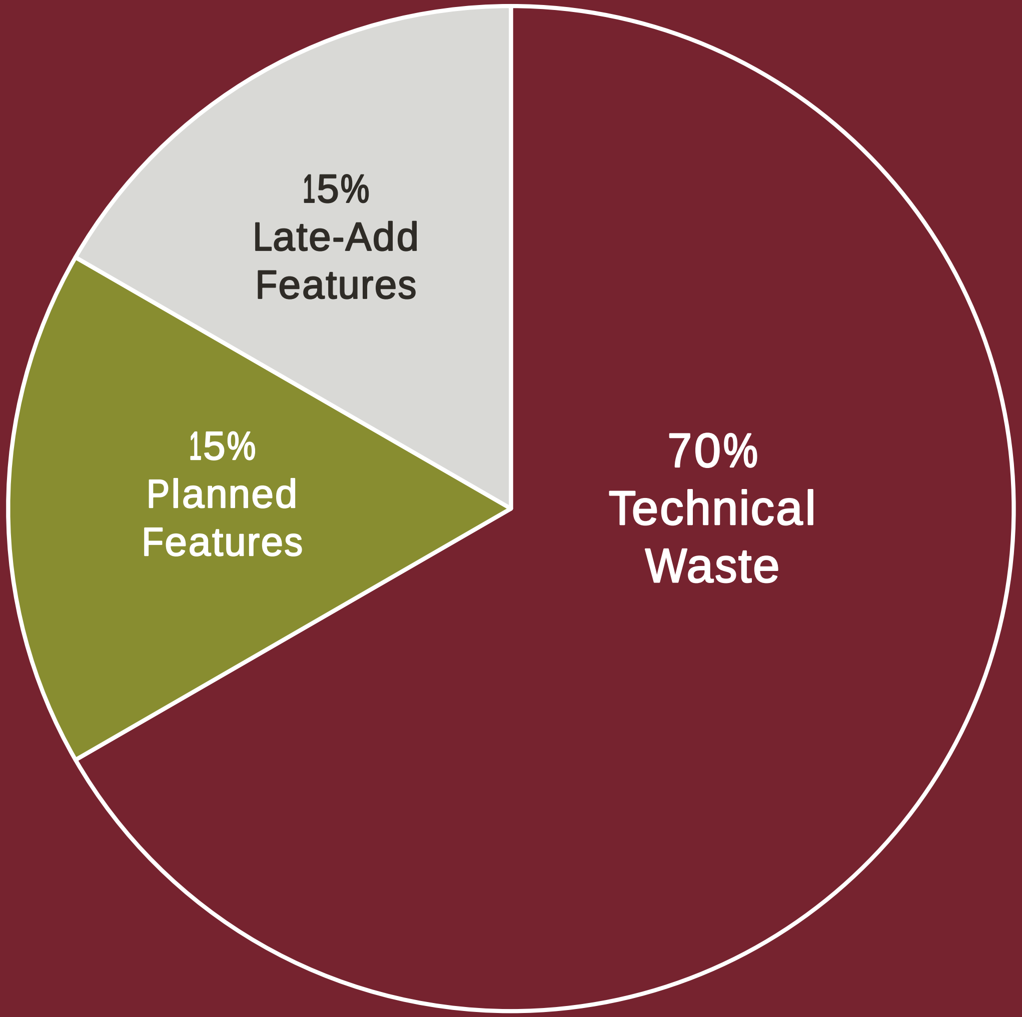
Your Path to More Features in Less Time

Creating a high productivity environment one day at a time.

sales@digdeeproots.com

DEEP ROOTS

Where time is spent in product development…



Does this seem too high?

Consider time spent on…

- Triage meetings
- Bugs (in all the ways)
- Stories that creep into next week
- Constraining features to cost
- Stabilization weeks
- Integrating legacy code

Also, did you think about downstream?

DEEP ROOTS

That’s What One Client Felt

Here’s their Zero Bug Story

DEEP ROOTS

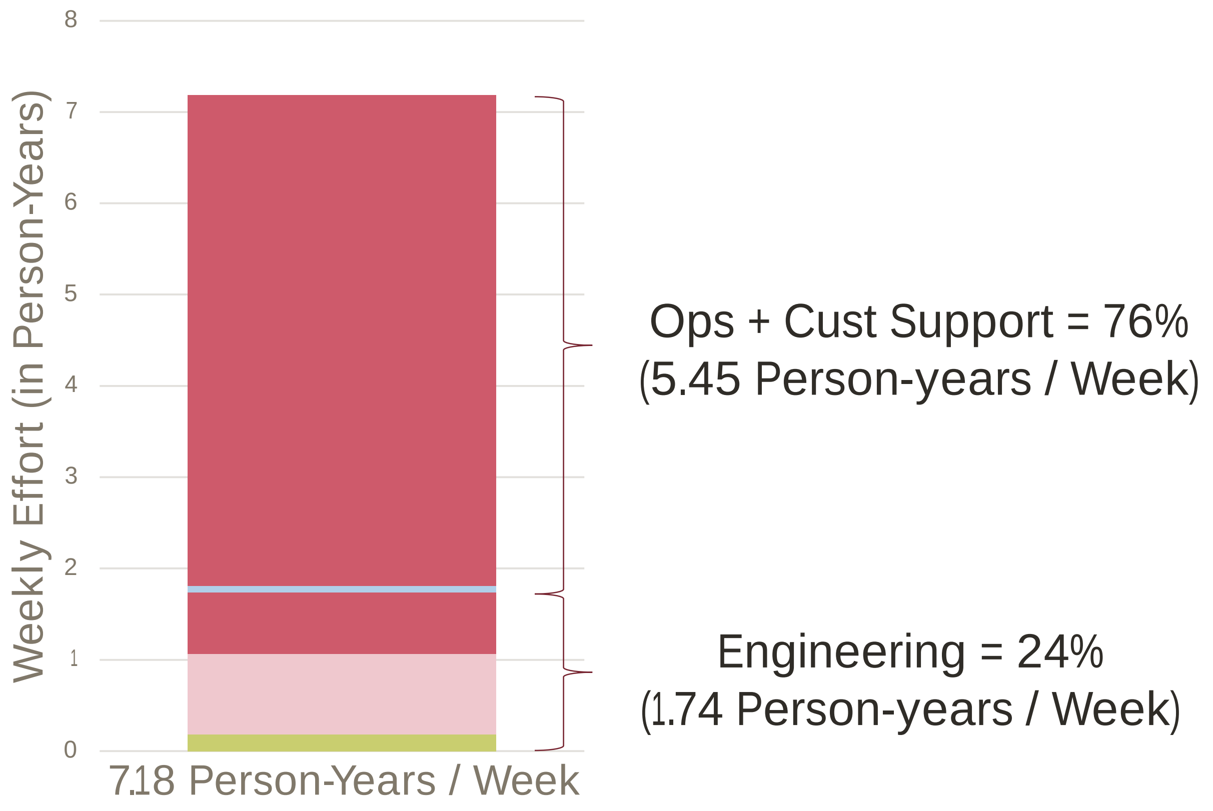
Chapter 1: Many Bugs

🟥 **Technical Waste**
- Tech Support (bugs)
- Failure Recovery
- Deployments
- Bug Fixing
- Defect Management

🟦 **Infrastructure**
- Infrastructure
- Non-Bug Support

▨ **Protection**
- Testing
- Emergency Features

🟩 **Value**
- Strategic Features



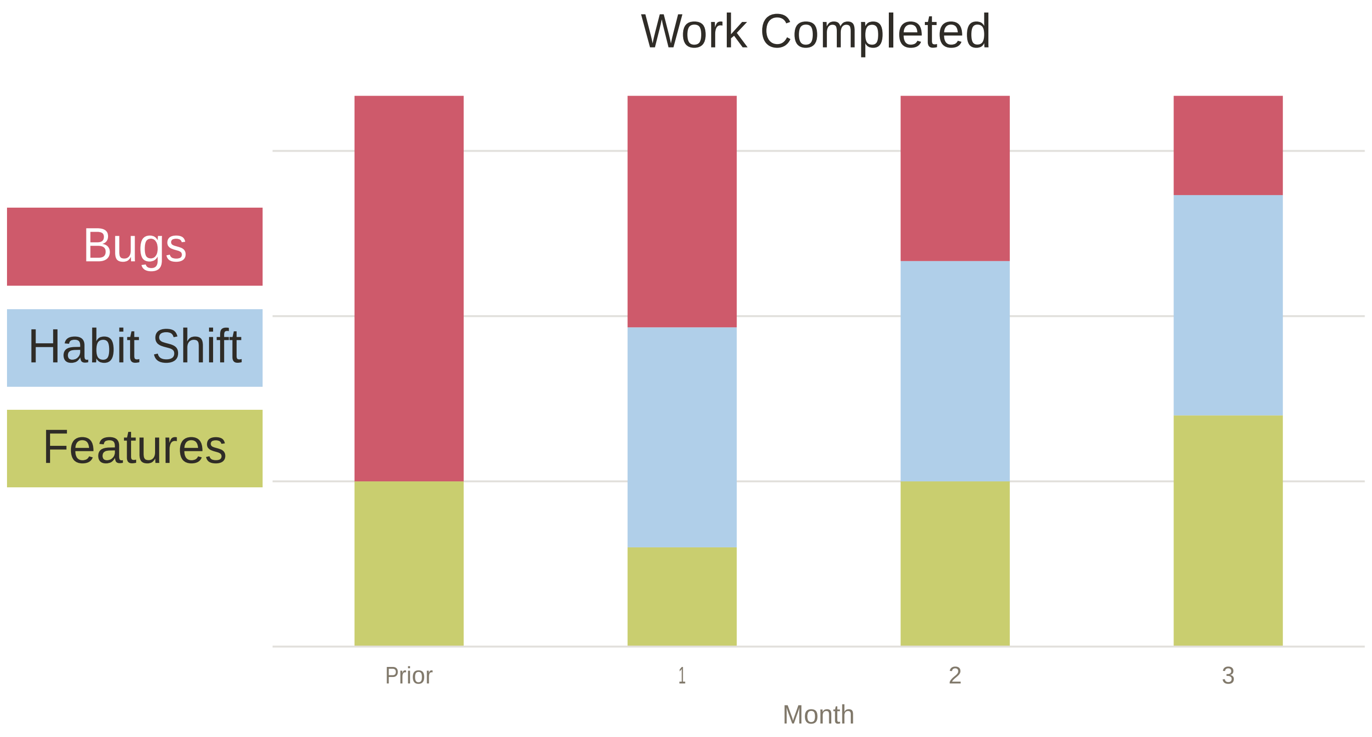
Total Effort:
- 93% Bugs
- 3% Planned Features
- 3% Emergencies
- 1% Overhead

Engineering Effort:
- 79% Bugs
- 11% Planned Features
- 11% Emergencies

DEEP ROOTS

Chapter 2: Our Solution

- 1 month to shift core behavior + 7 months to shift the active code.
- Change had universal support: 100% of managers and 100% of engineers.



Even the 1st Quarter was Breakeven!

DEEP ROOTS

Chapter 3: After Our Solution

Technical Waste

- Tech Support (bugs)
- Failure Recovery
- Deployments
- Bug Fixing
- Defect Management

Infrastructure

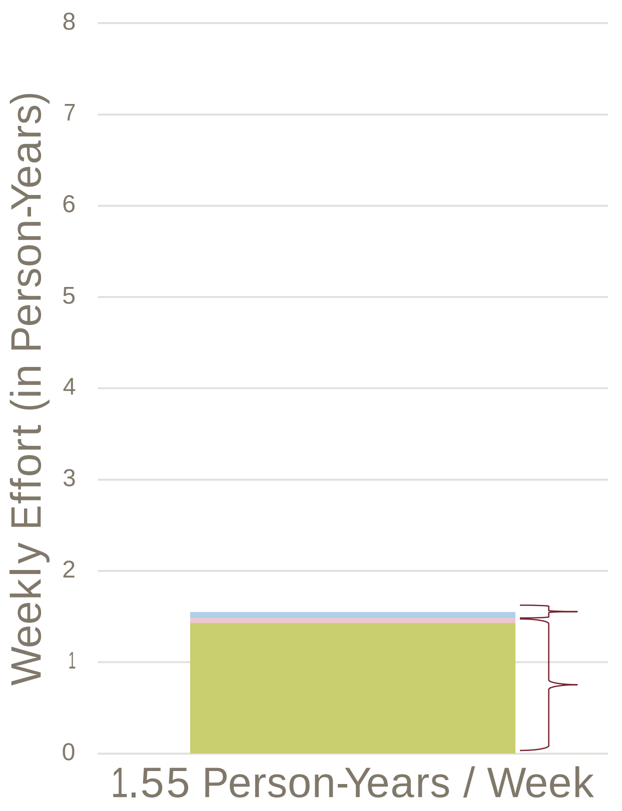
- Infrastructure
- Non-Bug Support

Protection

- Testing
- Emergency Features

Value

- Strategic Features



Ops + Cust Support = 4%
(0.07 Person-years / Week)
Engineering = 96%
(1.48 Person-years / Week)

Total Effort:

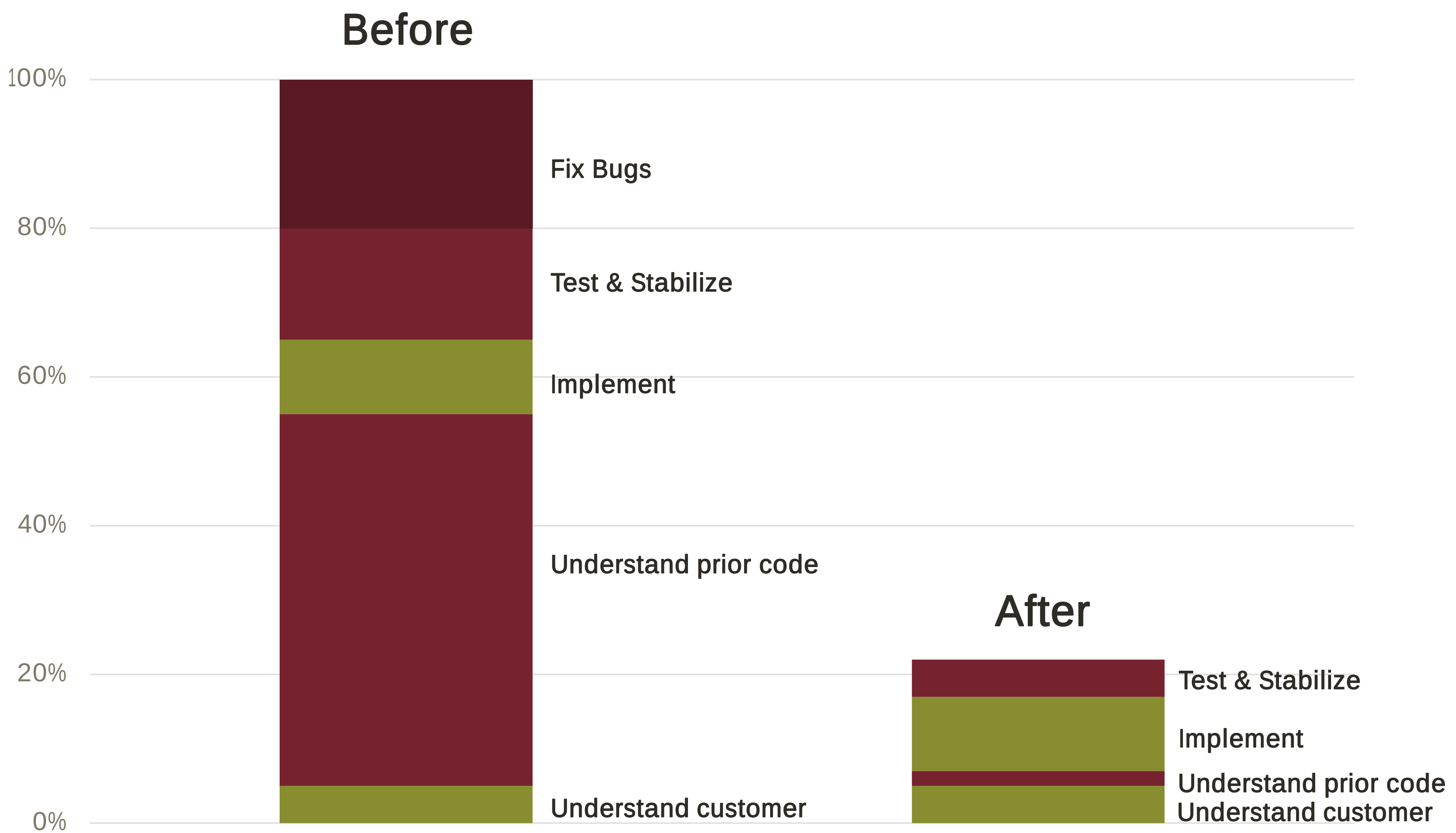
- 3% Bugs
- 93% Planned Features
- 0% Emergencies
- 4% Overhead

Engineering Effort:

- 3% Bugs
- 97% Planned Features
- 0% Emergencies

DEEP ROOTS

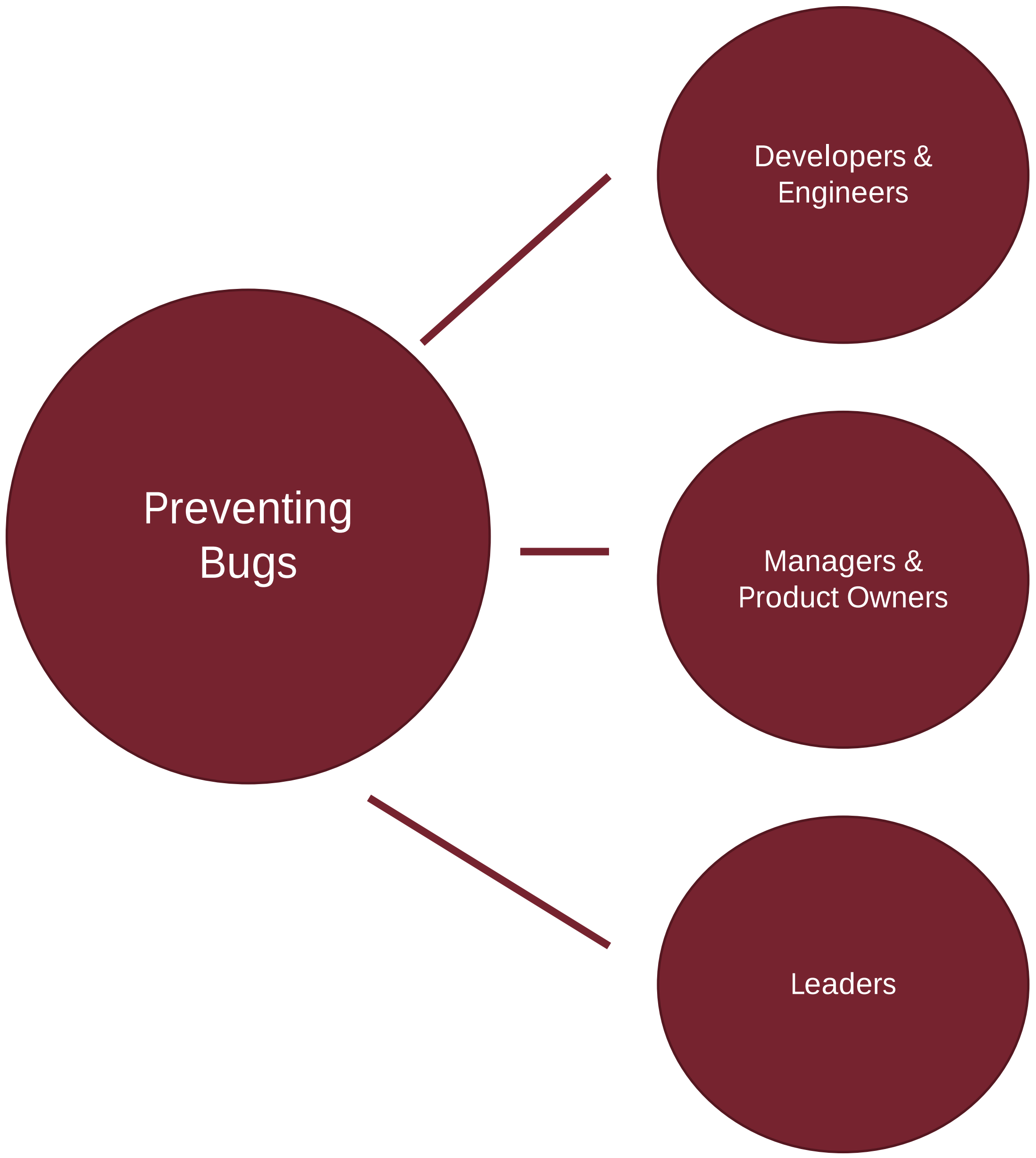
Also 4x stories done per Person-hour



DEEP ROOTS

Here’s What We Did

DEEP ROOTS



Code by Refactoring™

Progressively add practices that reduce the number of bugs written and allow preventing future bugs without paying more now.

Safeguarding™

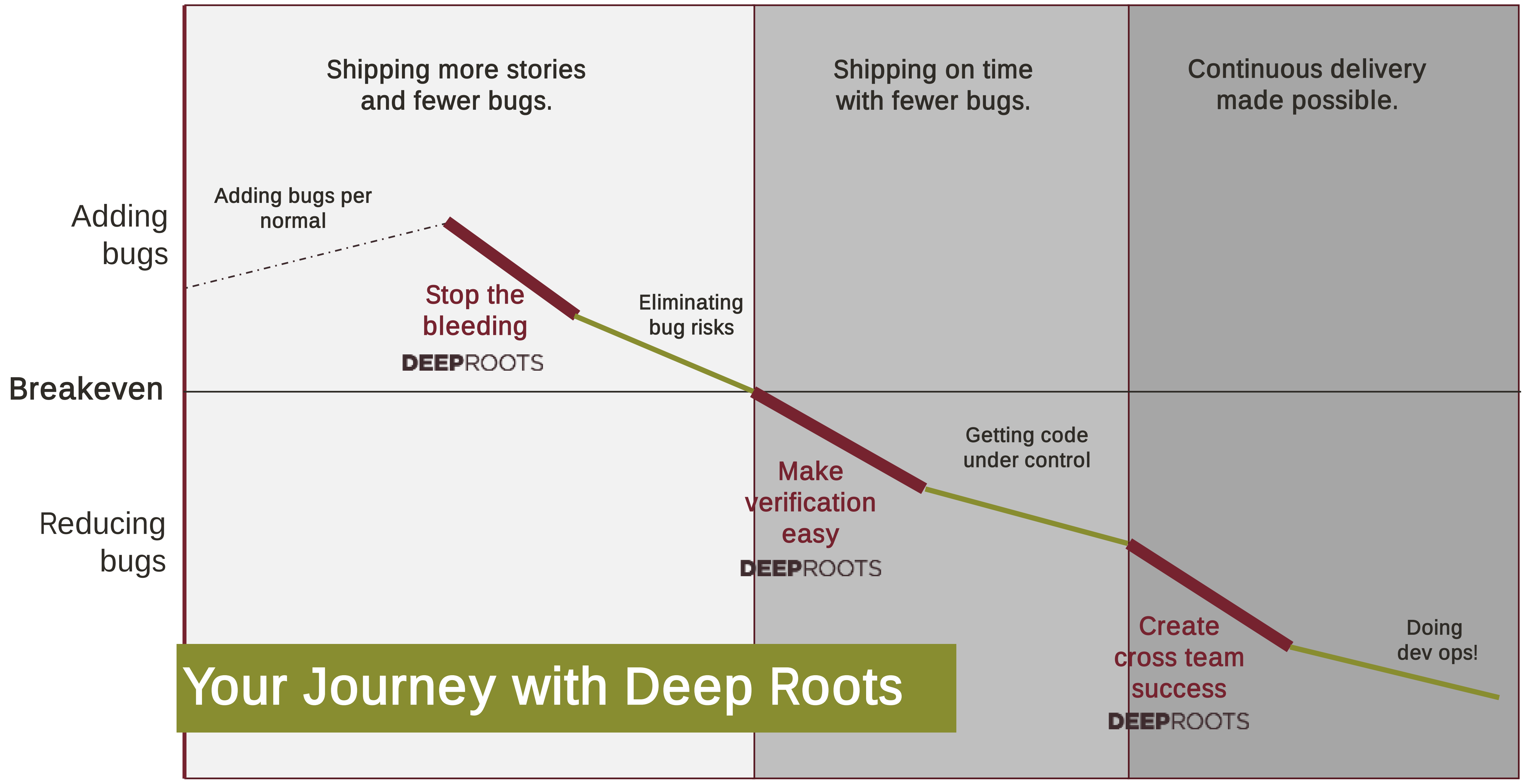
Amplify your existing bug-fixing habits with practices that identify and eliminate the hazards that led to those bugs in the first place.

Growing Responsible Ownership™

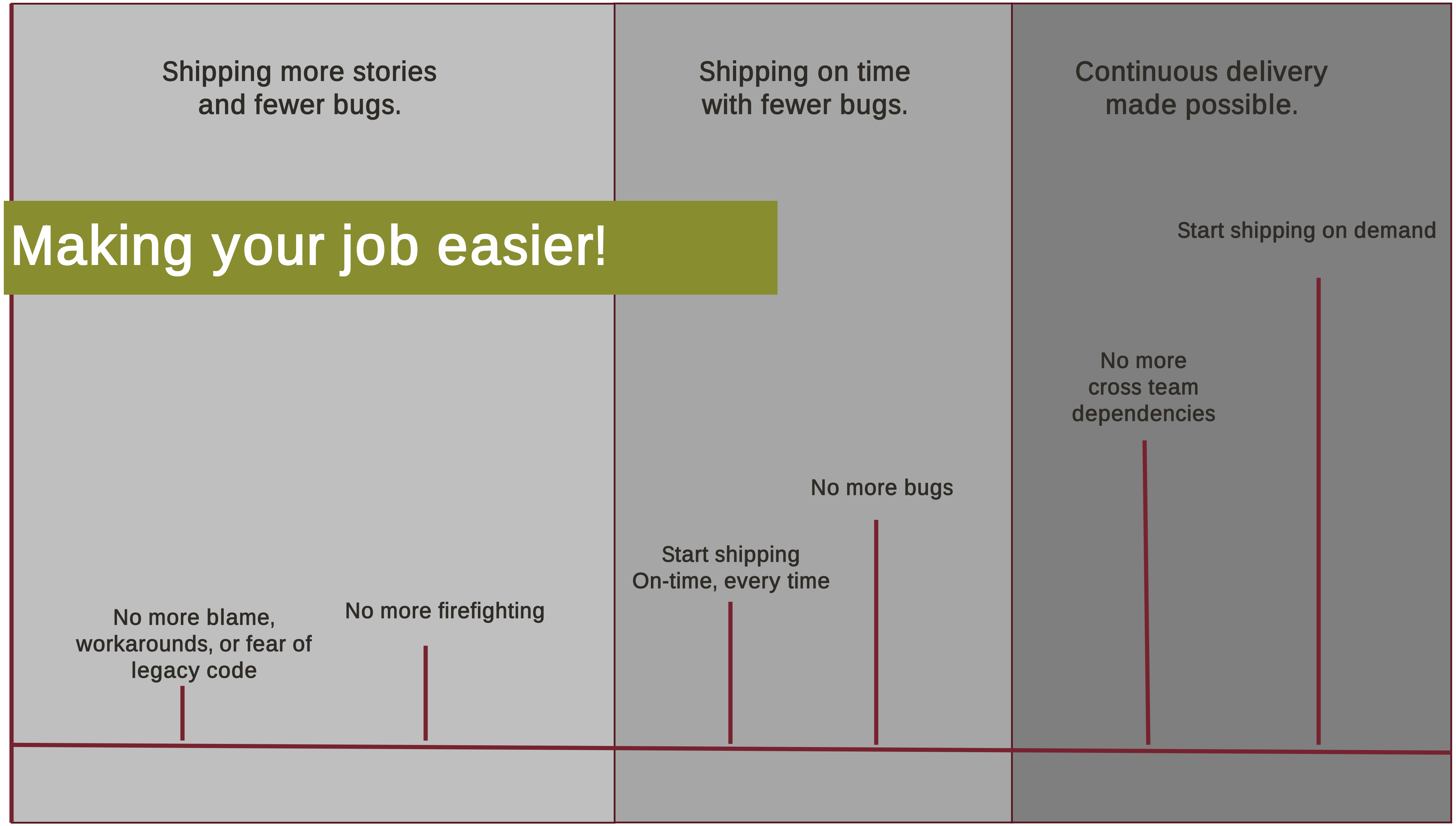
Create an organization that owns their technical debt and solves more problems than the leaders can simultaneously track.

DEEP ROOTS

Your Journey with Deep Roots



Making your job easier!


Shipping more stories and fewer bugs.
Shipping on time with fewer bugs.
Continuous delivery made possible.
No more blame, workarounds, or fear of legacy code
No more firefighting
Start shipping On-time, every time
No more bugs
No more cross team dependencies
Start shipping on demand

DEEP ROOTS

Phase One: Stop the Bleeding

Shipping more stories and fewer bugs.

- Less firefighting
- Less time on defects
- Aligned prioritization between features and tech debt

Phase Two: Make Shipping Easy

Shipping on time with fewer bugs.

- Automatically verify whole product at every commit
- Less in-team stress
- Ship on time

Phase Three: Create Cross Team Success

Continuous delivery made possible

- Not blocked by other teams
- Ignore technical constraints when prioritizing work
- Sustained technical excellence

Phase One		Phase Two		Phase Three	
4 Months		9 Months		6 Months	
1 Month Together	3 Months on your own with check-ins	3 Months Together	6 Months on your own with check-ins	3 Months Together	3 Months on your own with check-ins

DEEP ROOTS

Describing the Path

Financial Debt			Technical Debt
Control the spending.	Resolve >50% of all bugs and >50% of story development time **by** refactoring hard-to-read code.	Teams begin addressing the team-specific top risks.	Stop the bleeding.
Pay off expensive credit card debts.	Resolve 25% of all bugs and enable automated verification **by** refactoring duplicate code and non-local interactions.	Teams own tech debt funding decisions and ROI.	Make shipping easy.
Pay off less expensive personal loan debts.	Eliminate schedule dependencies, the need for release-level stabilization, and 10% of all bugs **by** refactoring cross team code dependencies.	Teams expands its quality ownership to include sustainability, secession, and other teams.	Create cross team success.

DEEP ROOTS

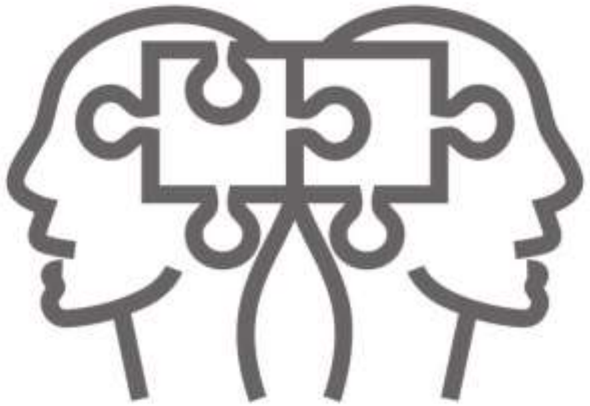
What Makes Us Different?

DEEP ROOTS

We don’t train. We change behaviors.

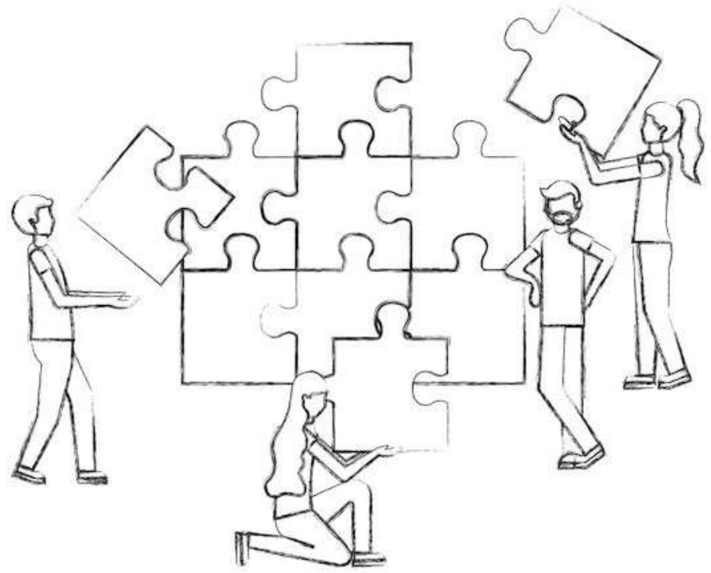
On the Job

We provide habit shifts integrated in the work every day.



Scale Across Teams

We provide structure that unlocks your in-house expertise



DEEP ROOTS

Exposing

Building

Training exposes people to new ideas.

A learning culture helps people change behaviors.

Exposing	Building
Trying new concept on simple problem out of context.	Learning on the job in your real context in small batches.
Learning a pre-defined set of slides and topics.	Experimenting with targeted feedback.
Engaging in a fun workshop for the day.	Engaging behavioral change that lasts.

DEEP ROOTS

How we work.

One day workshop

One Month Application

WK1	Shift Habit			Shift Habit		
	Demo shift	Practice shift on the job	Coach retro	Demo shift	Practice shift on the job	Coach retro
WK2	Shift Habit			Shift Habit		
	Demo shift	Practice shift on the job	Coach retro	Demo shift	Practice shift on the job	Coach retro
WK3	Shift Habit			Shift Habit		
	Demo shift	Practice shift on the job	Coach retro	Demo shift	Practice shift on the job	Coach retro
WK4	Shift Habit			Shift Habit		
	Demo shift	Practice shift on the job	Coach retro	Demo shift	Practice shift on the job	Coach retro

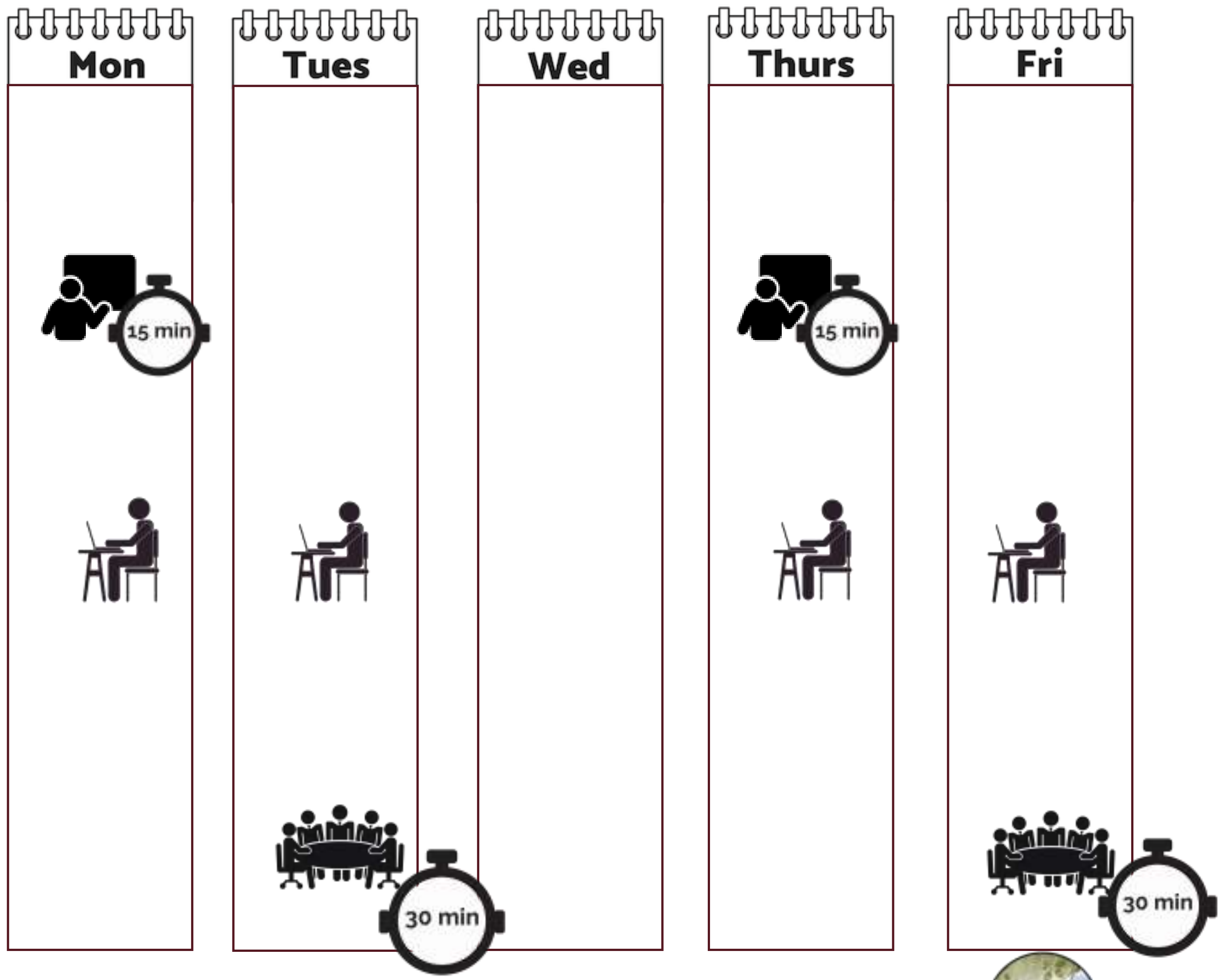
DEEP ROOTS

A Week in the Team's Life

Experience demo with Deep Roots
to identify what to shift

Integrate daily practice on the job
to apply and make change real

Share stories with Deep Roots
to make it sustainable



DEEP ROOTS

Ready for Zero Bugs?

Contact us for help!

sales@digdeeproots.com

DEEP ROOTS