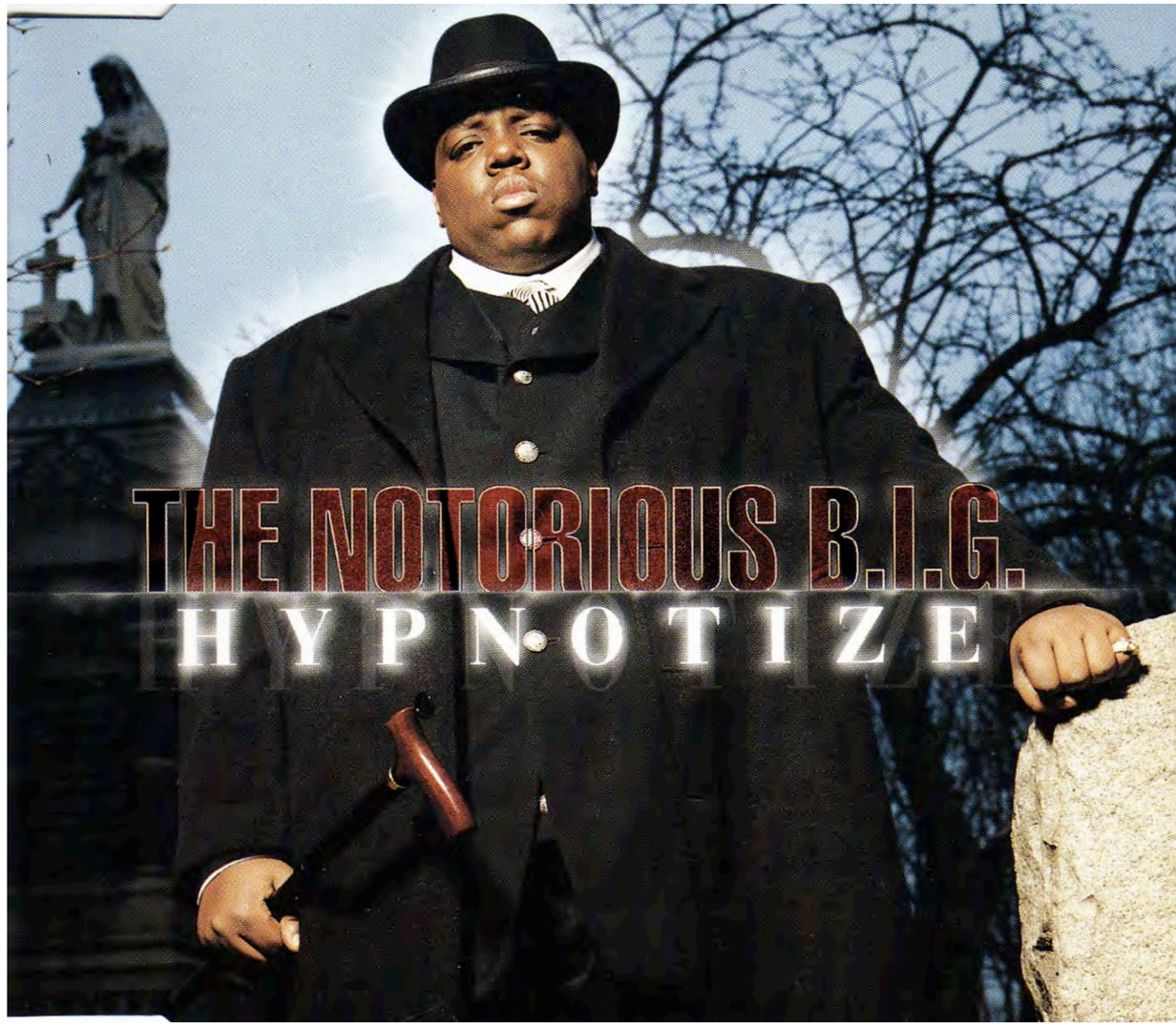
Why?

Why?

Why?

Five Why's

Root Cause Analysis



leankit

5 Why example: notoriously big things

Why does it have to be so big?

It does lots of things and has lots of components

Why does it do so many things?

Because they are interrelated.

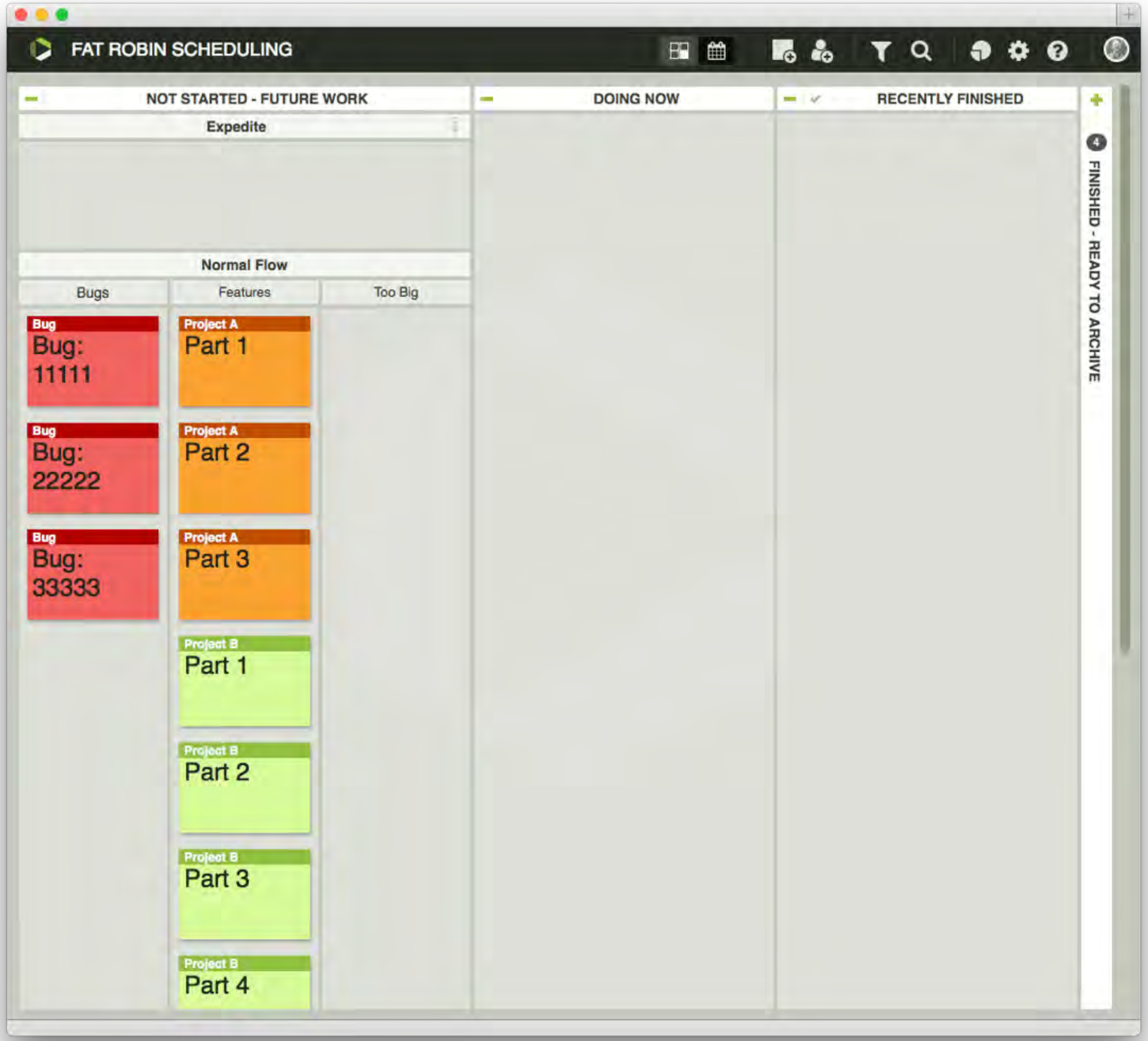
Why can't they be broken into several releases?

Because it is costly to deploy each of them

Why is that?

Because we do not have an automated testing and release process

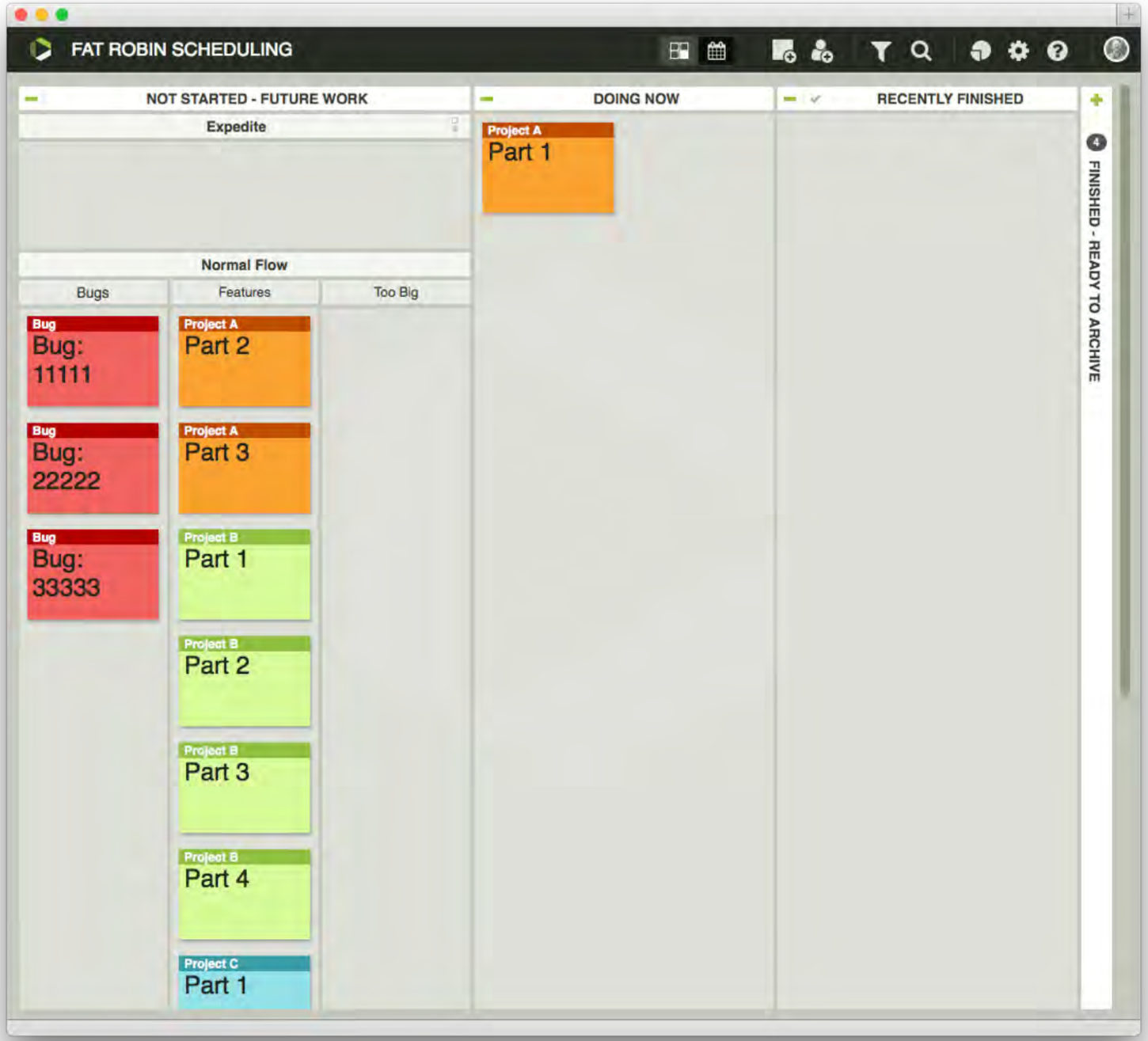
leankit



Round Robin Scheduling

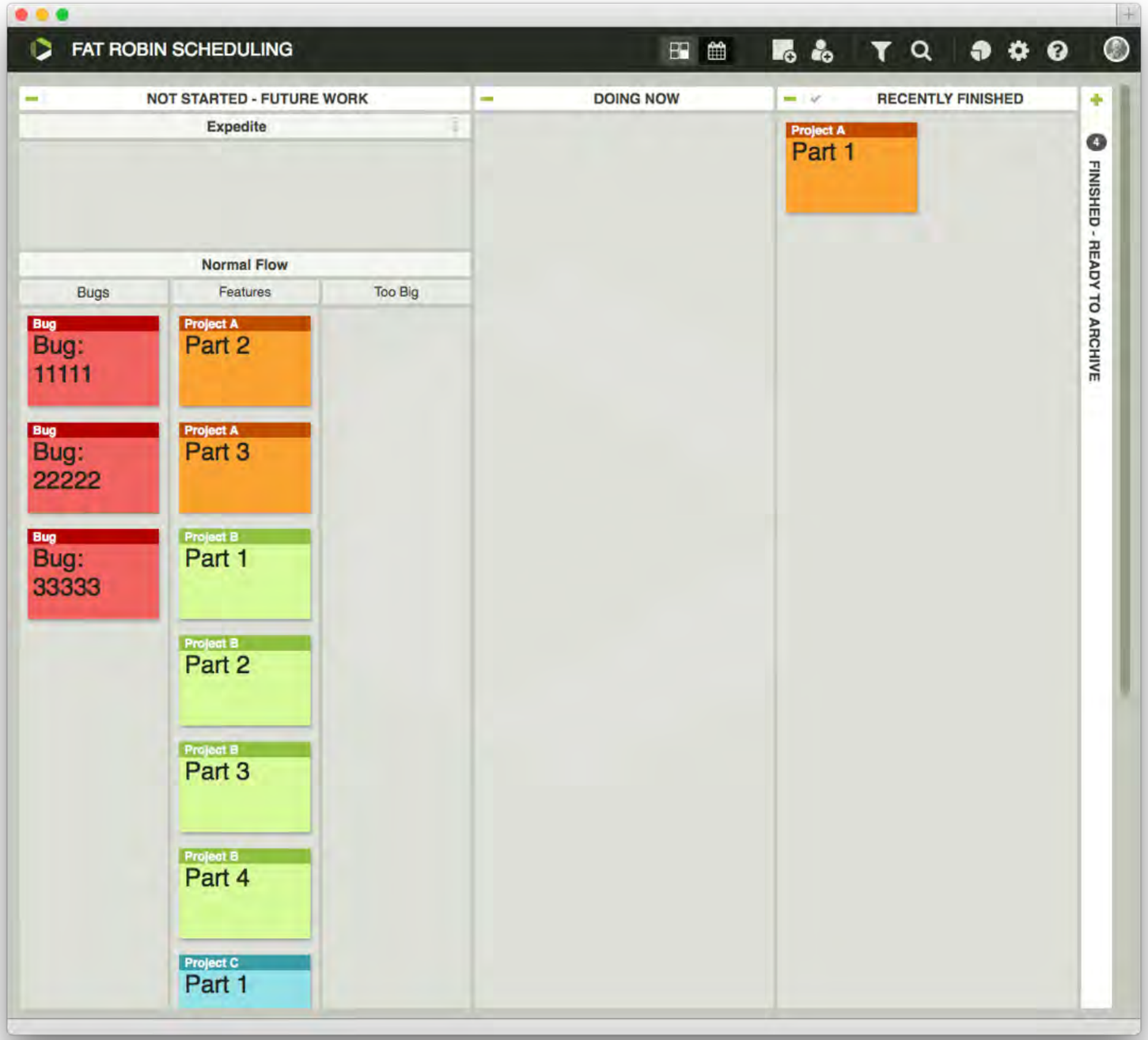
- Breaking work into smaller slices
- Just-in-time decision making

leankit



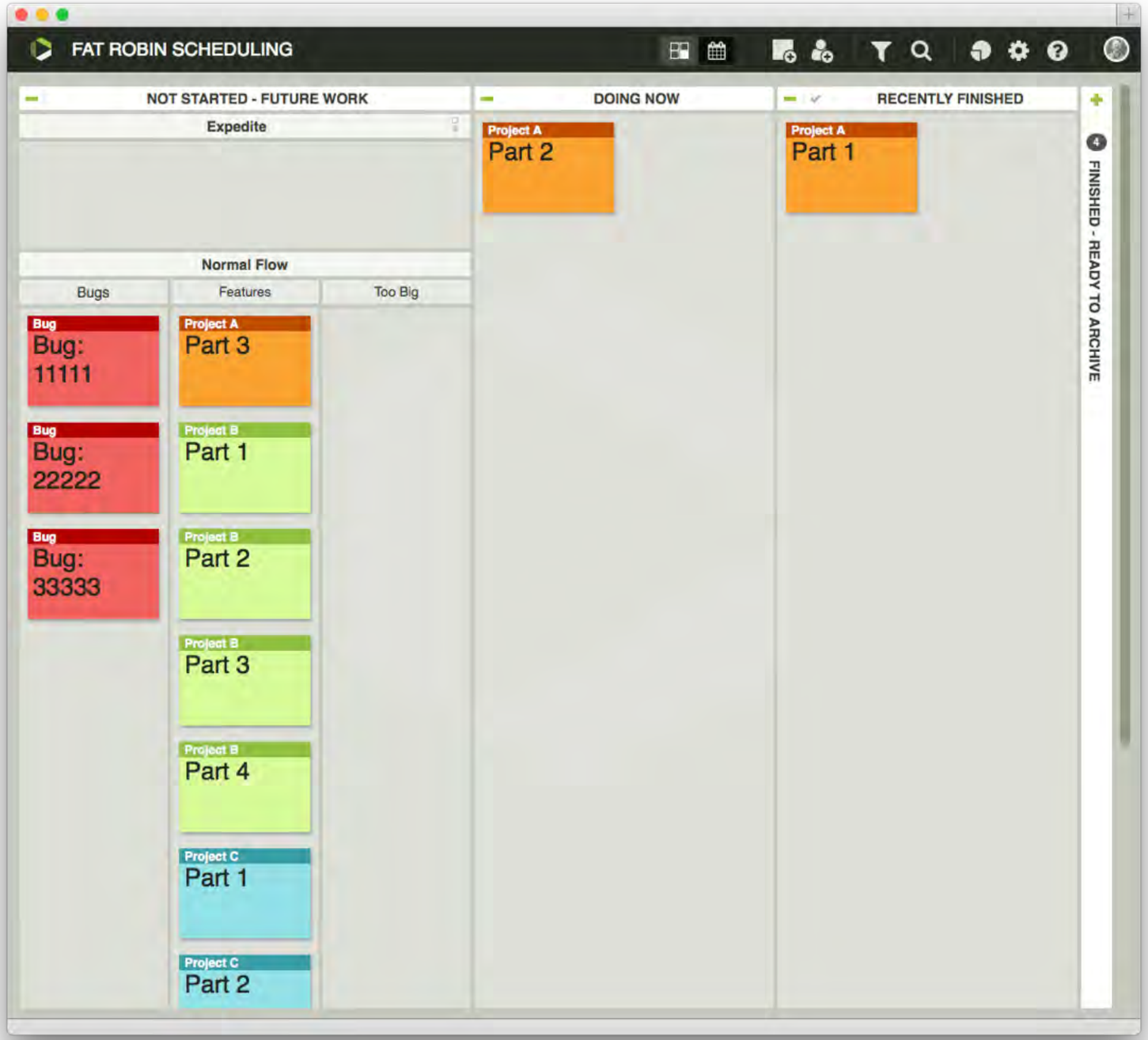
Round Robin Scheduling

leankit



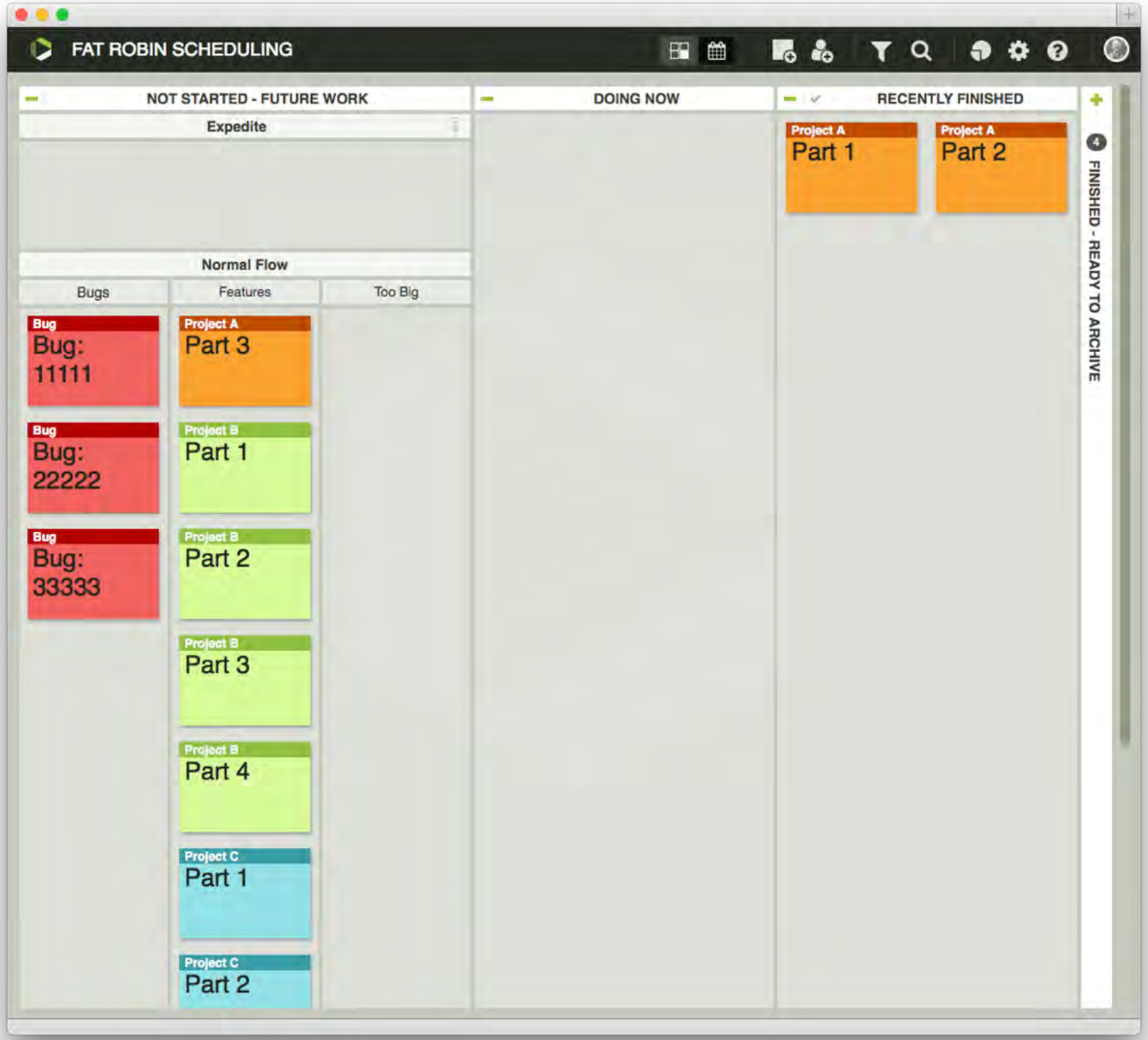
Round Robin Scheduling

leankit



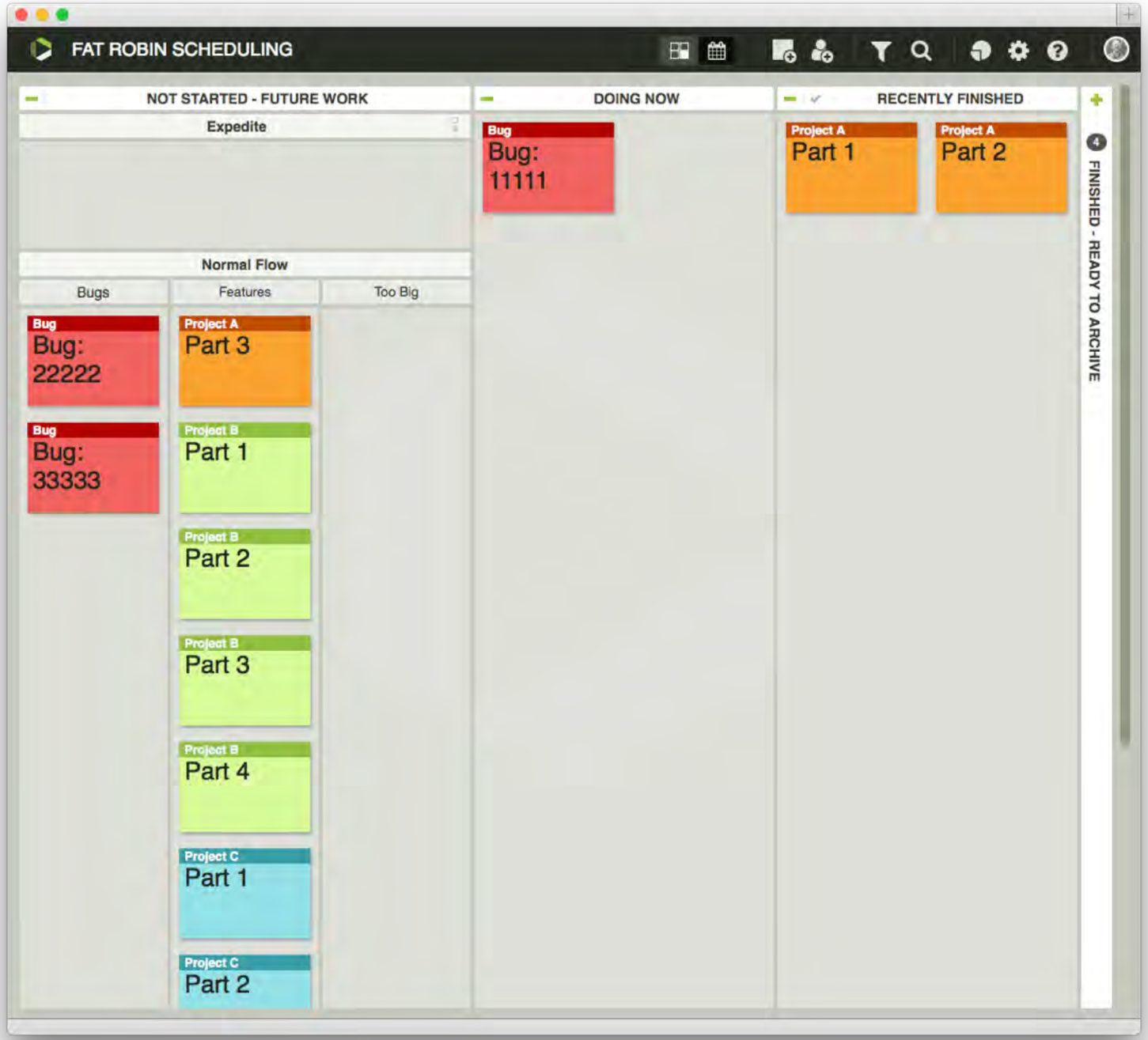
Round Robin Scheduling

leankit



Round Robin Scheduling

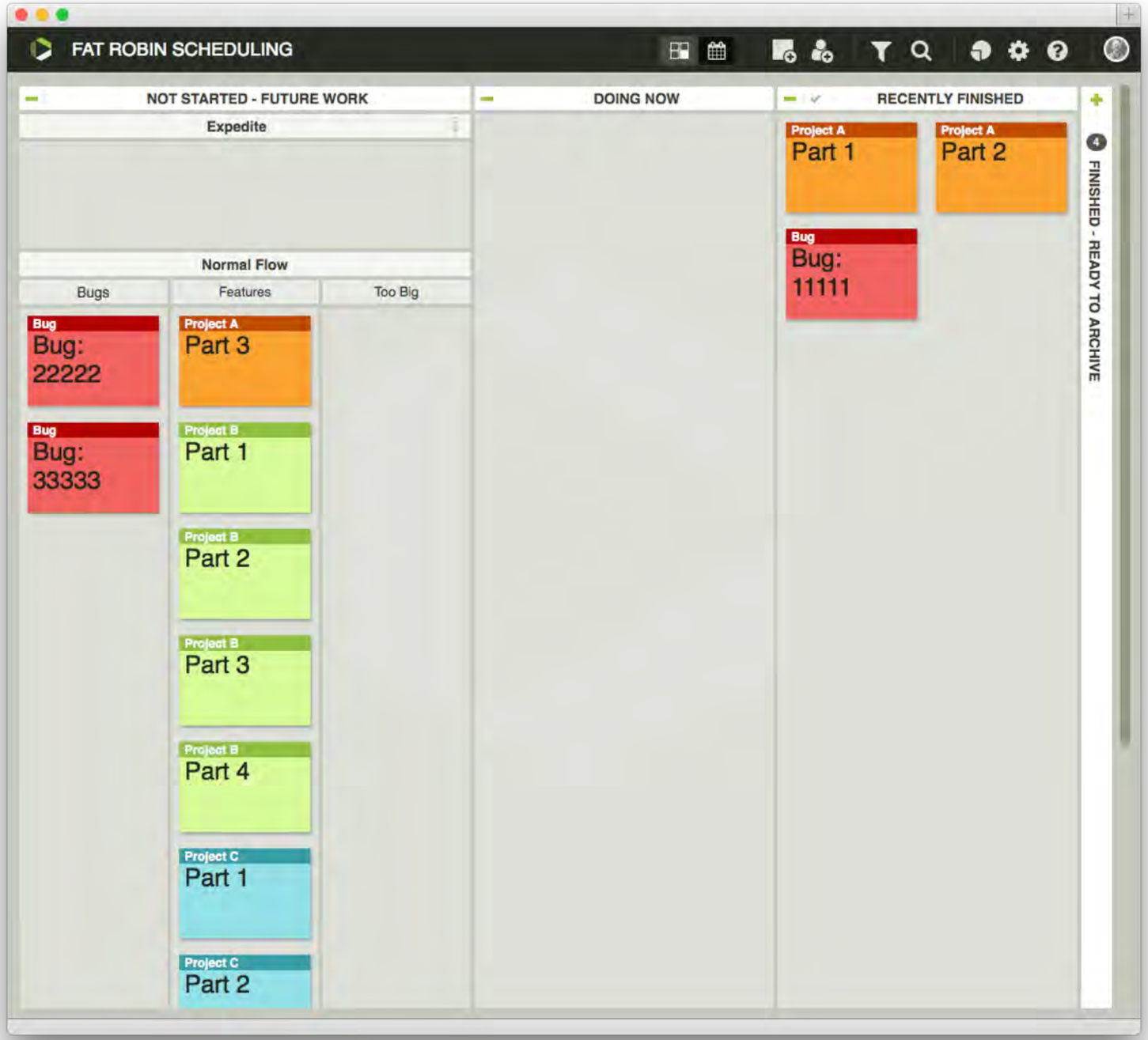
leankit



Round Robin Scheduling

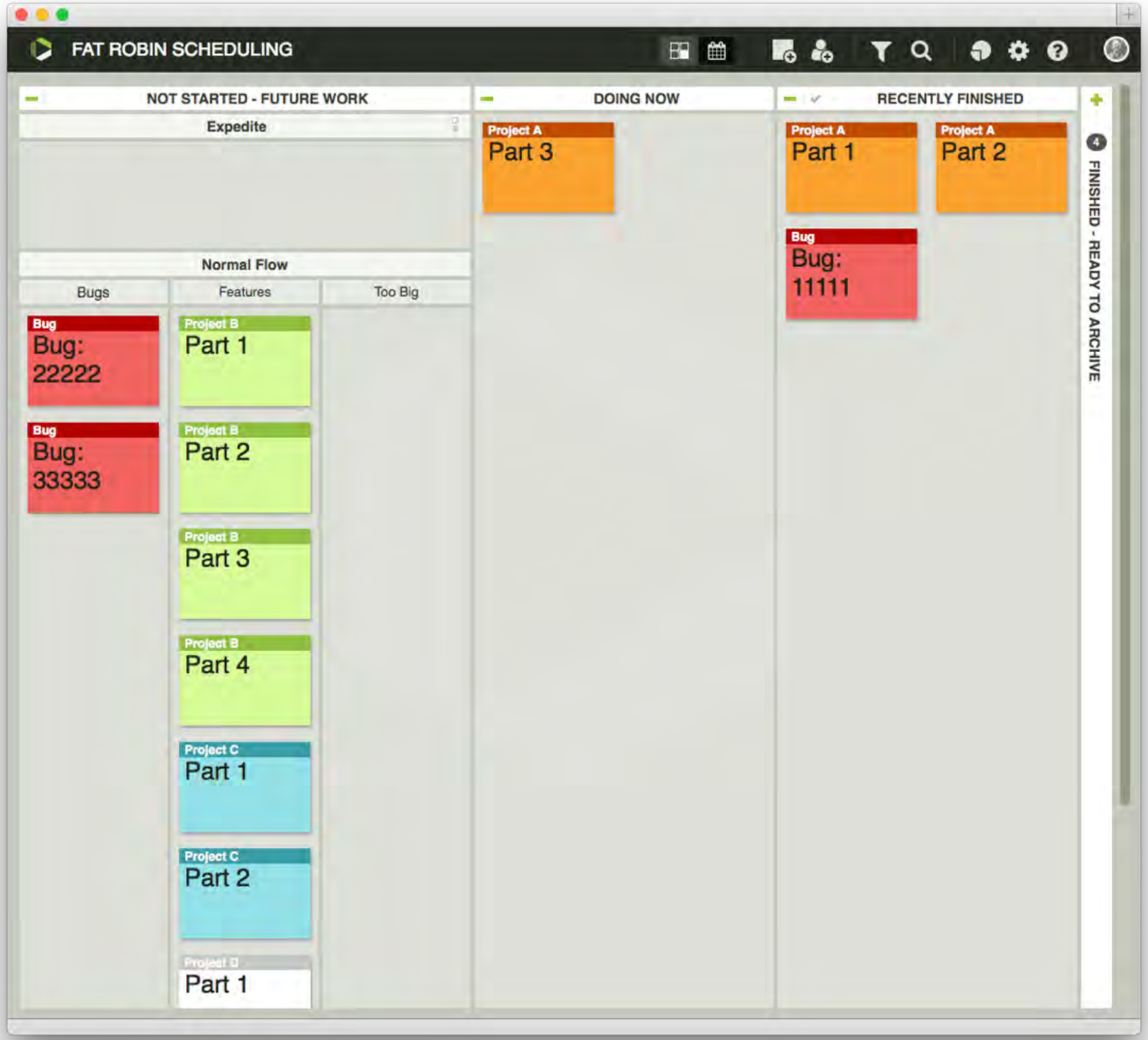
- Smaller batch size allows flexible reprioritization without context switching

leankit



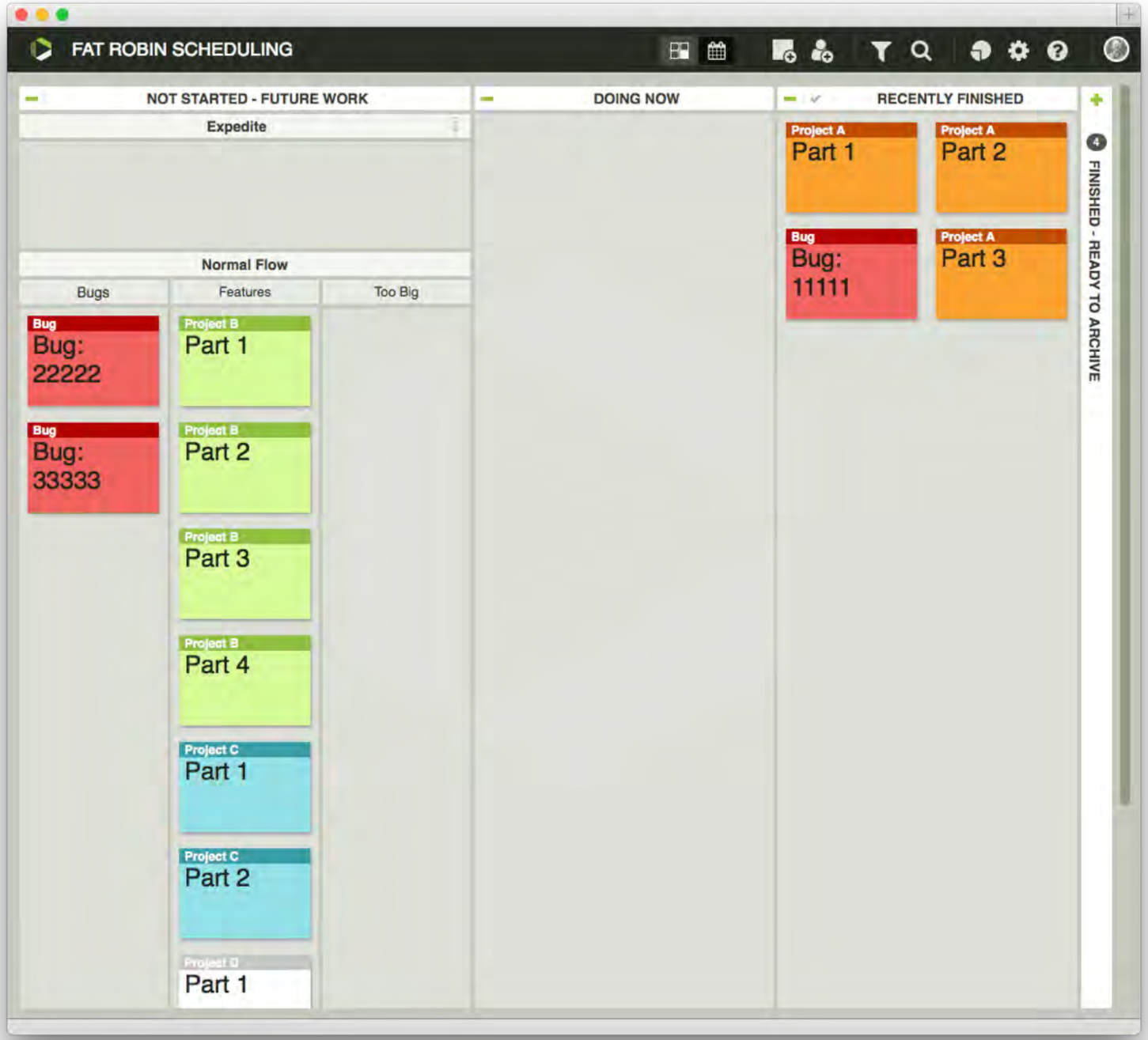
Round Robin Scheduling

leankit



Round Robin Scheduling

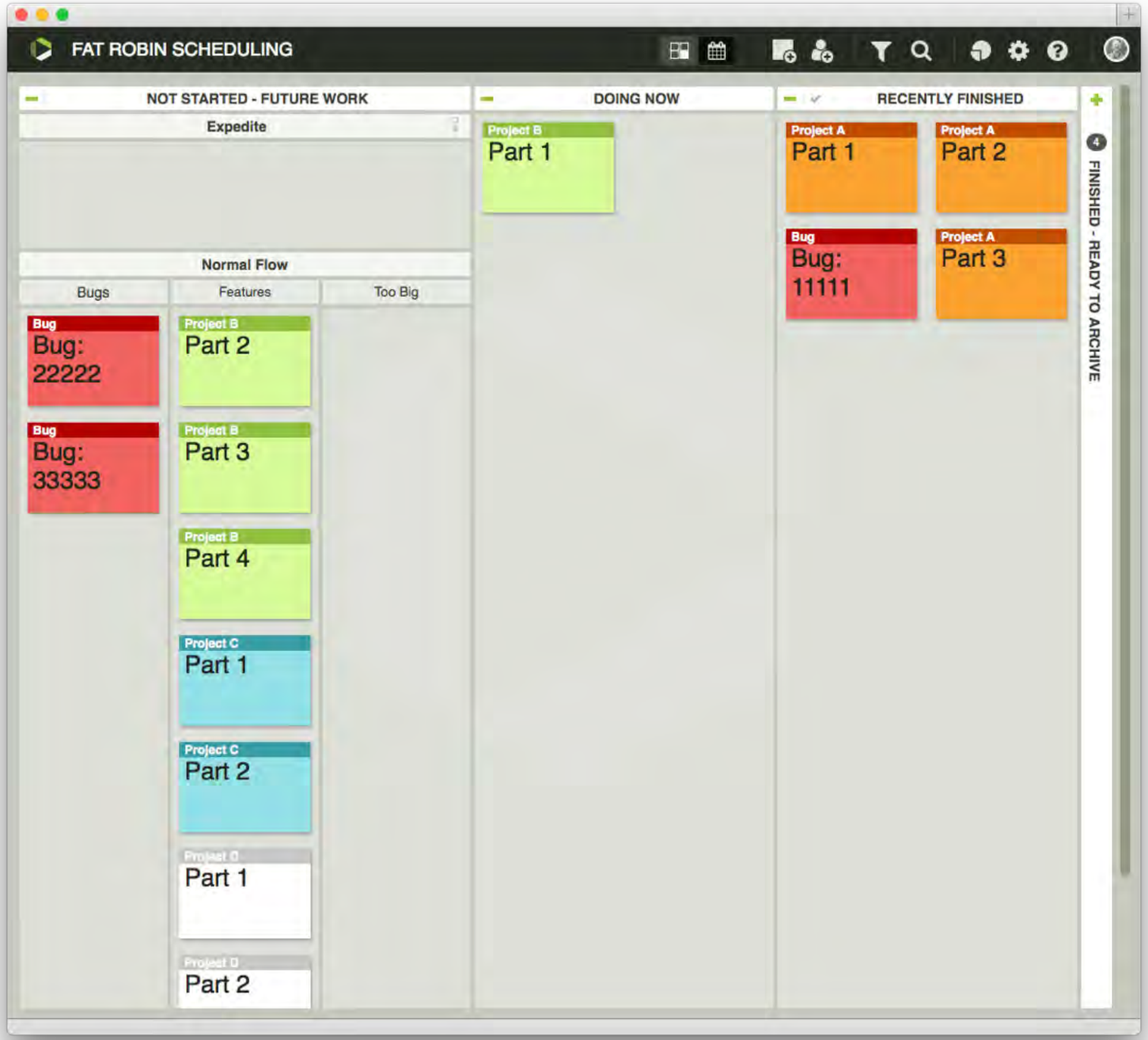
leankit



Round Robin Scheduling

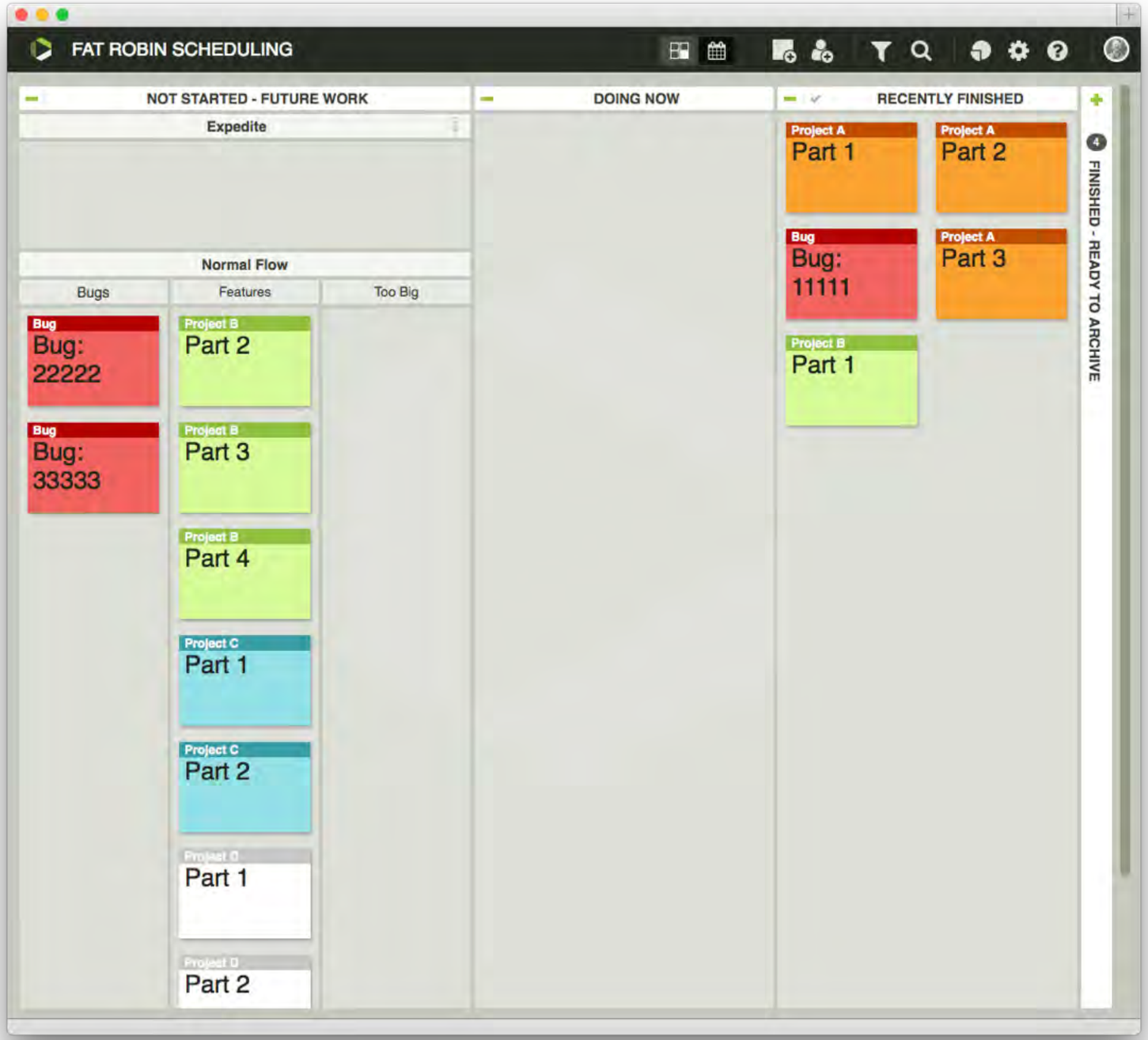
- What next?
  - Project B?
  - Bugs?
  - Maybe Project C?
  - Your choice!

leankit



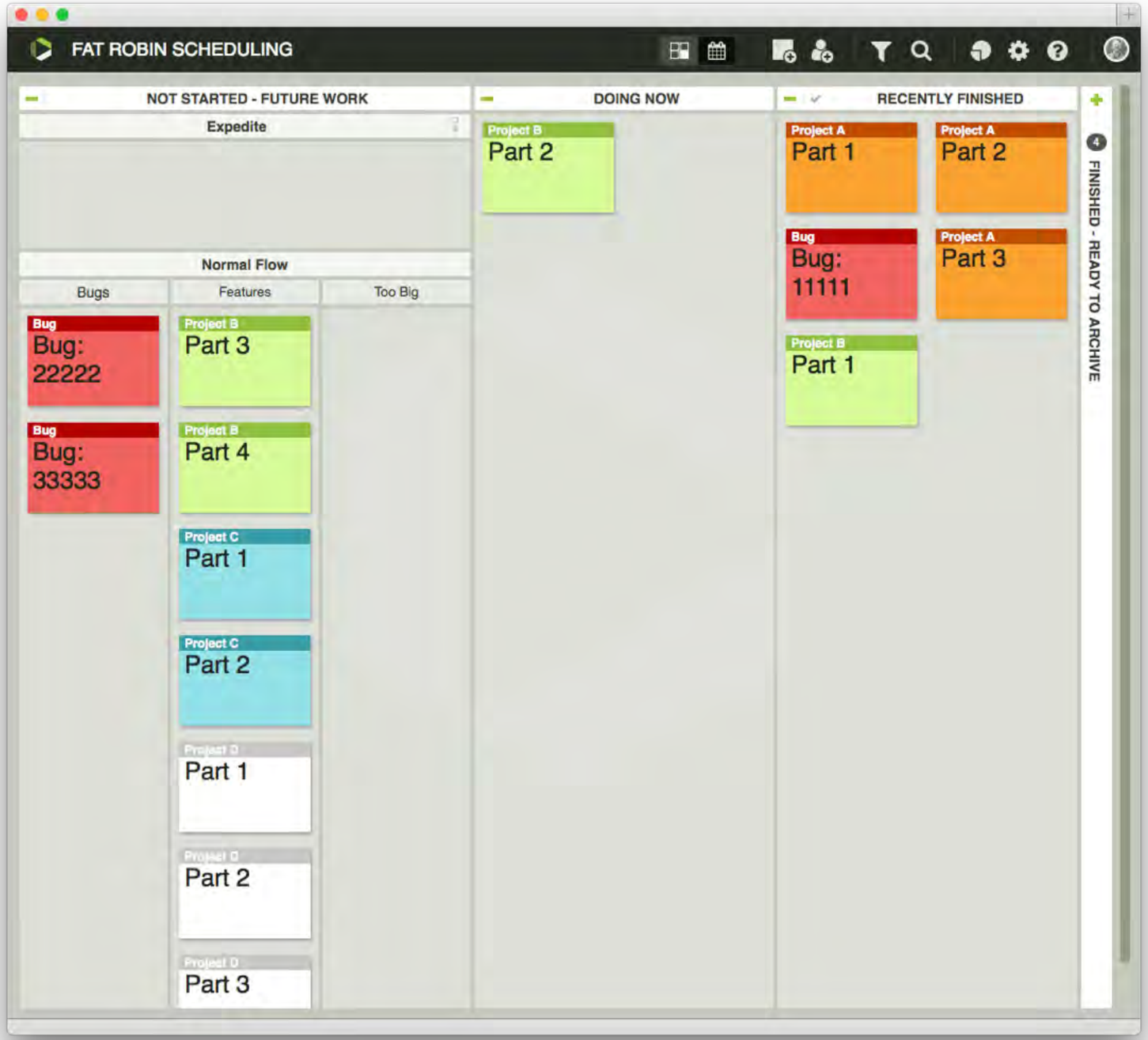
Round Robin Scheduling

leankit



Round Robin Scheduling

leankit



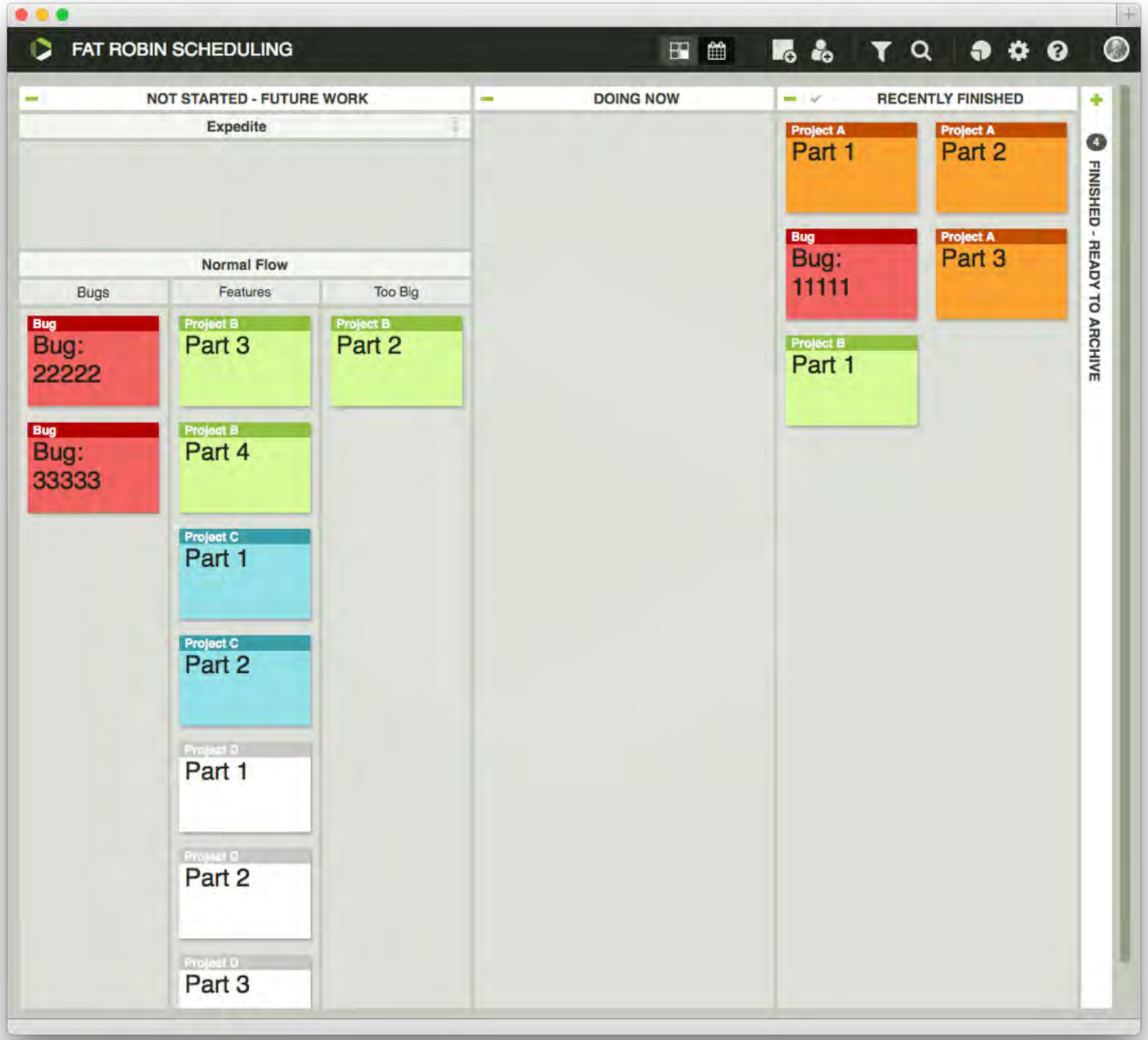
Round Robin Scheduling

- Oh no. Project B Part 2 is taking too long.
- Too big!
- Too fat!

leankit

Stop Digging

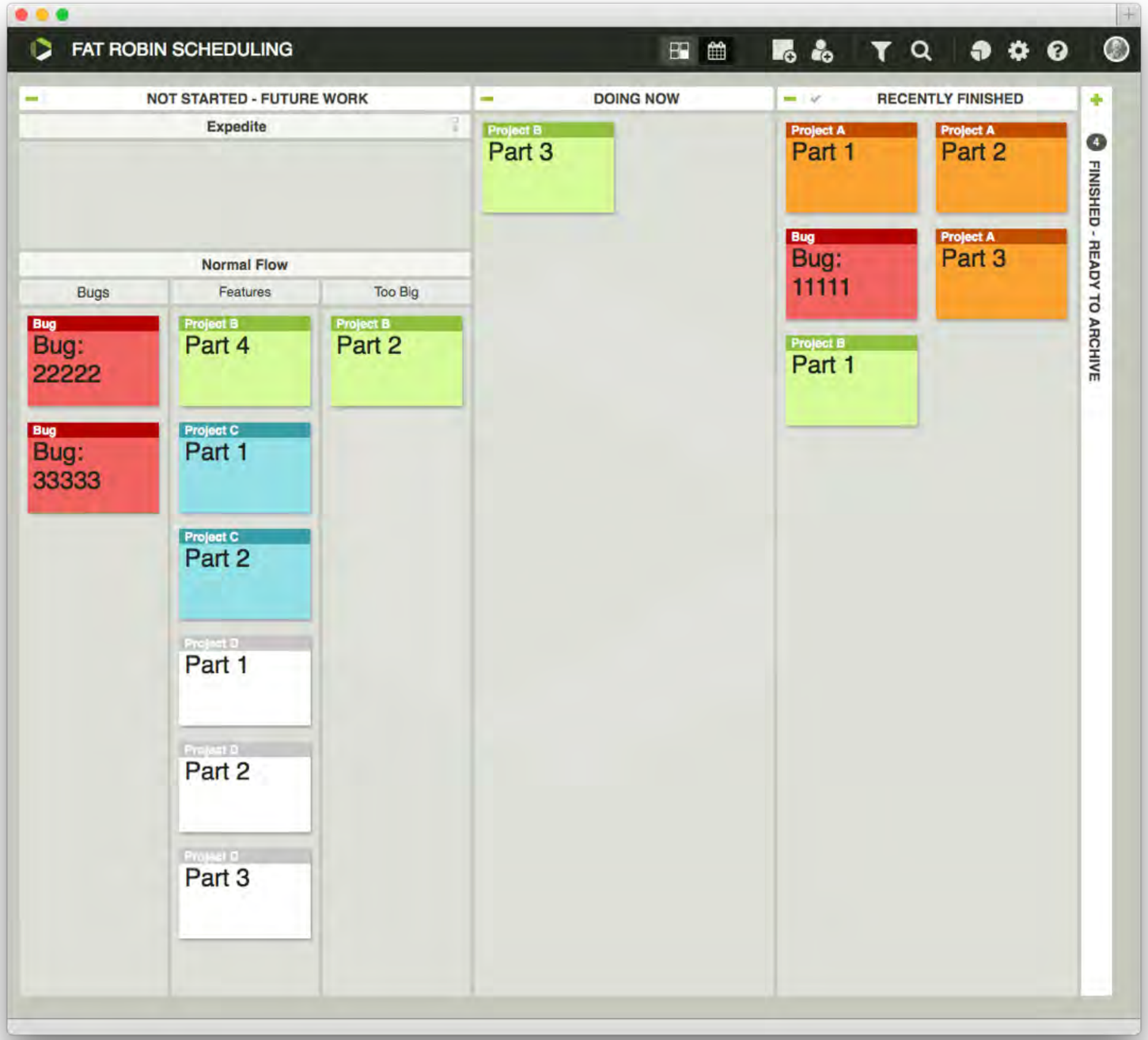
leankit



~~Round~~ Fat Robin Scheduling

- Move it aside
- Choose next priority

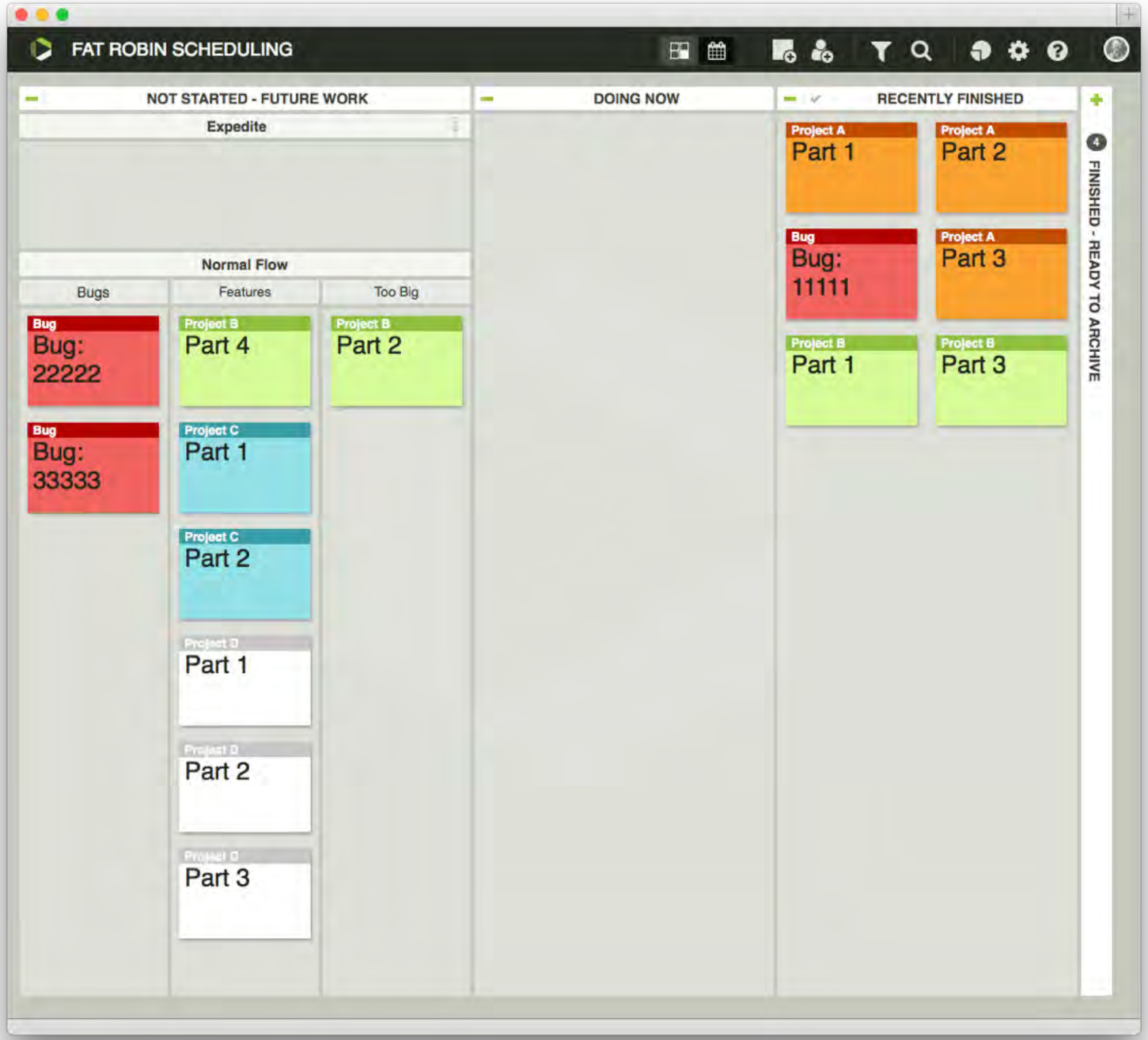
leankit



Fat Robin Scheduling

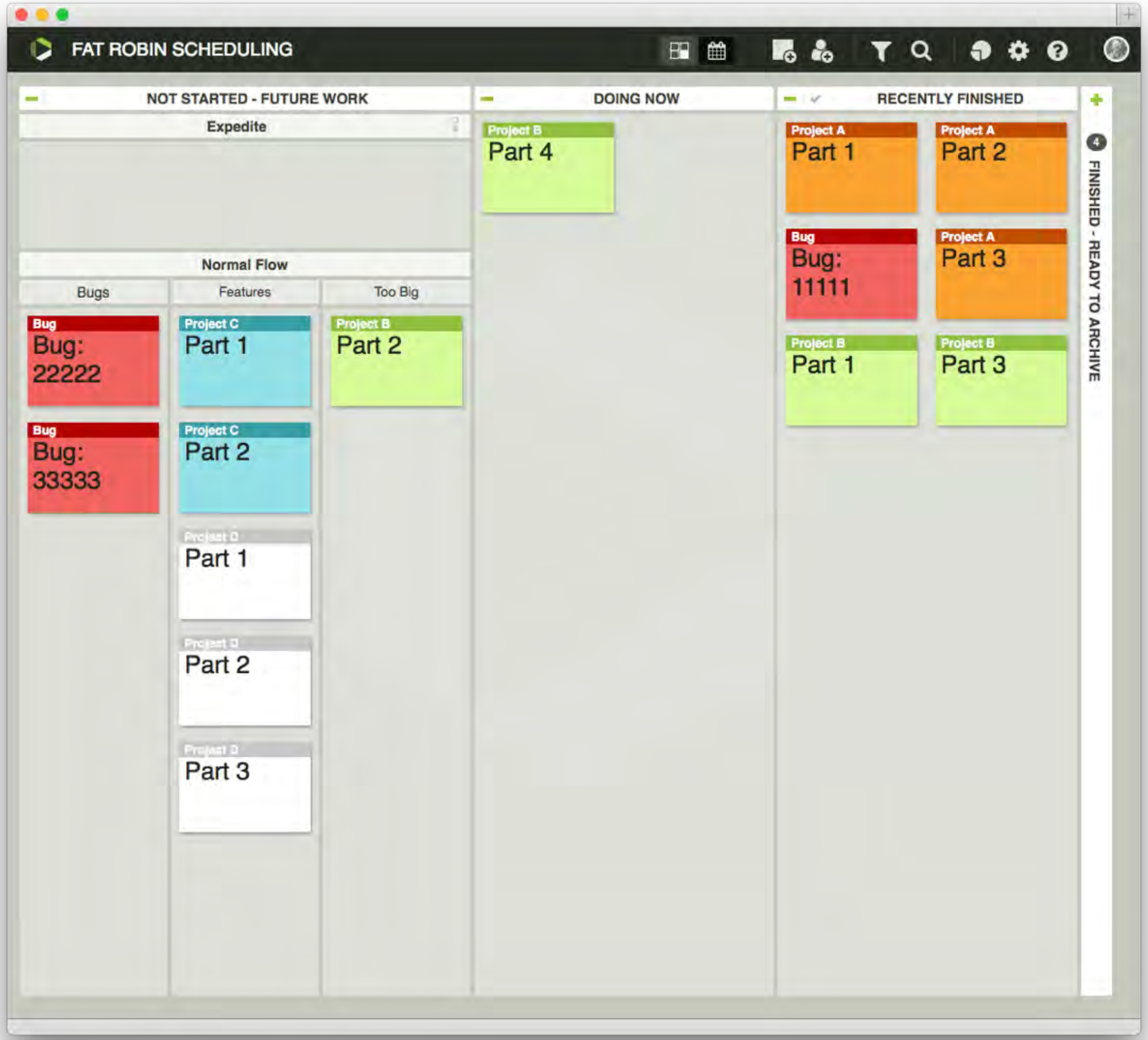
- Small & Decoupled allow us to complete, and gain value from, other portions of the project

leankit



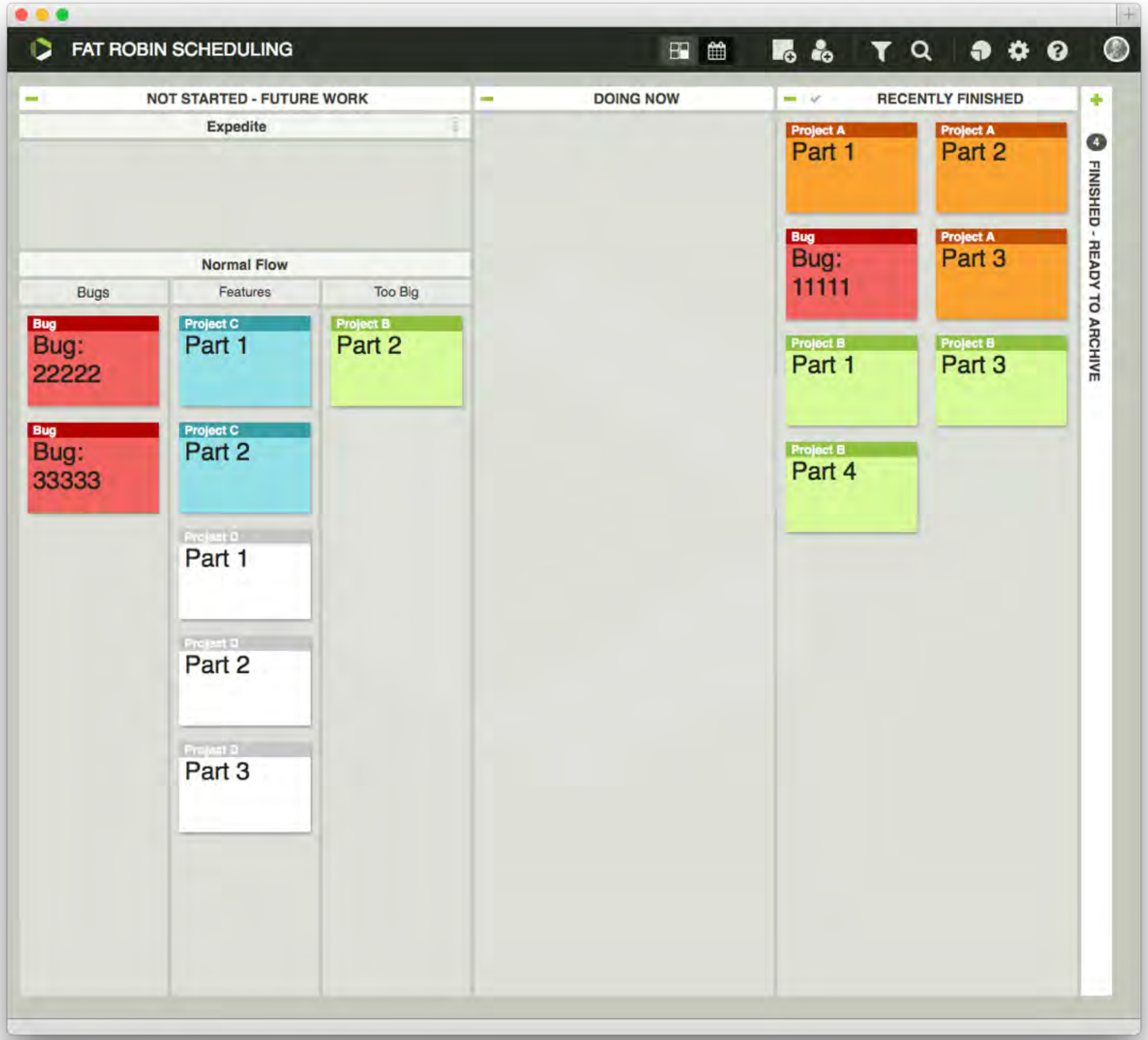
Fat Robin Scheduling

leankit



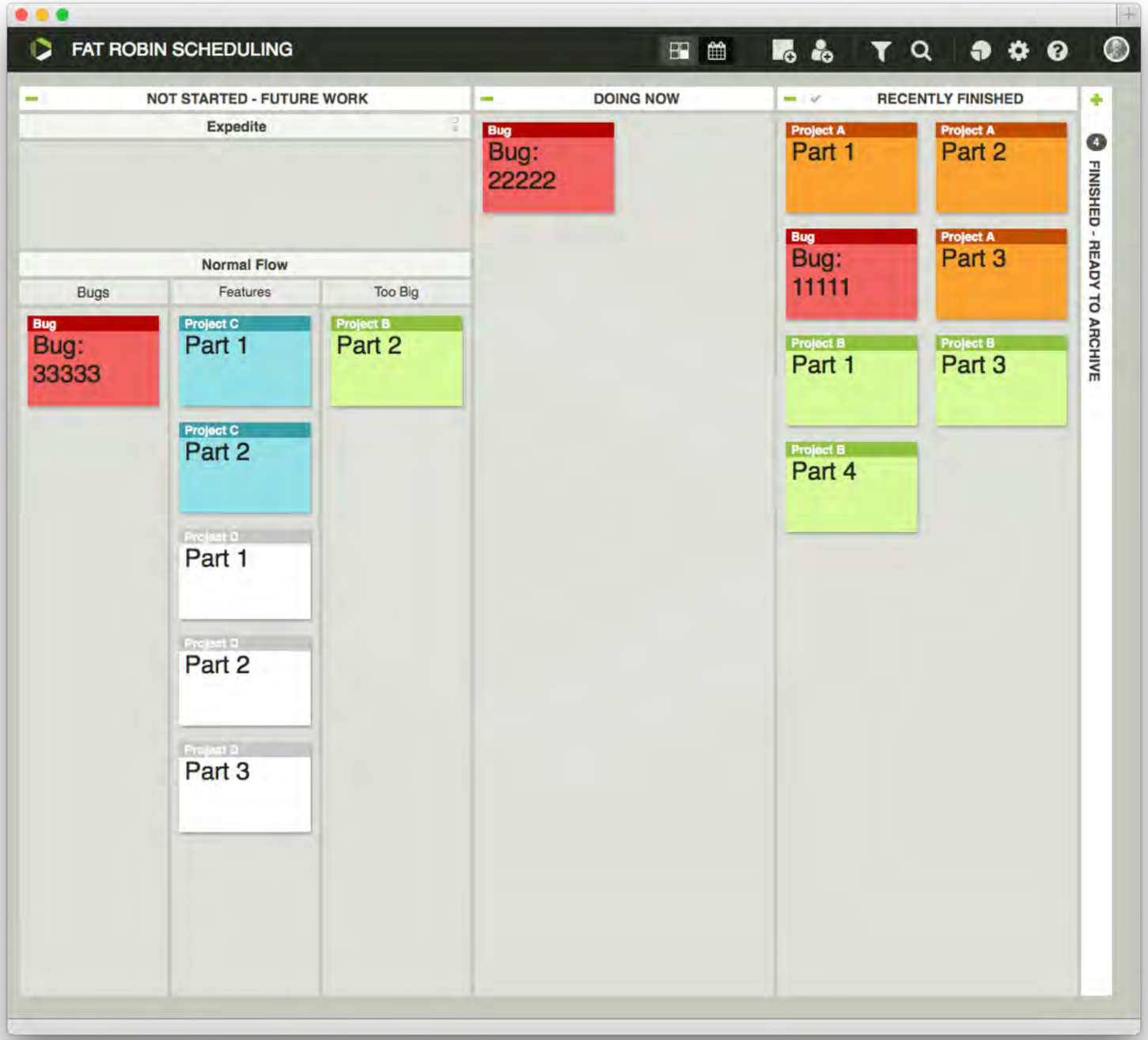
Fat Robin Scheduling

leankit



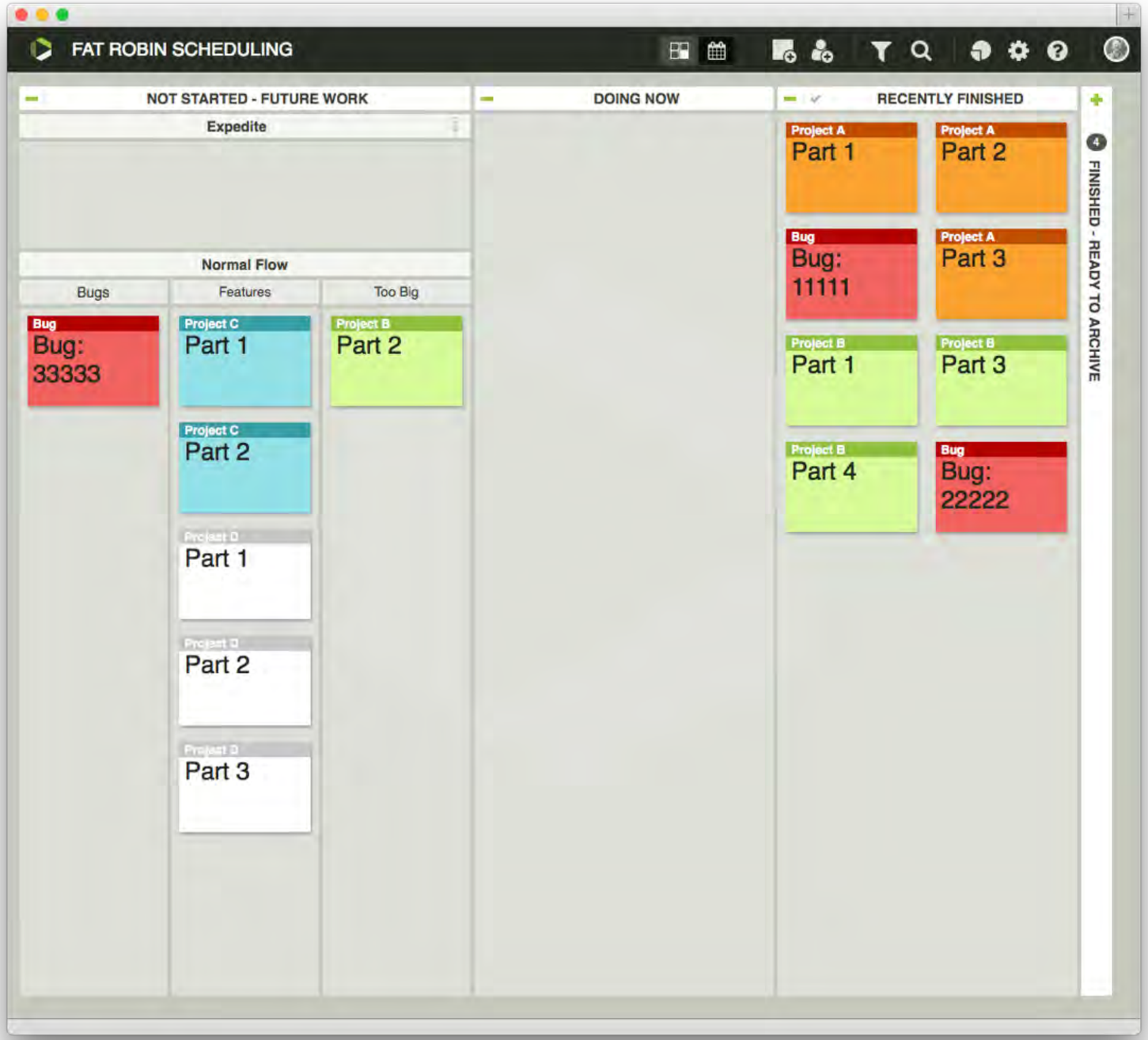
Fat Robin Scheduling

leankit



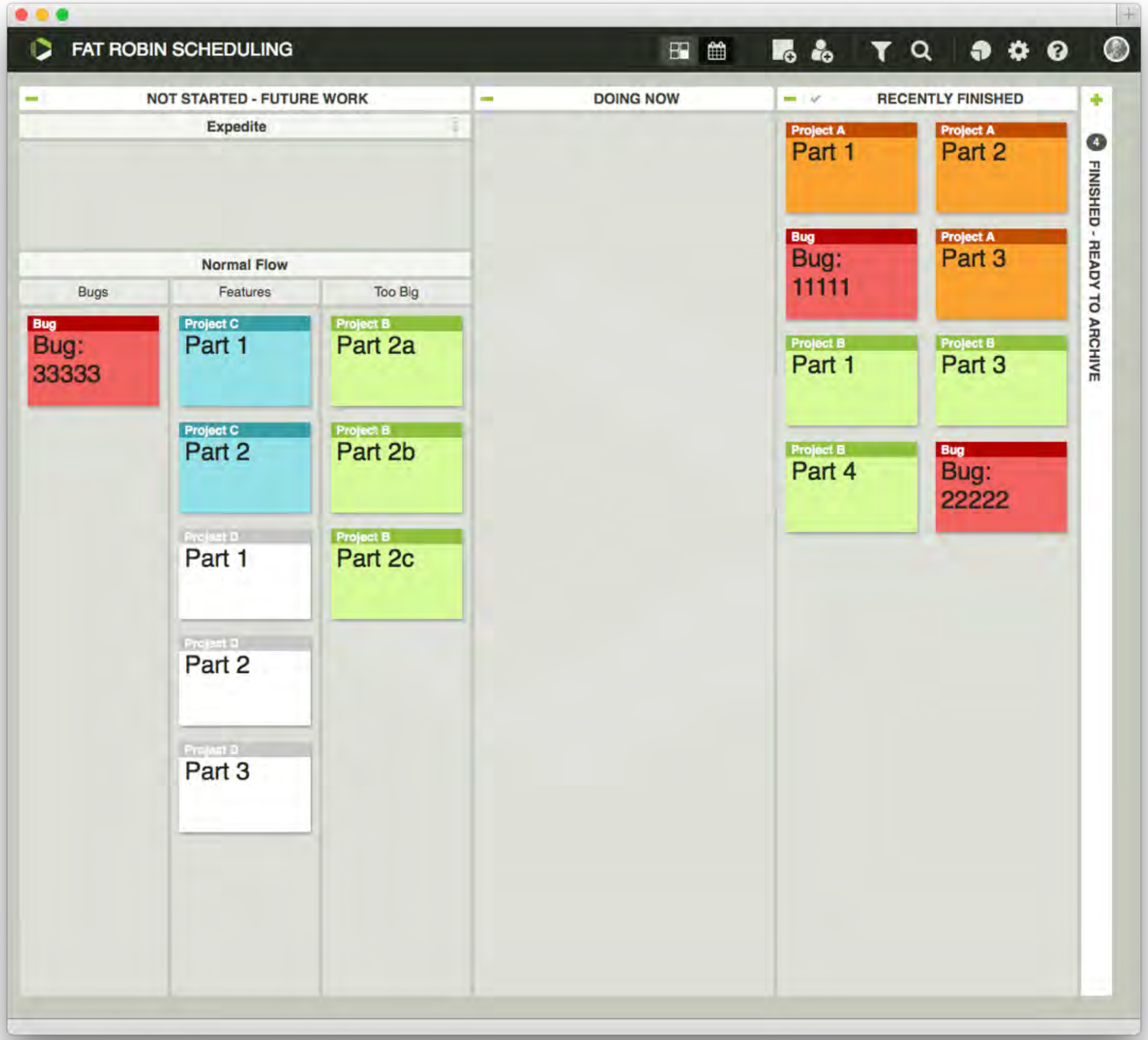
Fat Robin Scheduling

leankit



Fat Robin Scheduling

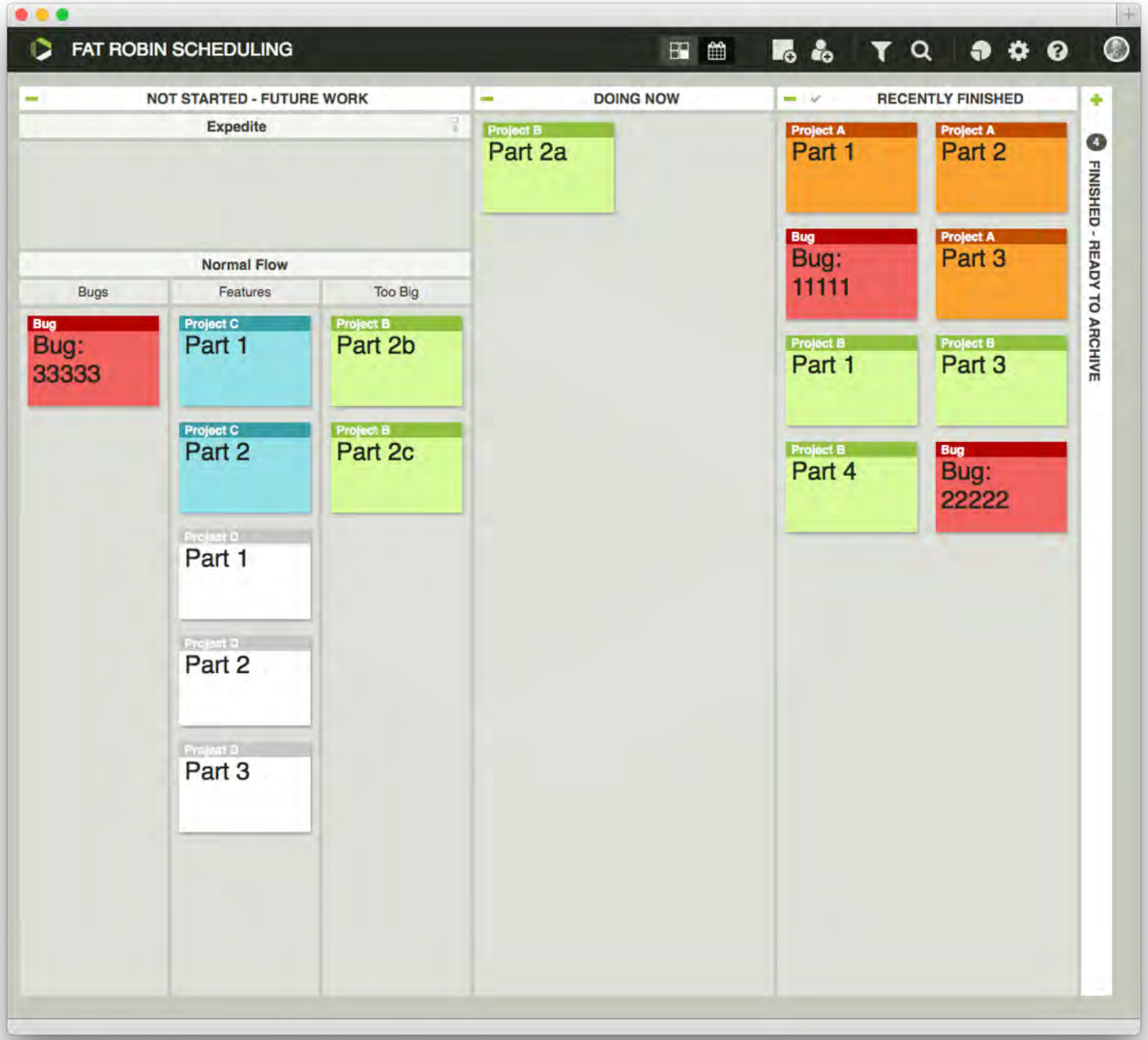
leankit



Fat Robin Scheduling

- Back to that fat robin
- Break him up into smaller pieces

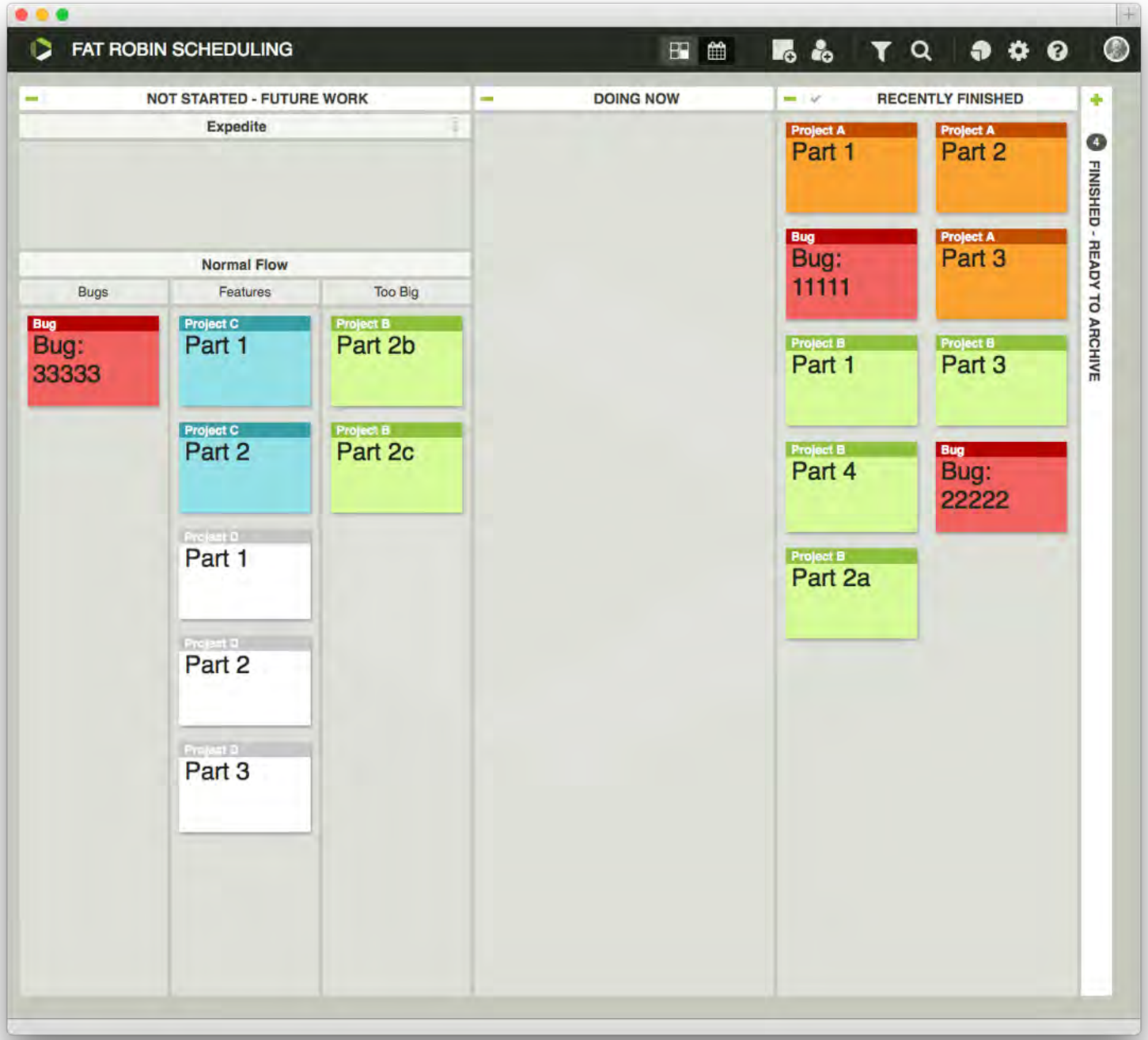
leankit



Fat Robin Scheduling

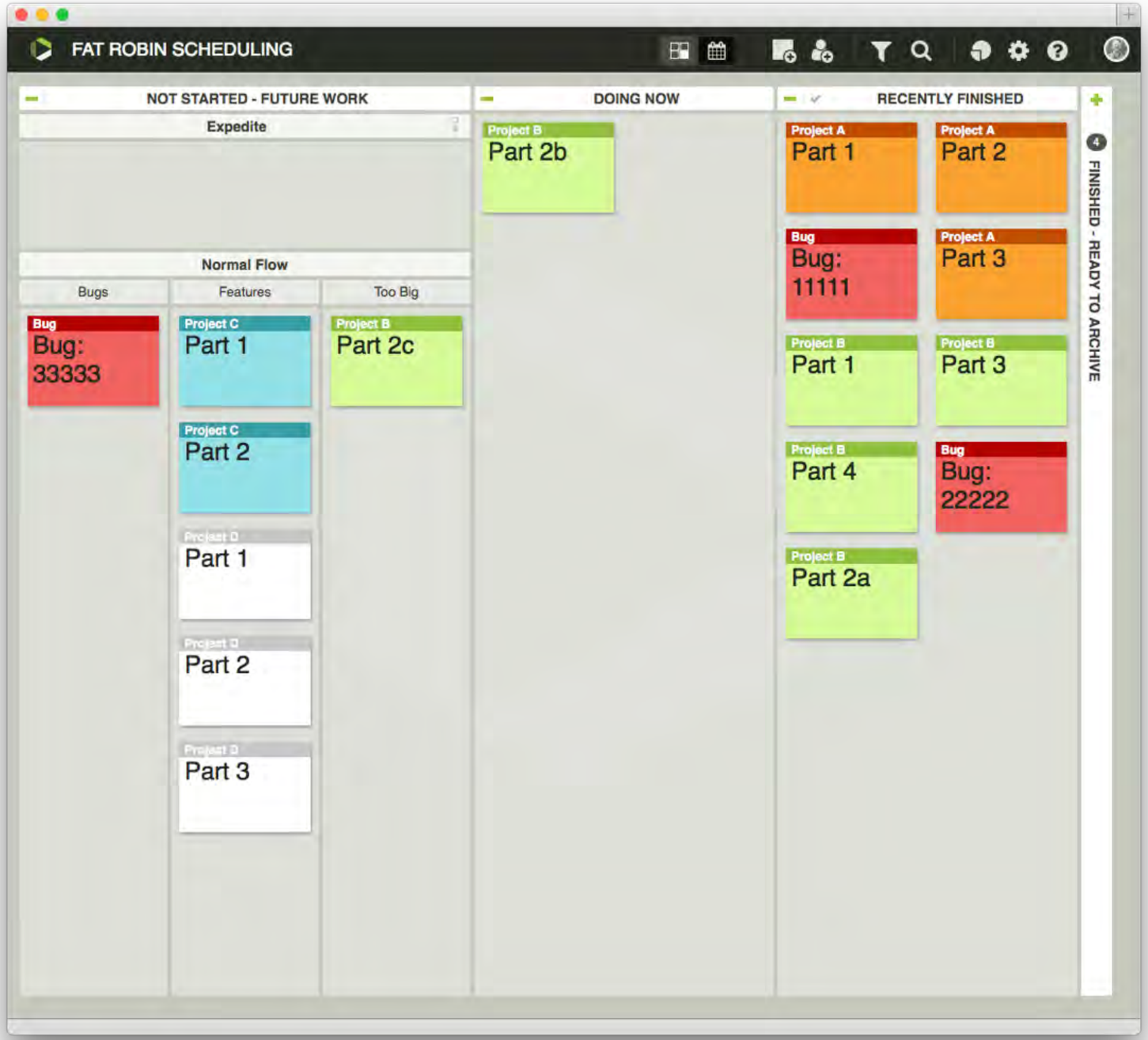
- Flow as usual

leankit



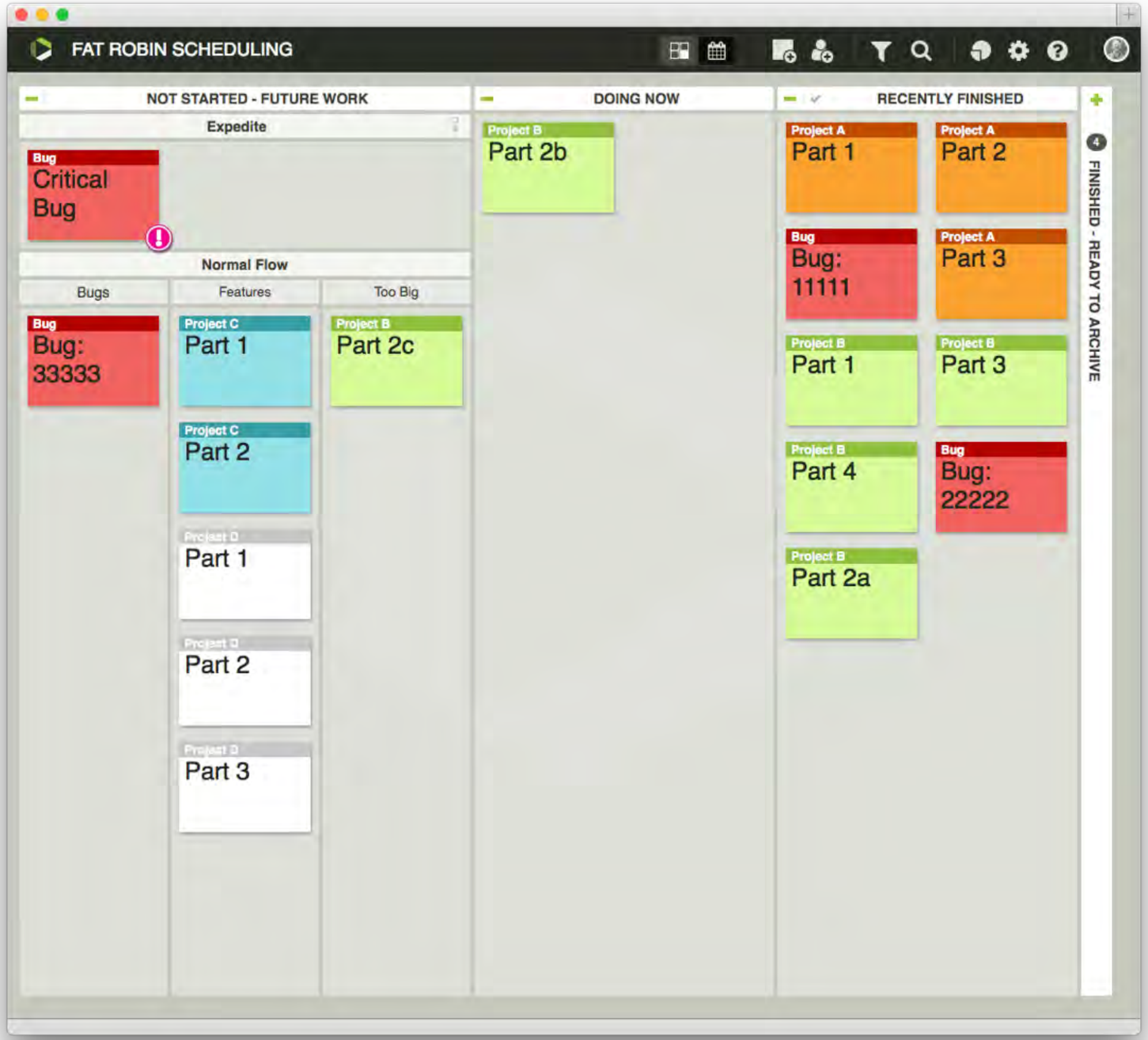
Fat Robin Scheduling

leankit



Fat Robin Scheduling

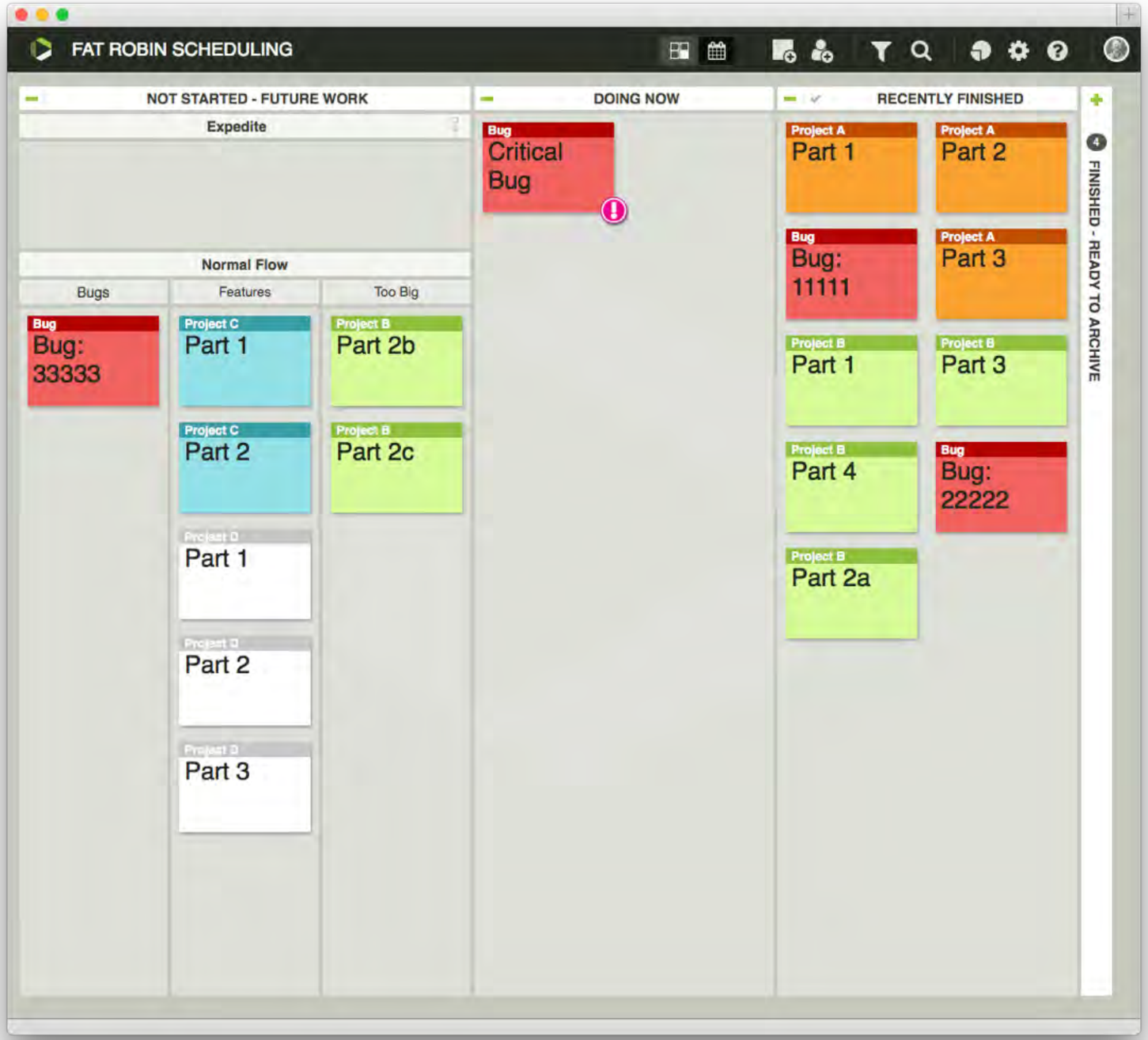
leankit



Fat Robin Scheduling

- What about critical issues?

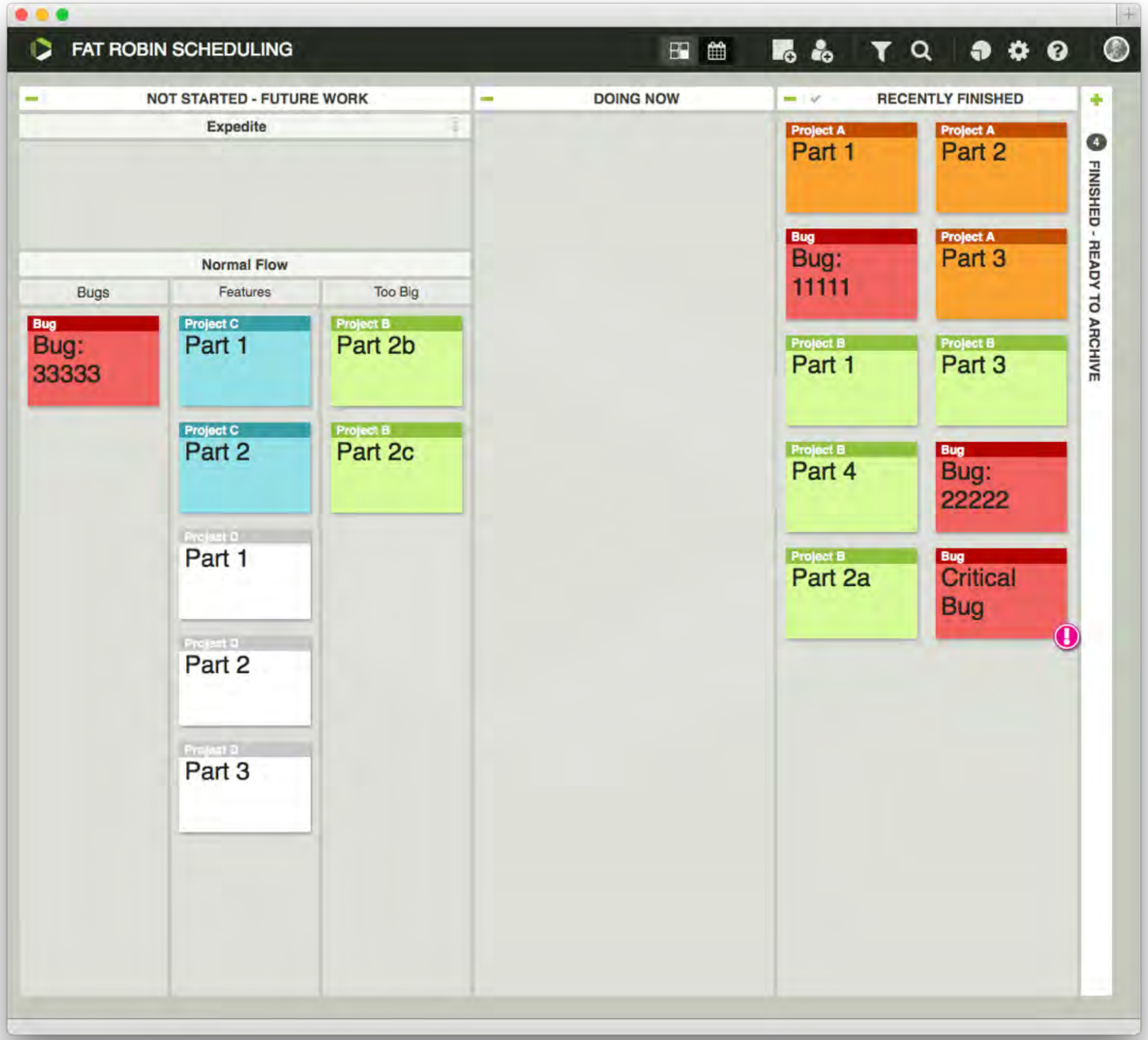
leankit



Fat Robin Scheduling

- Stop the line
- Tools down on other work
- Do not simply add more work to the team

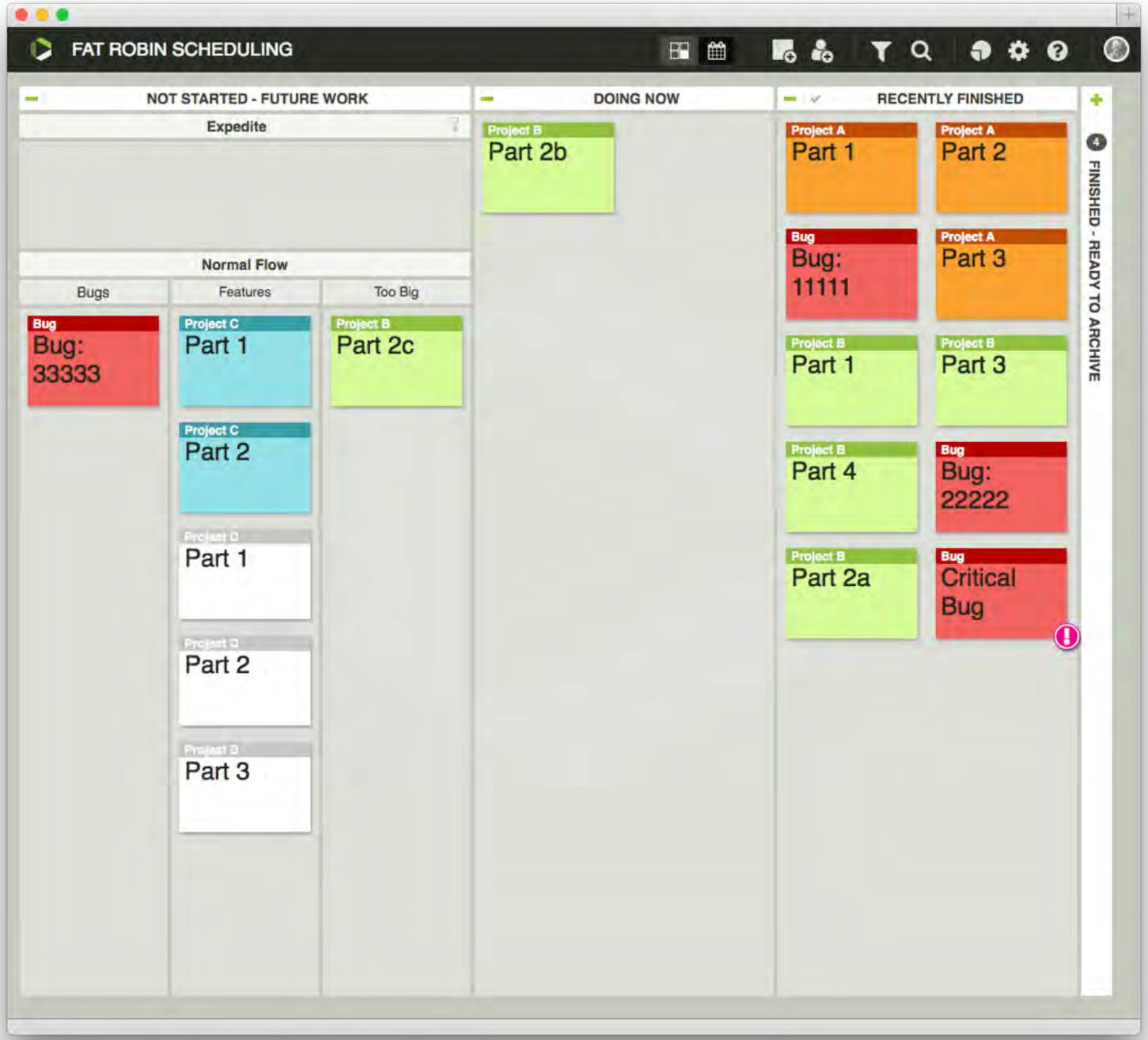
leankit



Fat Robin Scheduling

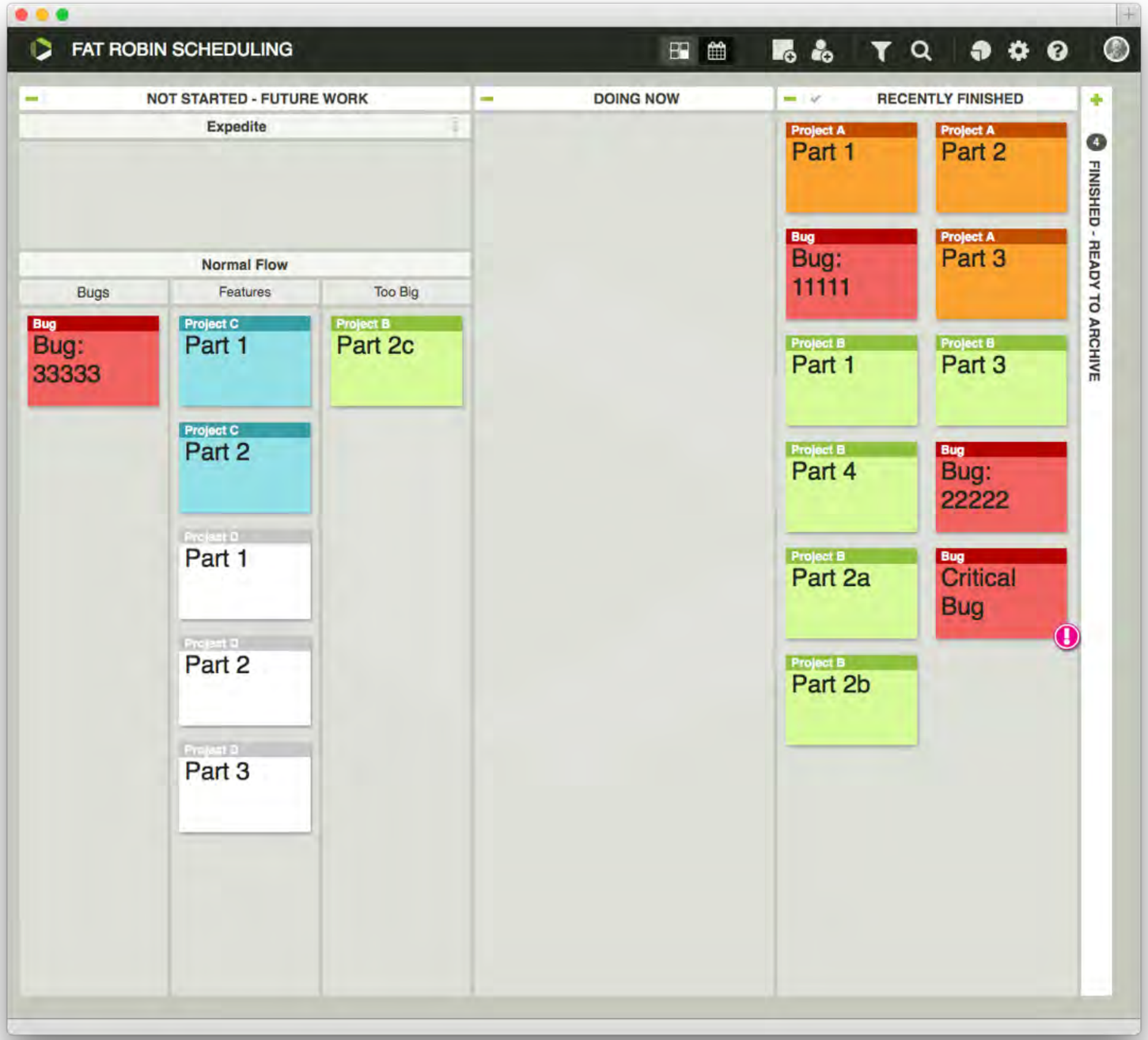
- Critical issue resolved, back to usual flow

leankit



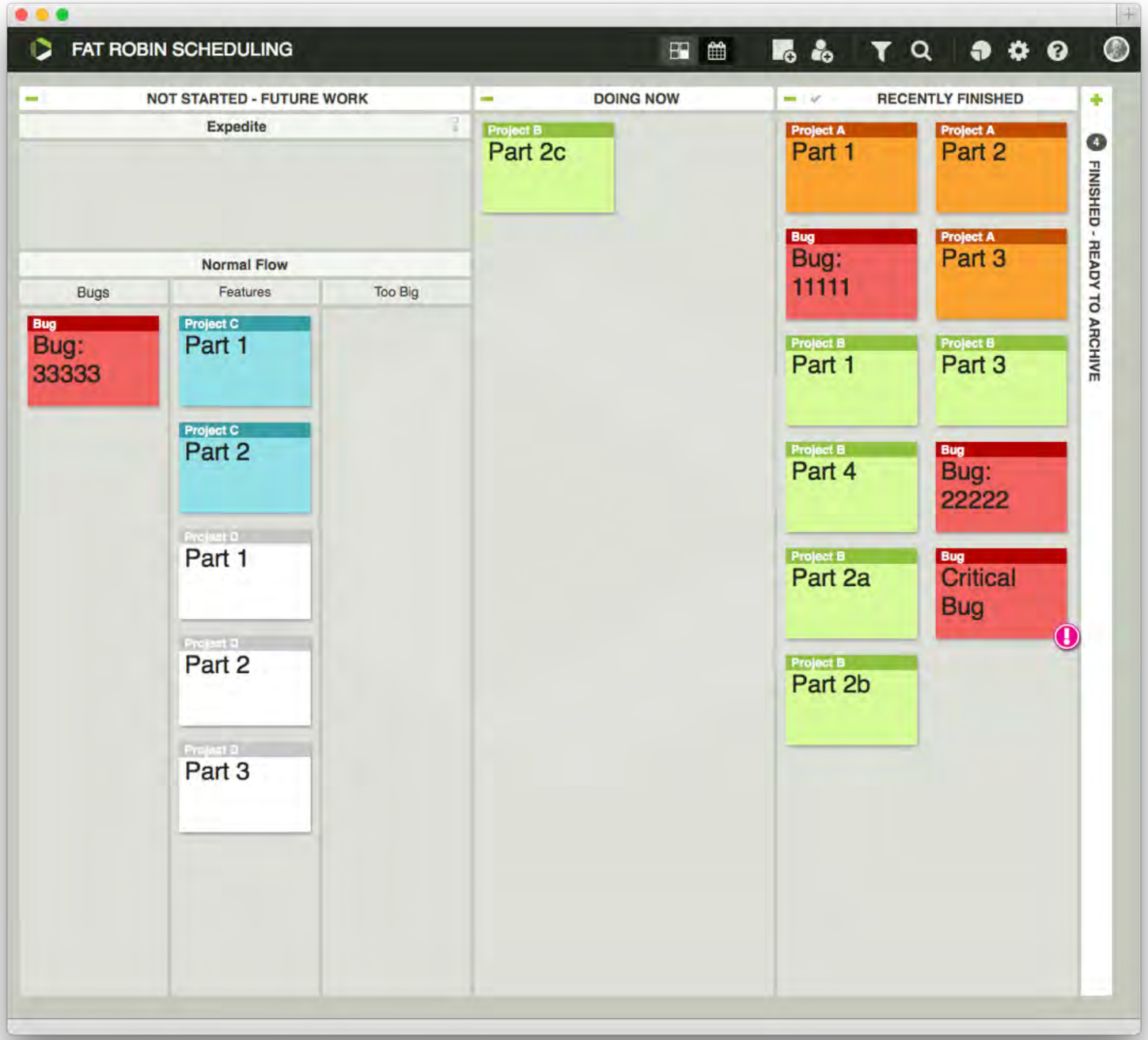
Fat Robin Scheduling

leankit



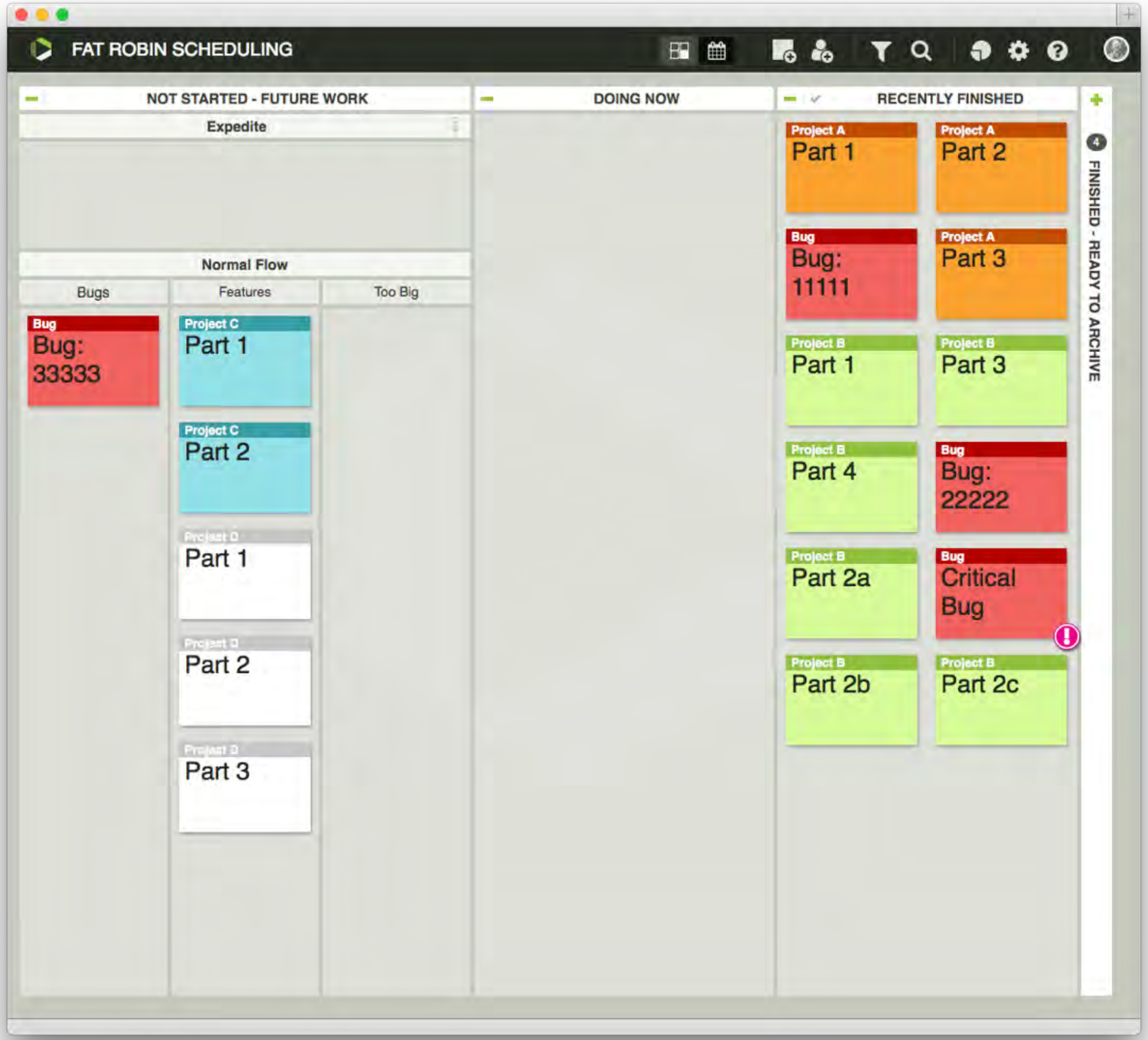
Fat Robin Scheduling

leankit



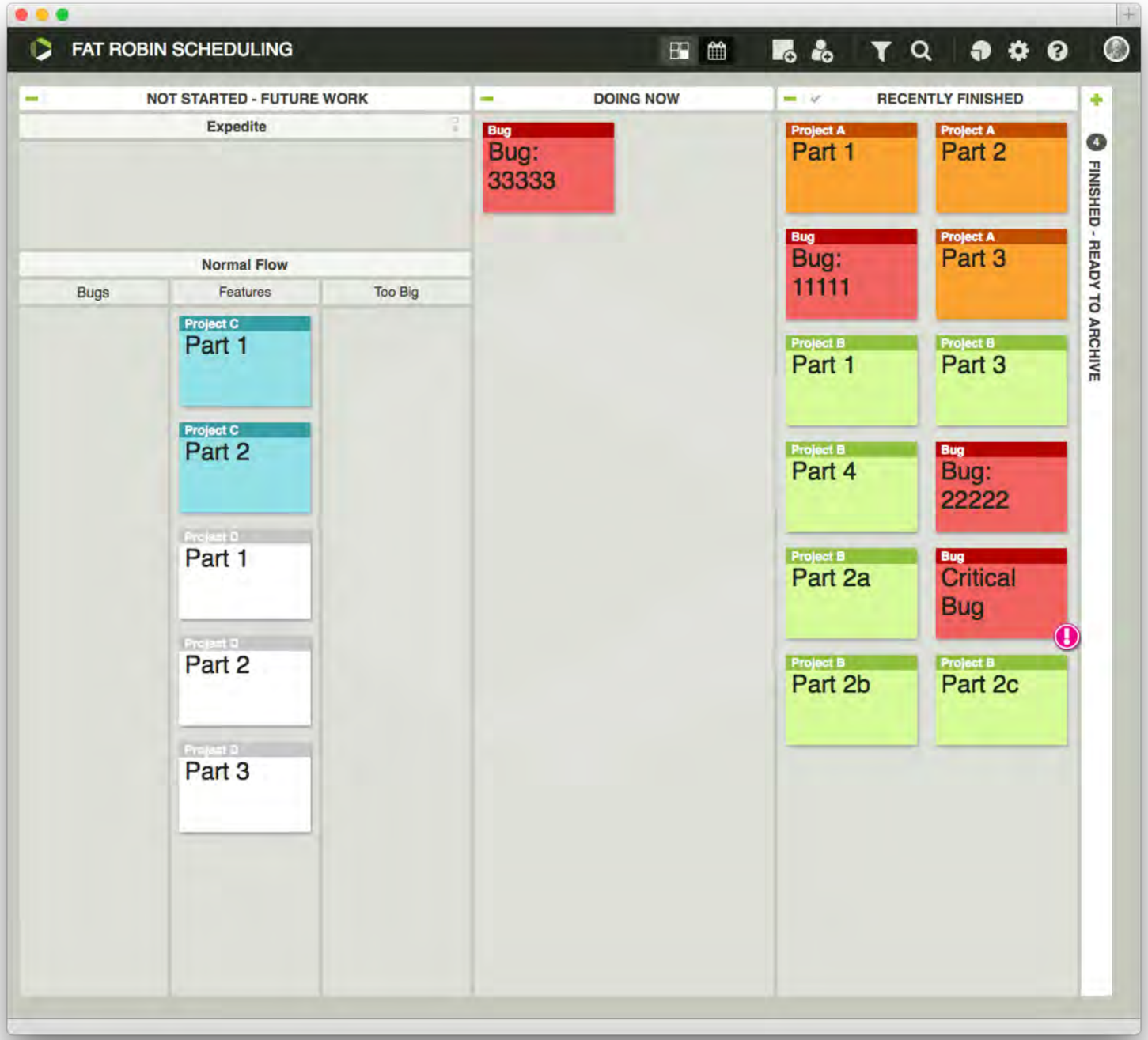
Fat Robin Scheduling

leankit



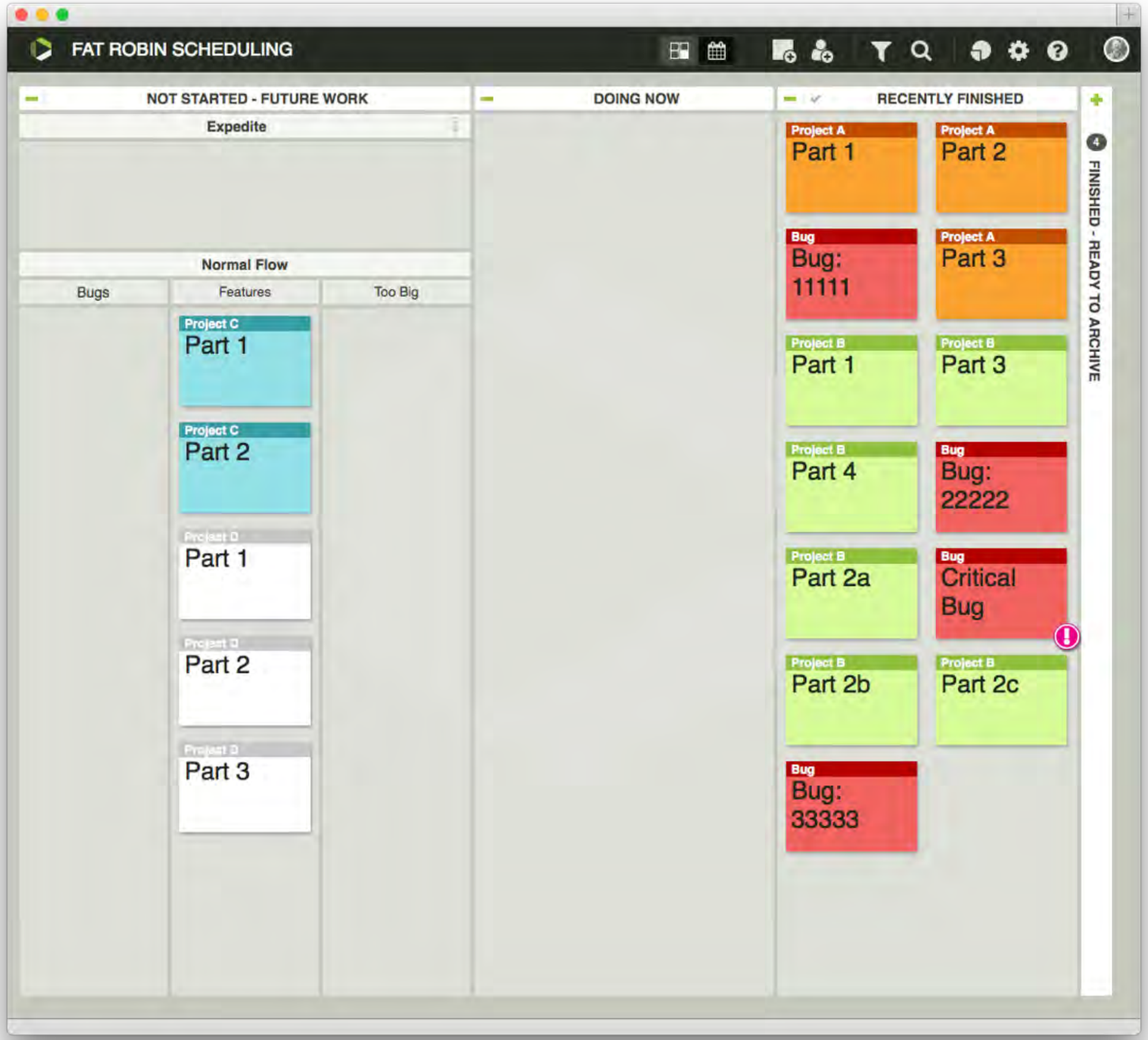
Fat Robin Scheduling

leankit



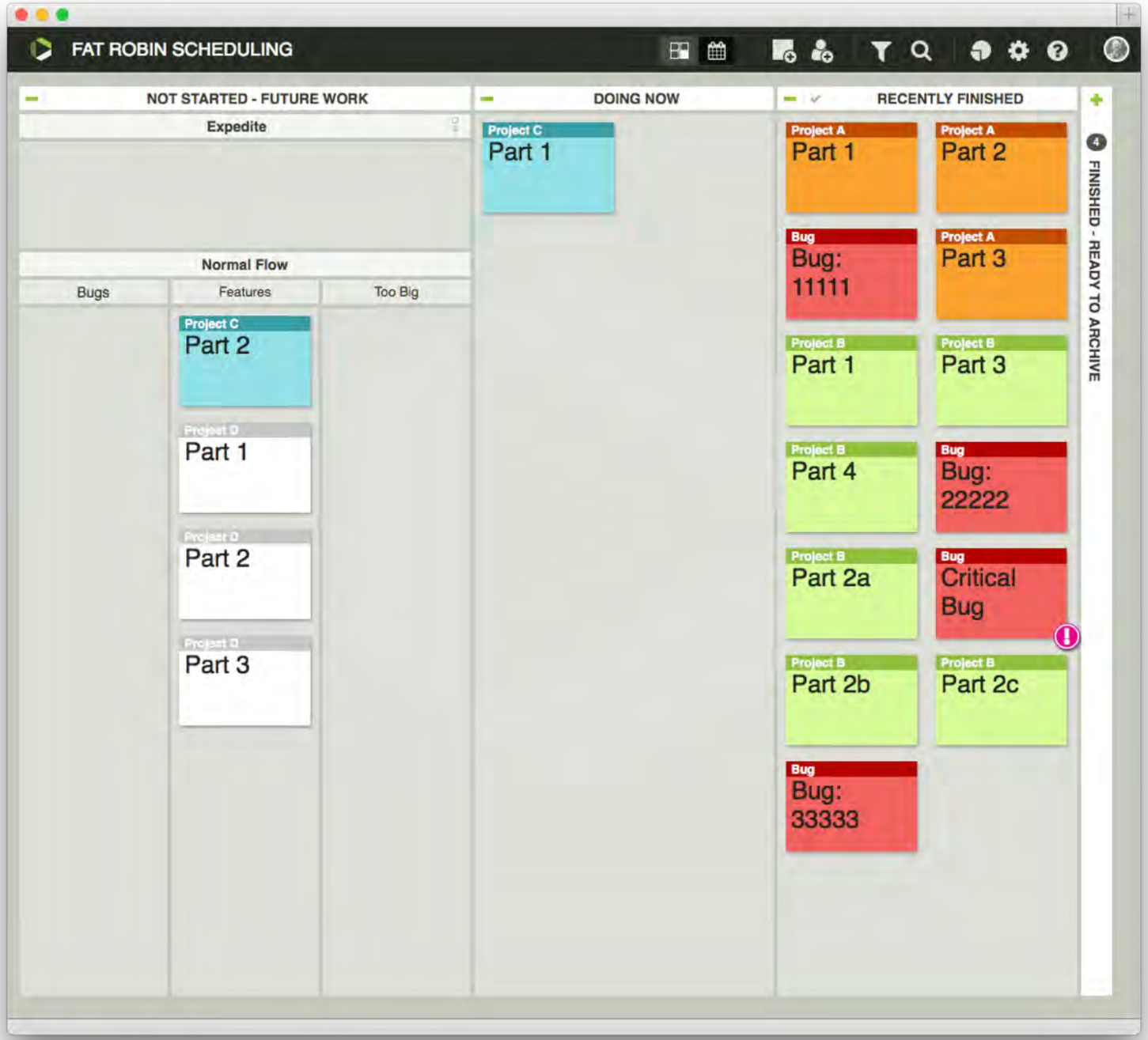
Fat Robin Scheduling

leankit



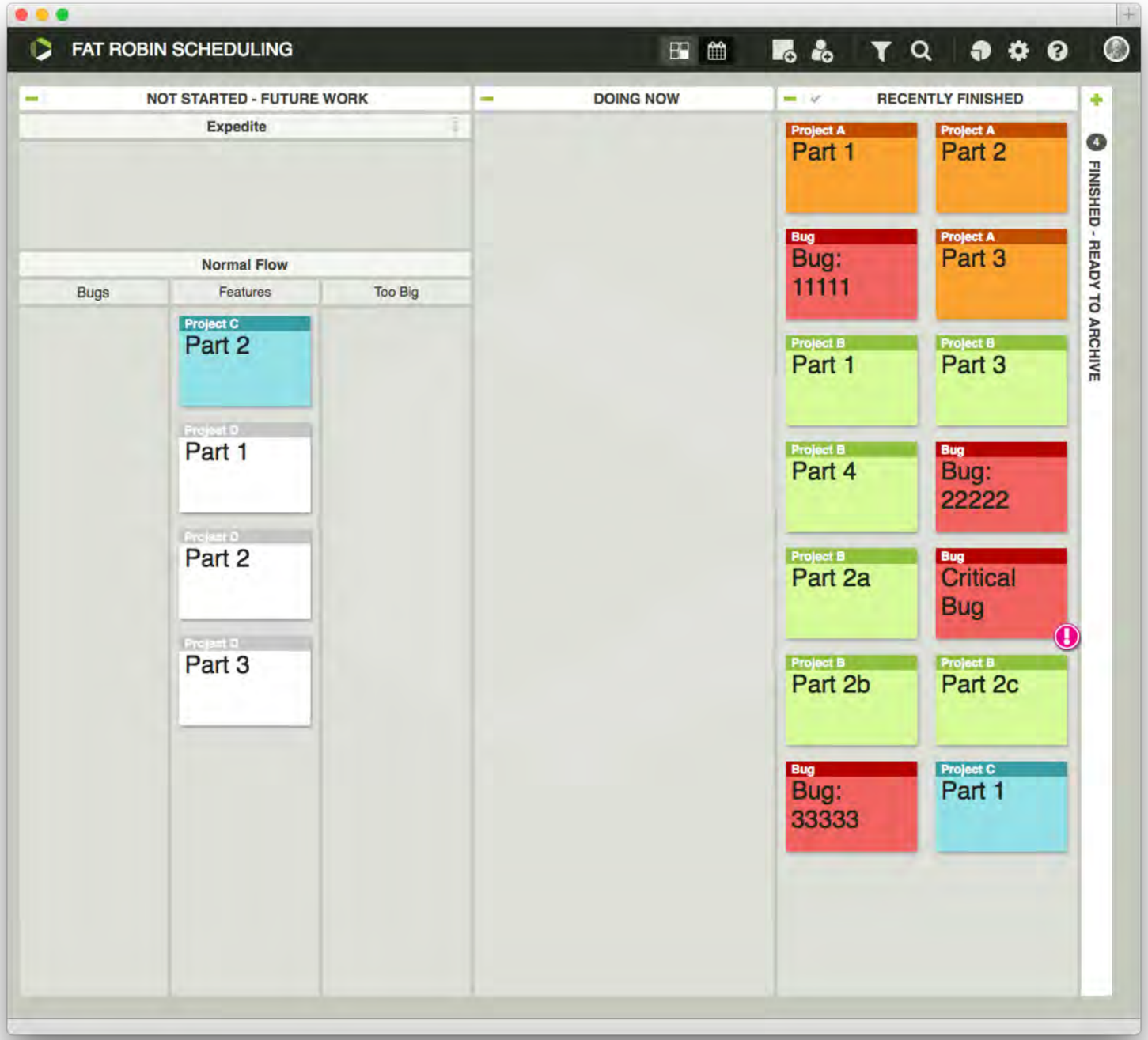
Fat Robin Scheduling

leankit



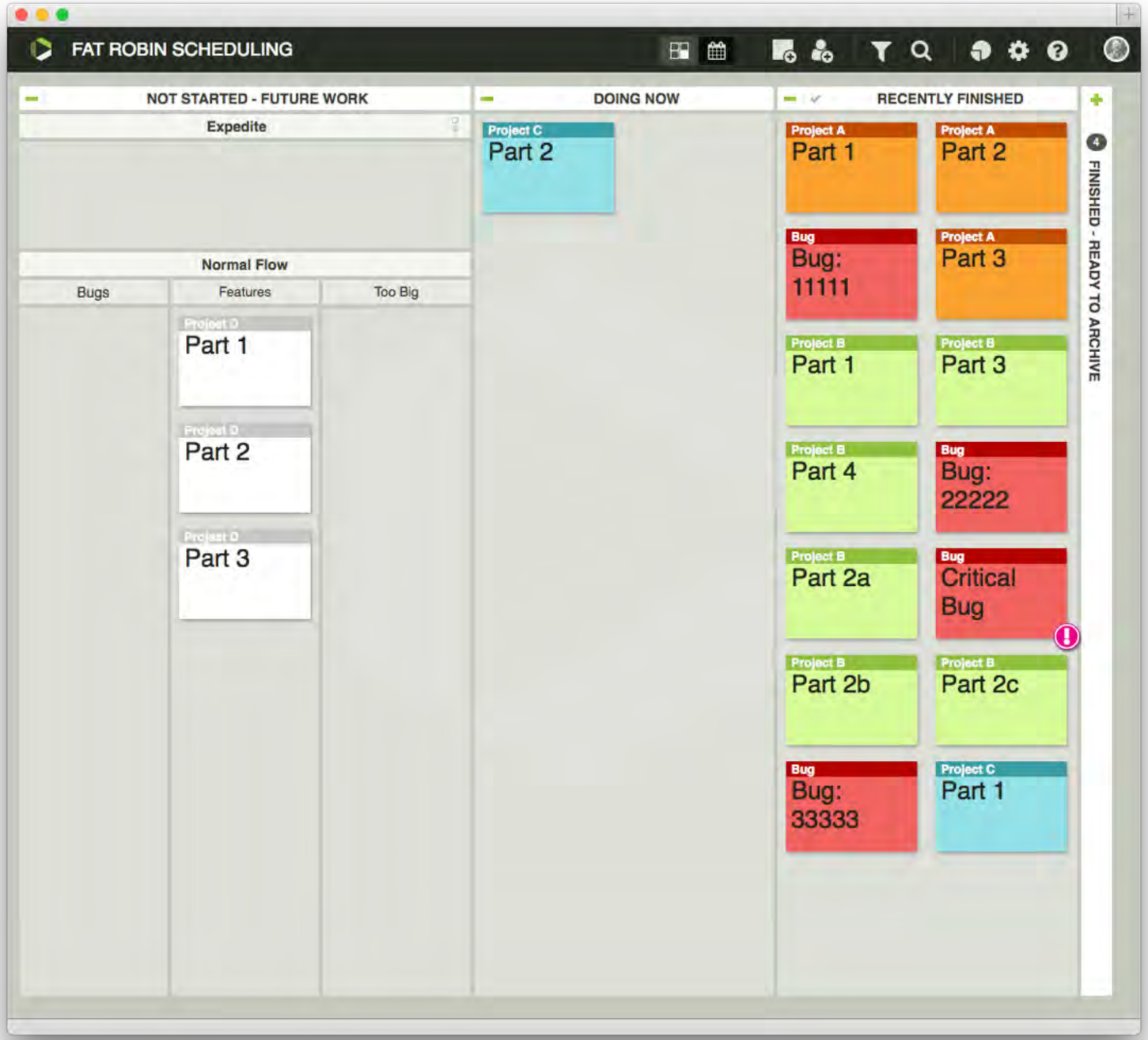
Fat Robin Scheduling

leankit



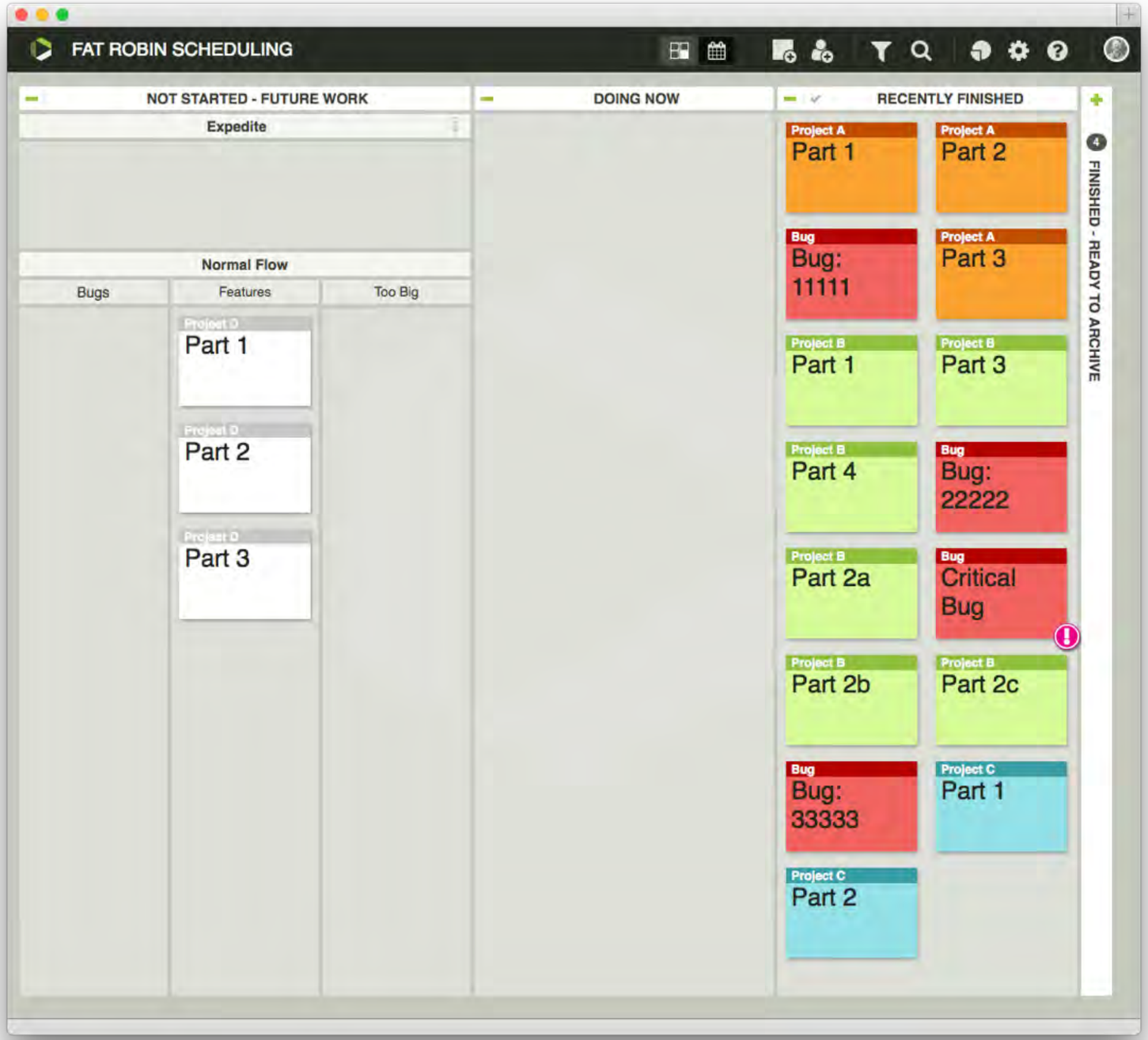
Fat Robin Scheduling

leankit



Fat Robin Scheduling

leankit



Fat Robin Scheduling

leankit

Frequent

Small

Good

Decoupled

Garbage

leankit

Tested
Logged
Documented
Reviewed

@ifandelse
github.com/ifandelse

leankit

Frequent

Small

Good

Decoupled

Garbage

Gold-plated

If you are not embarrassed by the first version of your product, you've launched too late.

Startup Quote!

REID HOFFMAN

FOUNDER, LINKEDIN

AT&T
9:42 AM
Camera
Photos
Tuesday
9
Calendar
73°
Weather
SMS
Text
Maps
Stocks
Settings
YouTube
Notes
Calculator
Clock
iPod
Safari
Mail
Phone

Cut, copy and paste
Works across apps
Undo support
Developer AP
Cocoa Touch
ort for text

Frequent

Small

Good

Decoupled Coordinated

Releasing a New Feature

Sales Team

Marketing Team

Browser Team

Mobile Team

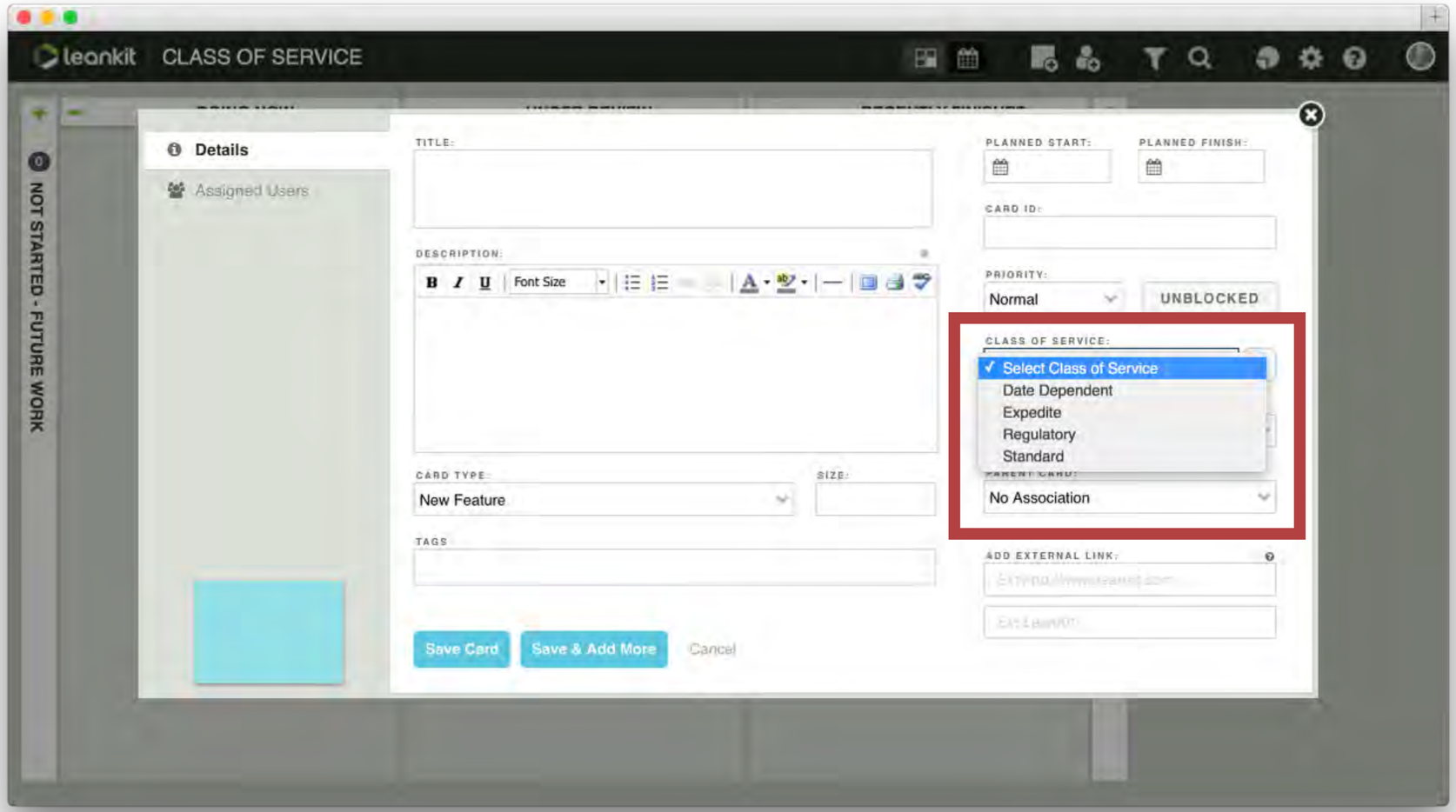
API Team

leankit

Decoupled Coordinated

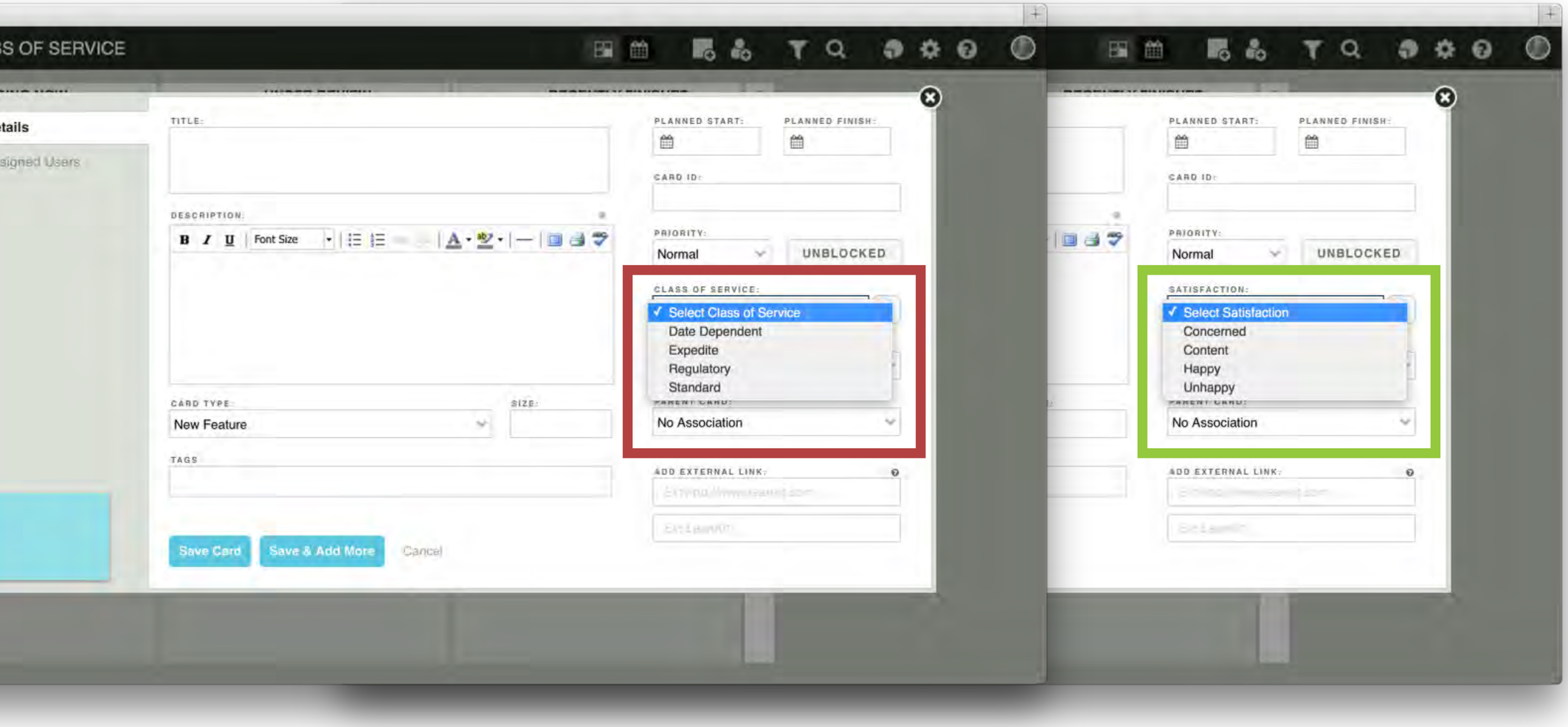
leankit

Example: Custom Icons



leankit

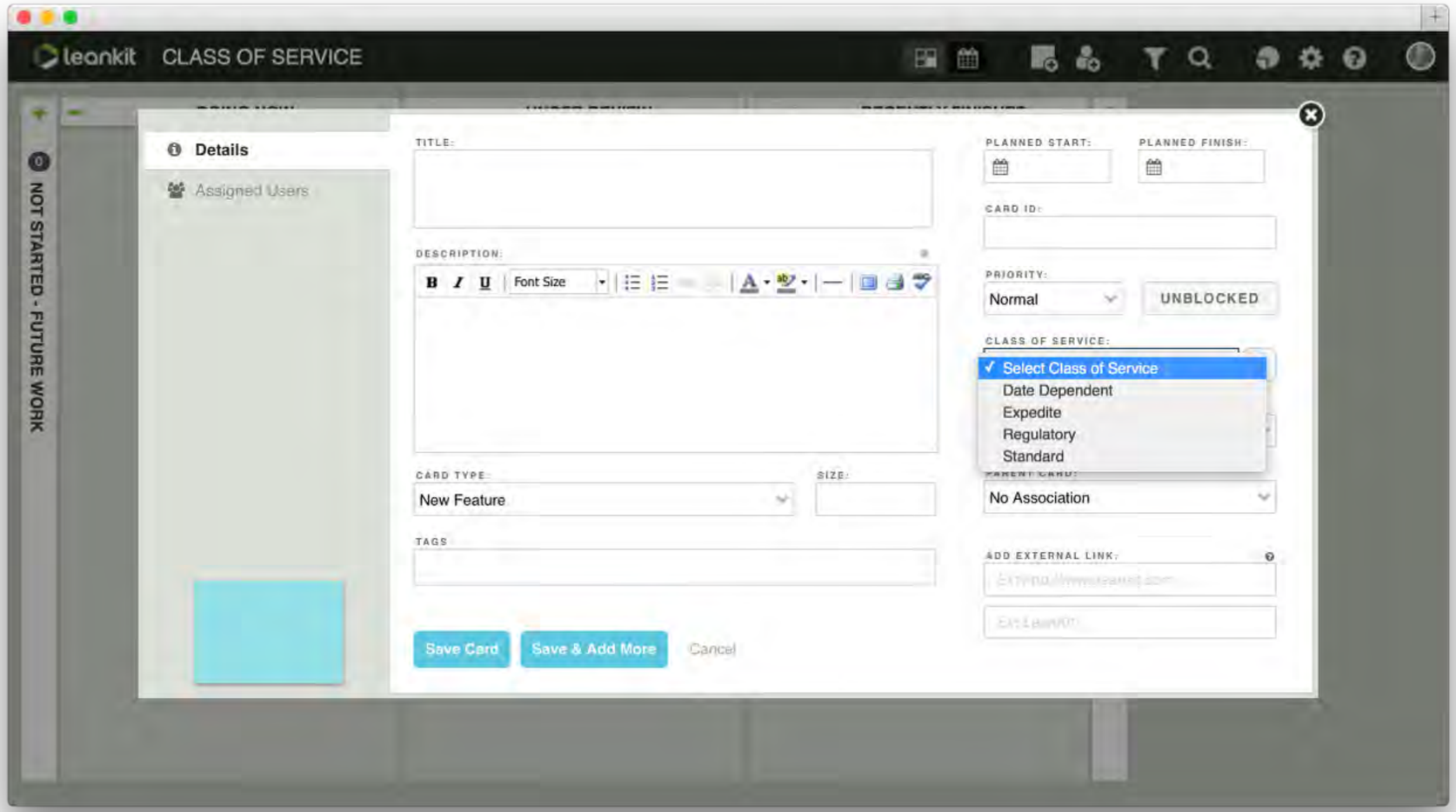
Example: Custom Icons



leankit

Example: Custom Icons

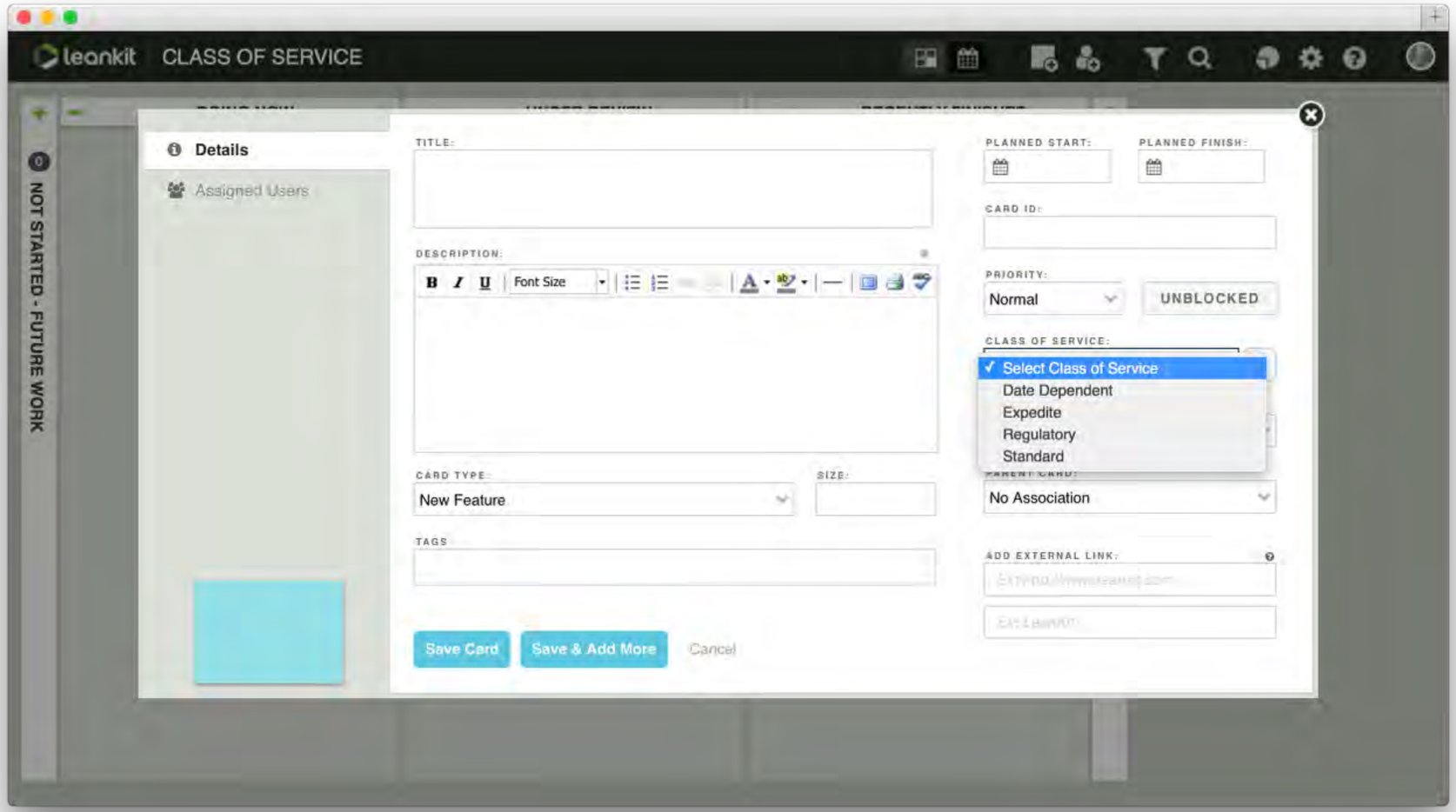
New field in the database, prepopulated … SHIP IT!



leankit

Example: Custom Icons

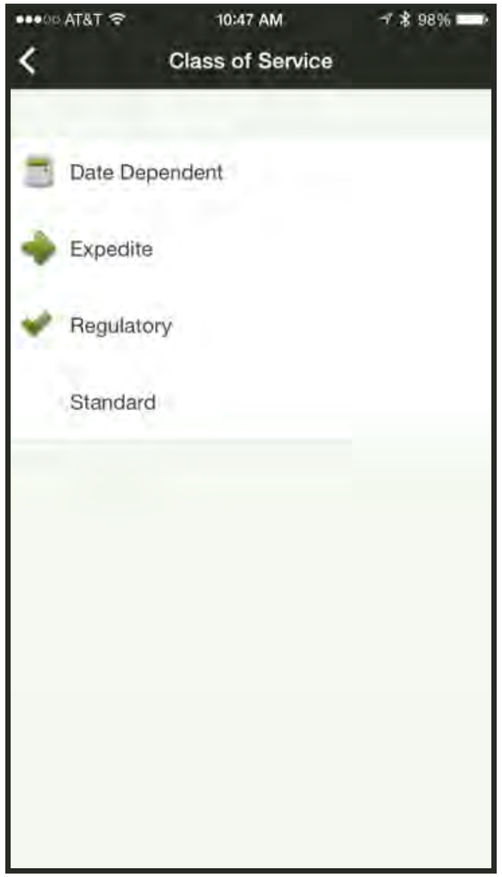
Use that to relabel UI in browser … SHIP IT!



leankit

Example: Custom Icons

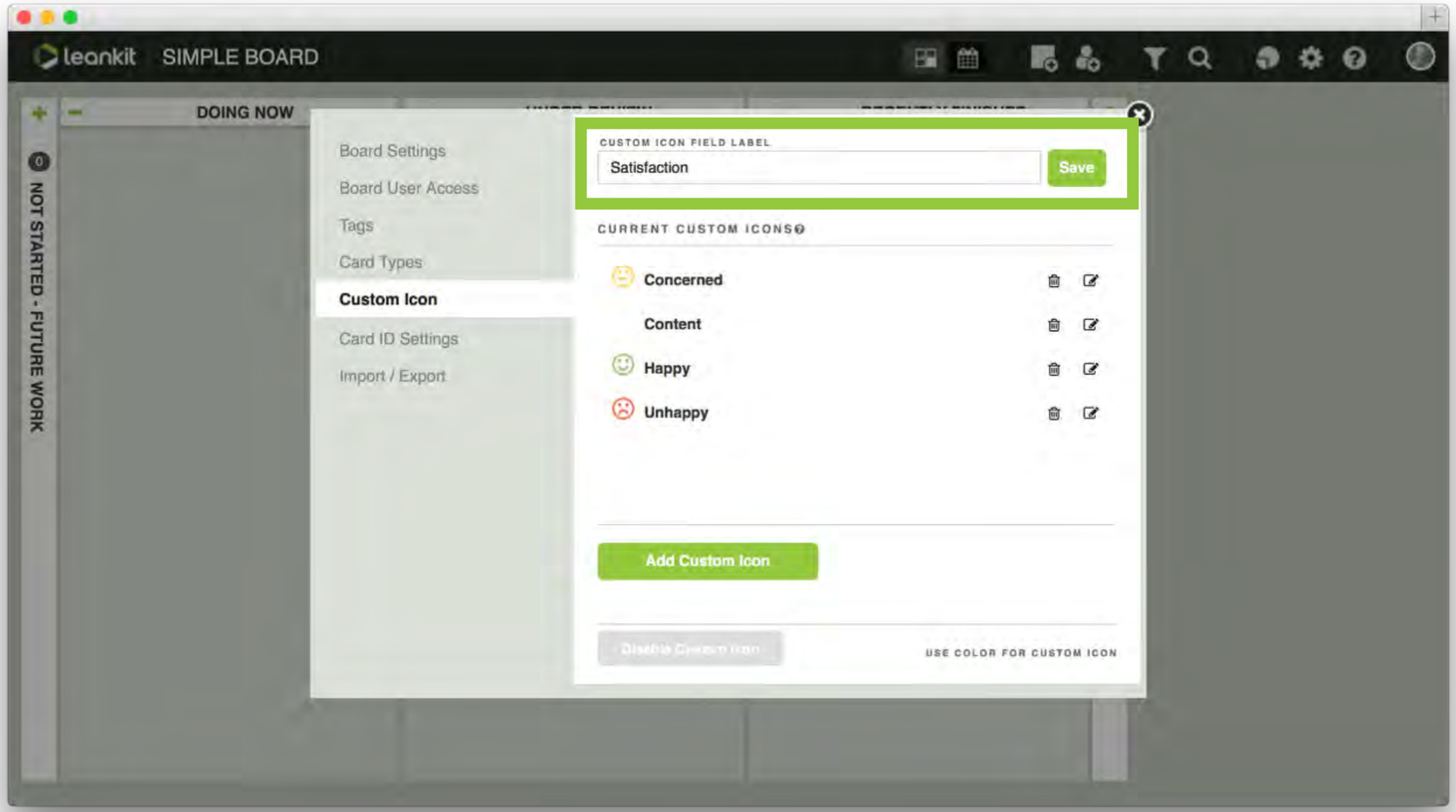
Use that to relabel UI in mobile … SHIP IT



leankit

Example: Custom Icons

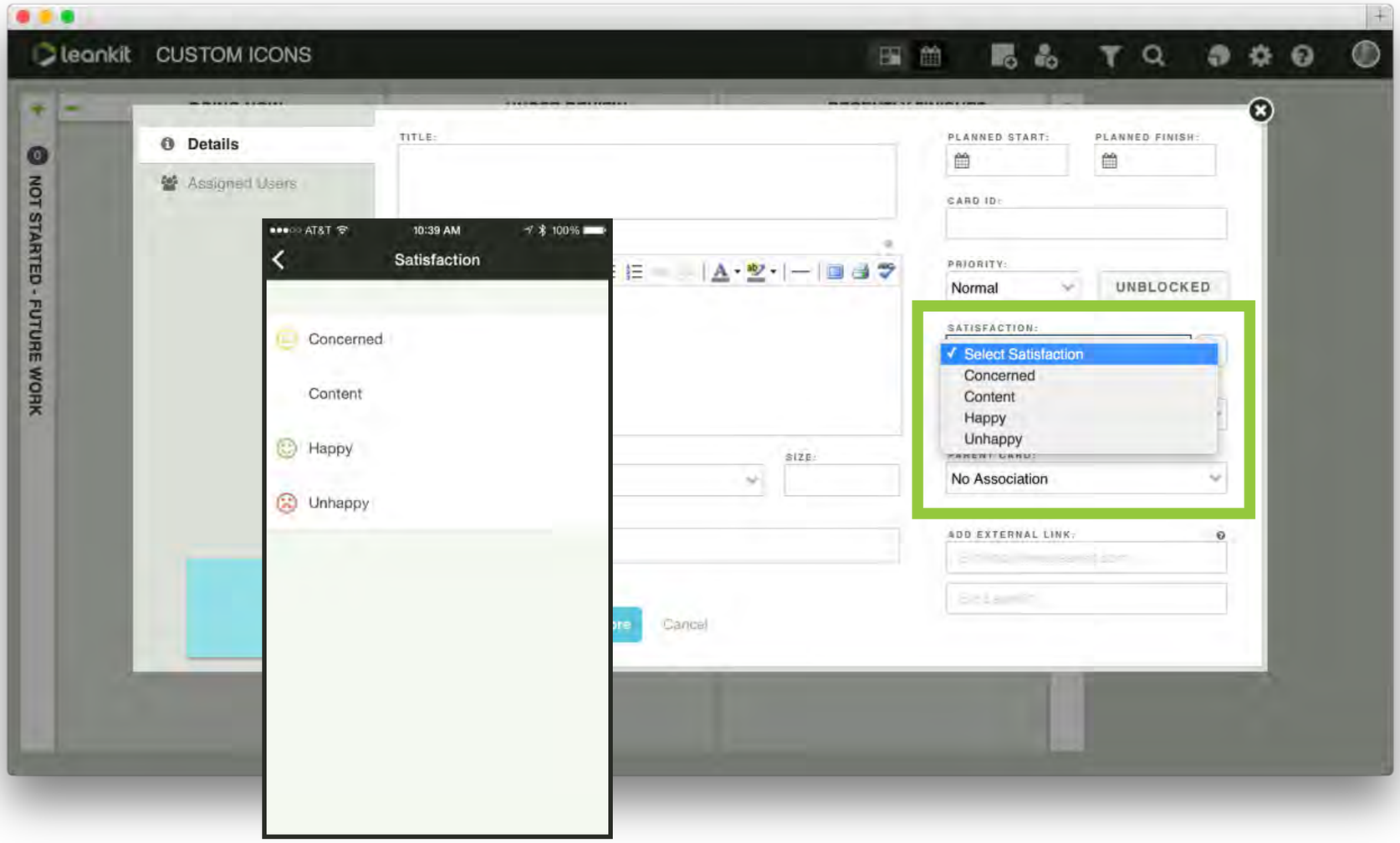
Add edit field to browser board edit UI … SHIP IT!



leankit

Example: Custom Icons

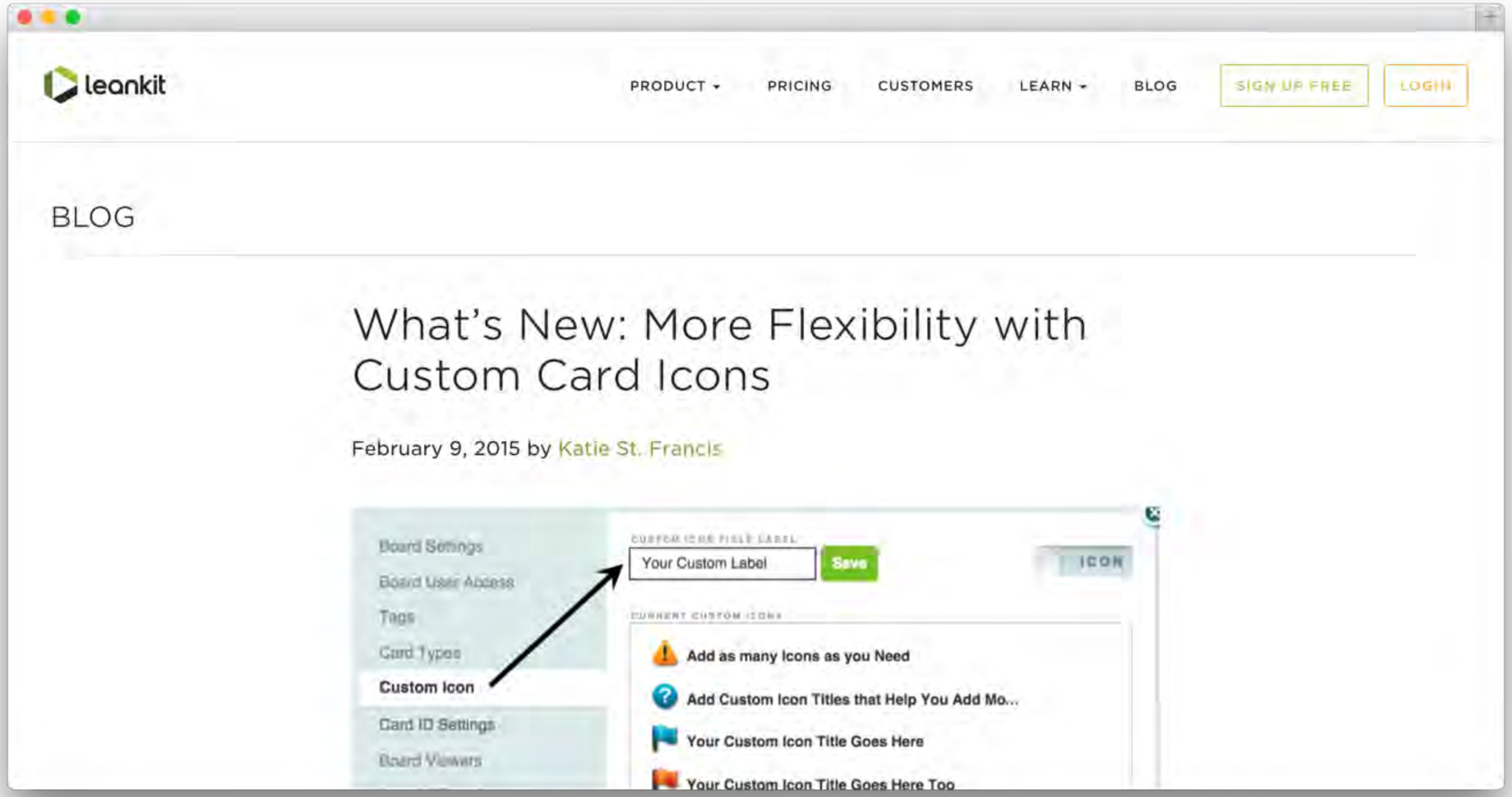
All the pieces are in place. It just works.



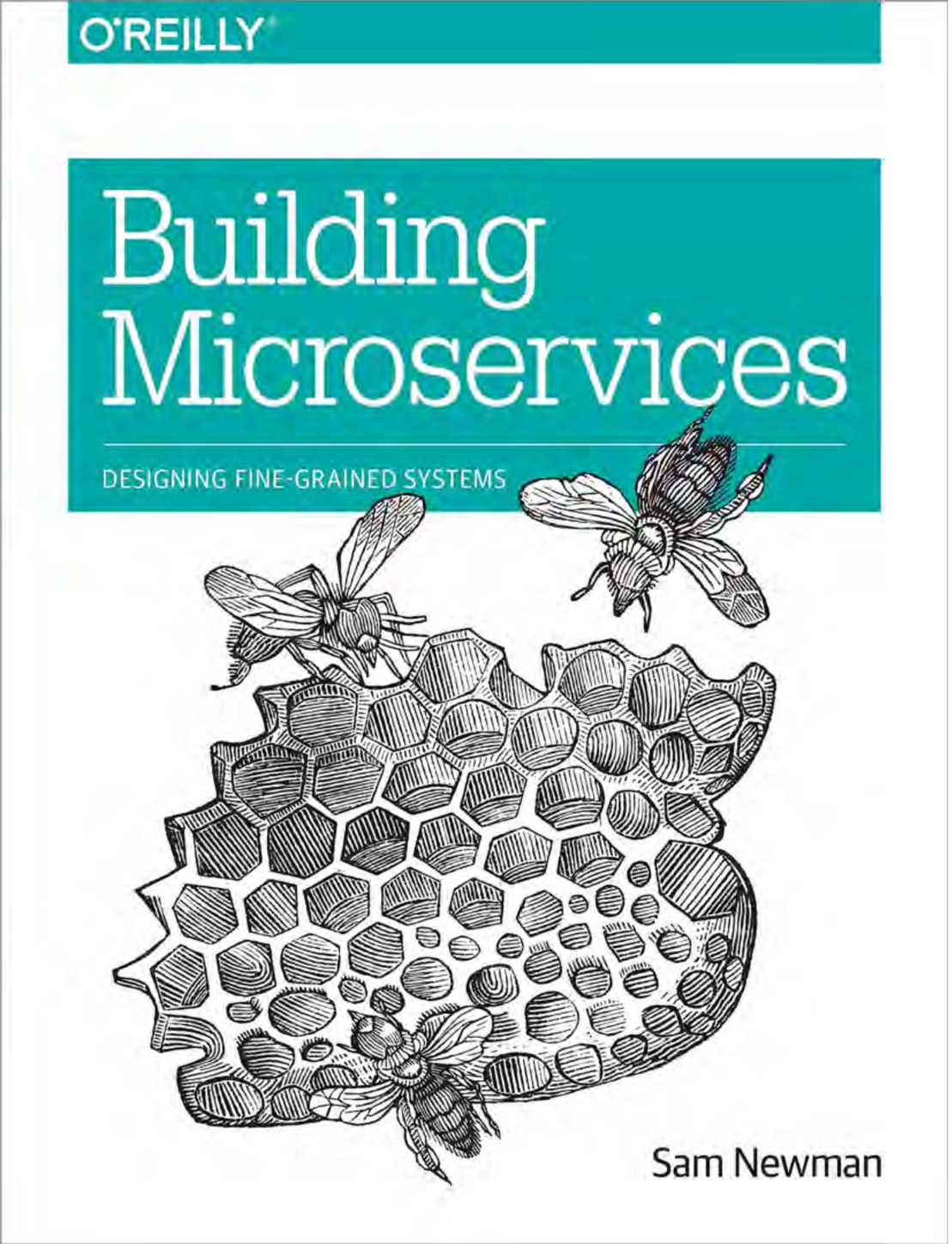
leankit

Example: Custom Icons

ANNOUNCE IT (prewritten … whenever)



leankit



"These services need to be able to change independently of each other, and be deployed by themselves without requiring consumers to change. ... Without decoupling, everything breaks down for us."

leankit

Frequent

Small

Good

Coordinated

Decoupled

Damaging

leankit

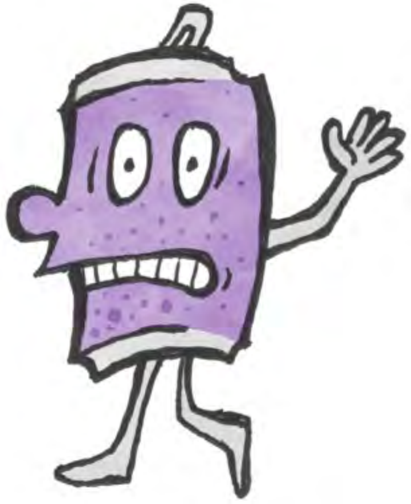
Frequent
Small
Garbage
Decoupled

leankit

Frequent
Small
Garbage
Decoupled

Frequent
Small
Good
Damaging

~~Frequent~~
~~Small~~
~~Good~~
~~Decoupled~~



leankit

PRIMUM NON NOCERE

leankit

PRIMUM NON NOCERE

First, do no harm

leankit

Frequent

Small

Good

Decoupled

(i.e. non damaging, do no harm)

bioethics

beneficence

non-maleficence

leankit

Frequent
Small
Good
Decoupled

Frequent
Small
Good
Decoupled

Frequent
Small
Good
Decoupled

Frequent
Small
Good
Decoupled

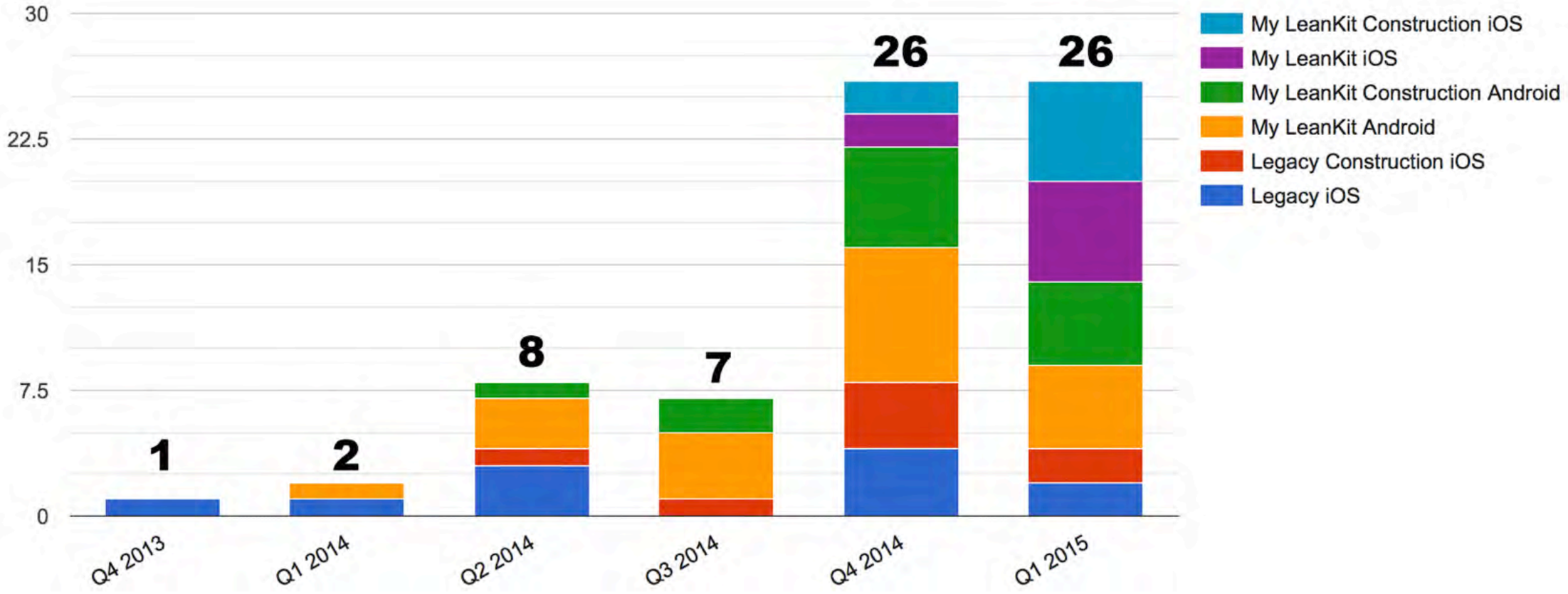
Frequent
Small
Good
Decoupled

leankit

Results

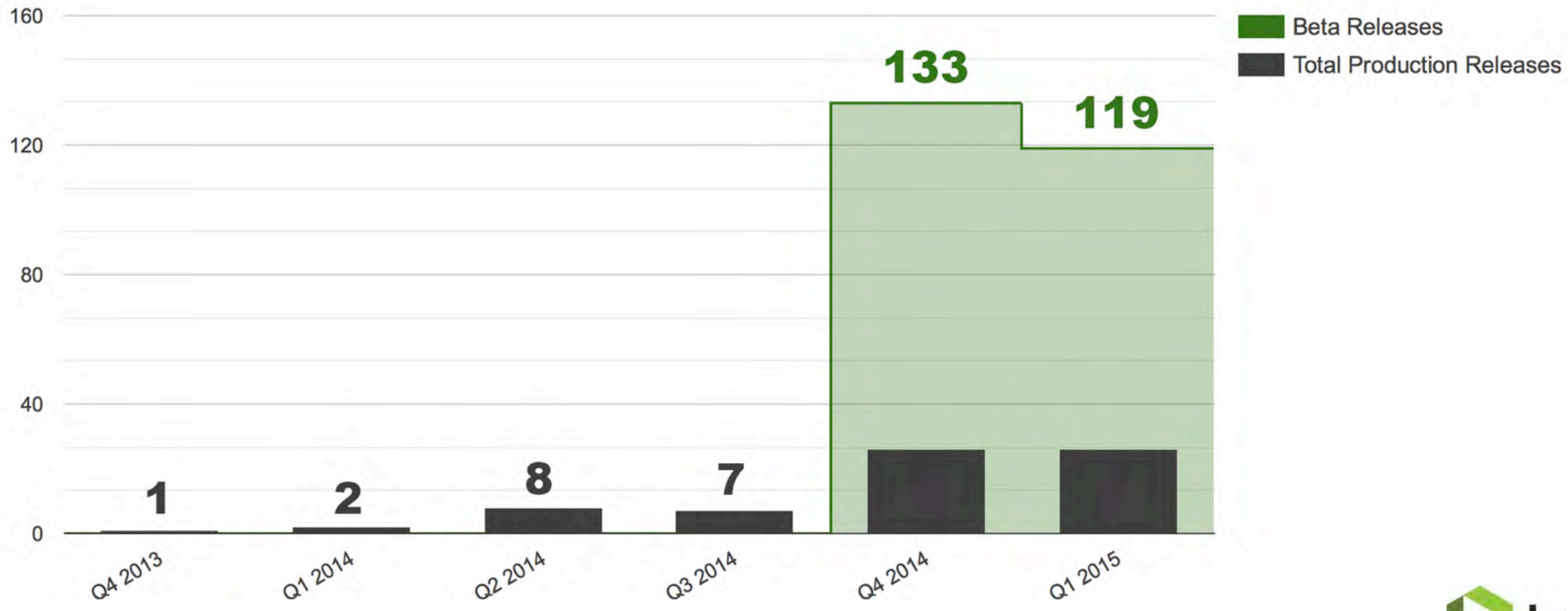
leankit

LeanKit Mobile Team Releases



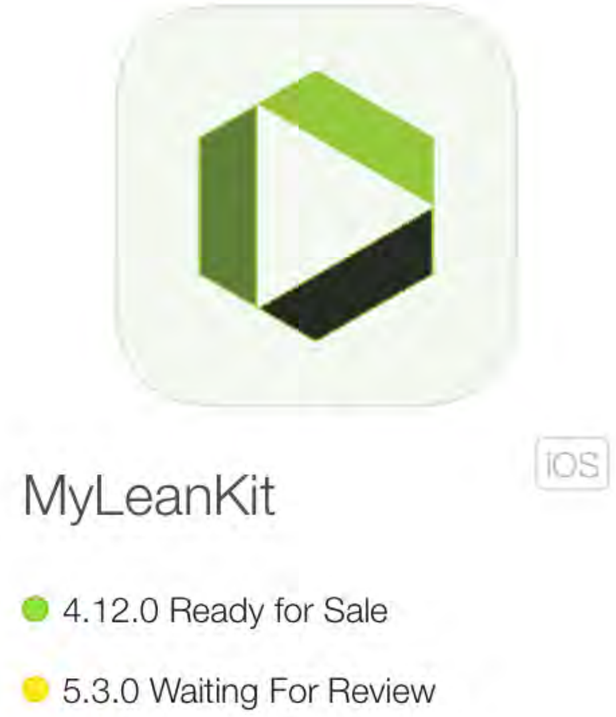
leankit

LeanKit Mobile Team Beta Releases



leankit

LeanKit Mobile Team Releases



leankit

Frequent
Small
Good
Decoupled

@daniel_norton
github.com/danielnorton
leankit.com/fsgd

leankit

2014 by LEANKIT – Daniel Norton, Jon Terry and Chris Hefley

FSGD (Fizz Good) – A LeanKit Way of Working

FSGD (Fizz Good) is made available under the Creative Commons Attribution-ShareAlike 4.0 International License:

http://creativecommons.org/licenses/by-sa/4.0/

View our FSGD (Fizz Good) content and download this presentation: leankit.com/FSGD