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

MCIEA

- Consortium of districts in the state of Massachusetts

Another company 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 accellerated 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 dissaggregate 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
- Rails is still slower than a designer
- There is no faster code than code you don't have to write
- Working out the designs allows you skip implementations that were doomed to fail from the start
- Figma: All the work we didn't do

Points

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

Test everything

- I hate testing but I write them anyway
- On a team of one the only way to know you've introduced a regression is to write tests
- It's a million times harder to debug production so be proactive

Code quality

- Do as little as possible to get the story done
- If it's hard to reason about, refactor it
- Leave the code better than you found it
- The ultimate goal is working software

Automate as much as possible

- Enrollment numbers
- Staffing data
- Admin data

Continuous Improvement

- We don't have time or manpower to do things inefficiently

Make space for learning

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

The meetings that work for us

- stand up
- weekly goals
- retros
- ipm
- speedback

Four key metrics

- Change lead time
    - Deployment frequency
    - Change failure rate
    -