

Single developer

- ▶ The project started out with a single senior developer
- ▶ 2017-2020 working on the code part time
- ▶ He created useful software that schools used to make decisions
- ▶ Interesting experiments were done: twilio
- ▶ Field testing the school quality framework
- ▶ All happy families are alike; each unhappy family is unhappy in its own way.

- ▶ Consortium of districts in the state of Massachusetts

A company was brought in to modernize the application

- ▶ 50 person company
- ▶ 10 years of experience
- ▶ Used by 1 in 5 schools
- ▶ \$100,000 budget, no working software was delivered

VMWare

- ▶ CVS, Albertons, Garmin
- ▶ 3 Developers, 1 product manager, 1 designer
- ▶ Dream team
- ▶ Sept 2021 - December 2021
- ▶ React devs
- ▶ Last time they used rails was rails 3
- ▶ Pair programming with experienced developers accelerated my learning.

The app

- ▶ Delivered the welcome, overview, and browse
- ▶ Upgraded the application from rails 5.0.1 now the latest version of 7
- ▶ Two apps live in the same codebase

RPP project

- ▶ The success of the MCIEA project led to a new project with the district of Lowell
- ▶ Multiple surveys per year
- ▶ Early education surveys
- ▶ Modifying language used throughout the framework
- ▶ Additional ways to disaggregate data

ECP

- ▶ The ecp continues the work

Team

- ▶ One Developer
- ▶ One Product Manager

Rails allows fast prototyping

- ▶ Rails allows us to get a prototype up and running quickly
- ▶ The prototype can be used to get feedback from the client

All the work we didn't do

- ▶ Rails is still slower than a designer
- ▶ Figma
- ▶ Designers are still slower than gathering requirements
- ▶ Miro

Work in small batches

- ▶ 1 point = 1 day
- ▶ 2 points = a few days
- ▶ 3 points = Stories I say no to

Code quality

- ▶ Write only enough code to complete the story
- ▶ This includes tests
- ▶ Each time I return to a file to modify or add features:
- ▶ If it's hard to reason about, refactor it
- ▶ This is only possible because tests are already written
- ▶ I leave it better than I found it
- ▶ High churn files eventually become better written software
- ▶ Ugly code that never needs to be touched isn't a problem
- ▶ The ultimate goal is working software

Automate as much as possible

- ▶ Enrollment numbers
- ▶ Staffing data
- ▶ Admin data

Github copilot

- ▶ Is there a way to code faster?
- ▶ It makes me faster writing boilerplate code

Continuous Improvement

- ▶ Retros
- ▶ Speedback

Make space for learning

- ▶ Make time to learn
- ▶ On a small team you have to know a little of everything

Open source