Software Engineering

For Fun and Profit

Introduction

Hi, I'm Casey Stella!

Introduction

A little bit about the now...

- I am a staff engineer at Stripe

Introduction

A little bit about the now...

- I am a staff engineer at Stripe
- Stripe is a fintech company which processes a good chunk of the internet's payments

Introduction

A little bit about the now...

- I am a staff engineer at Stripe
- Stripe is a fintech company which processes a good chunk of the internet's payments
- Specifically, I'm the lead of the Machine Learning Infrastructure

Introduction

A little bit about the now...

- I am a staff engineer at Stripe
- Stripe is a fintech company which processes a good chunk of the internet's payments
- Specifically, I'm the lead of the Machine Learning Infrastructure
- Stripe uses a lot of machine learning: fraud detection, image recognition, etc.

Introduction

A little bit about the now...

- I am a staff engineer at Stripe
- Stripe is a fintech company which processes a good chunk of the internet's payments
- Specifically, I'm the lead of the Machine Learning Infrastructure
- Stripe uses a lot of machine learning: fraud detection, image recognition, etc.
- We ensure ML models can run at the speed and scale of internet commerce.

Introduction

A little bit about the now...

- I am a staff engineer at Stripe
- Stripe is a fintech company which processes a good chunk of the internet's payments
- Specifically, I'm the lead of the Machine Learning Infrastructure
- Stripe uses a lot of machine learning: fraud detection, image recognition, etc.
- We ensure ML models can run at the speed and scale of internet commerce.

Open Questions:
What even is software engineering?
How in the heck did I end up doing this?

Software Engineering

Programming is the art of turning fuzzy ideas into working code.

Software Engineering is the discipline of programming in such a way that your coworkers don't want to kill you in your sleep

Software Engineering is the discipline of programming in such a way that your coworkers don't want to kill you in your sleep...often.

Software Engineering: Unexpected Skills

There are a few unexpected skills that you might not have thought to cultivate:

Software Engineering: Unexpected Skills

There are a few unexpected skills that you might not have thought to cultivate:

- The Scientific Method: Software Engineering is an experimental science/engineering discipline.

Software Engineering: Unexpected Skills

There are a few unexpected skills that you might not have thought to cultivate:

- The Scientific Method: Software Engineering is an experimental science/engineering discipline.
- Logic and Reasoning: Most practical software is a complex set of systems with complex dependencies and expectations. Reasoning about structure is important.

Software Engineering: Unexpected Skills

There are a few unexpected skills that you might not have thought to cultivate:

- The Scientific Method: Software Engineering is an experimental science/engineering discipline.
- Logic and Reasoning: Most practical software is a complex set of systems with complex dependencies and expectations. Reasoning about structure is important.
- Soft Skills are more important than technical skills: Humans are more complex than software...by a lot.

How in the heck did I get here?

Education: Lessons Learned, Dues Paid

BS in Computer Science / Math and a MS in Math, but what lessons were learned?

Education: Lessons Learned, Dues Paid

BS in Computer Science / Math and a MS in Math, but what lessons were learned?

- Liberal Arts Education: An investment in soft skills

Education: Lessons Learned, Dues Paid

BS in Computer Science / Math and a MS in Math, but what lessons were learned?

- Liberal Arts Education: An investment in soft skills
- Don't Specialize too Early: Math taught me how to think in a structured way

Education: Lessons Learned, Dues Paid

BS in Computer Science / Math and a MS in Math, but what lessons were learned?

- Liberal Arts Education: An investment in soft skills
- Don't Specialize too Early: Math taught me how to think in a structured way
- Choosing electives is like investing in yourself

Education: Lessons Learned, Dues Paid

BS in Computer Science / Math and a MS in Math, but what lessons were learned?

- Liberal Arts Education: An investment in soft skills
- Don't Specialize too Early: Math taught me how to think in a structured way
- Choosing electives is like investing in yourself
- Go slow: Don't be in such a hurry to leave

Starting Up with a Startup

First job out of grad school and I was out of the frying-pan and into the fire

Starting Up with a Startup

First job out of grad school and I was out of the frying-pan and into the fire

- Negotiating: I was a well-educated moron

Starting Up with a Startup

First job out of grad school and I was out of the frying-pan and into the fire

- Negotiating: I was a well-educated moron
- Academic programming $\neq$ Software Engineering: Interesting $\not\Rightarrow$ Lucrative

Starting Up with a Startup

First job out of grad school and I was out of the frying-pan and into the fire

- Negotiating: I was a well-educated moron
- Academic programming $\neq$ Software Engineering: Interesting $\not\Rightarrow$ Lucrative
- Software floats on a sea of money: Learn about economics and business

Starting Up with a Startup

First job out of grad school and I was out of the frying-pan and into the fire

- Negotiating: I was a well-educated moron
- Academic programming $\neq$ Software Engineering: Interesting $\not\Rightarrow$ Lucrative
- Software floats on a sea of money: Learn about economics and business
- Startups are a mood, especially when they stop being startups

Starting Up with a Startup

First job out of grad school and I was out of the frying-pan and into the fire

- Negotiating: I was a well-educated moron
- Academic programming $\neq$ Software Engineering: Interesting $\not\Rightarrow$ Lucrative
- Software floats on a sea of money: Learn about economics and business
- Startups are a mood, especially when they stop being startups
- Leaving is a Strategy

Oil Industry Days

More Data $\Longrightarrow$ More Problems, but some lessons

Oil Industry Days

More Data $\implies$ More Problems, but some lessons

- Careers are an optimization problem: Specialize to the area which maximizes your interests, difficulty of entry, and usefulness in making revenue

Oil Industry Days

More Data $\implies$ More Problems, but some lessons

- Careers are an optimization problem: Specialize to the area which maximizes your interests, difficulty of entry, and usefulness in making revenue
- Rule of Thumb: Be close enough to the revenue stream to get your toes wet but not drown

Oil Industry Days

More Data $\implies$ More Problems, but some lessons

- Careers are an optimization problem: Specialize to the area which maximizes your interests, difficulty of entry, and usefulness in making revenue
- Rule of Thumb: Be close enough to the revenue stream to get your toes wet but not drown
- Legacy codebases are like archaeological digs filled with dynamite: make changes with great care

Wash, Rinse and Repeat: Sisyphus has nothing on me

Consulting: That Billable Hours Grind

Consulting in a hot Silicon Valley startup is still...consulting

Consulting: That Billable Hours Grind

Consulting in a hot Silicon Valley startup is still...consulting

- Consulting: sometimes your job is to be the neck to choke

Consulting: That Billable Hours Grind

Consulting in a hot Silicon Valley startup is still...consulting

- Consulting: sometimes your job is to be the neck to choke
- Everything is negotiable

Consulting: That Billable Hours Grind

Consulting in a hot Silicon Valley startup is still...consulting

- Consulting: sometimes your job is to be the neck to choke
- Everything is negotiable
- Pigeonholes are just suggestions

Consulting: That Billable Hours Grind

Consulting in a hot Silicon Valley startup is still...consulting

- Consulting: sometimes your job is to be the neck to choke
- Everything is negotiable
- Pigeonholes are just suggestions
- Sometimes you can do everything right and still fail

Stripe: The Present

And now you're caught up...

Stripe: The Present

And now you're caught up...

- Go where you're needed most (and where they're willing to pay)

Stripe: The Present

And now you're caught up...

- Go where you're needed most (and where they're willing to pay)
- Different companies have different professional cultures

Stripe: The Present

And now you're caught up...

- Go where you're needed most (and where they're willing to pay)
- Different companies have different professional cultures
- Ambition can only take you so far: Knowing when to stop...

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer
- Soft Skills become more of the job as your career advances

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer
- Soft Skills become more of the job as your career advances
- Develop thick skin against firing and judgement

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer
- Soft Skills become more of the job as your career advances
- Develop thick skin against firing and judgement
- Learn how to grow within a company

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer
- Soft Skills become more of the job as your career advances
- Develop thick skin against firing and judgement
- Learn how to grow within a company
- Know when to leave: Moving every 2 years early on is pretty good strategy for many

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer
- Soft Skills become more of the job as your career advances
- Develop thick skin against firing and judgement
- Learn how to grow within a company
- Know when to leave: Moving every 2 years early on is pretty good strategy for many
- Know what you like and what you don't. Do the former, don't do the latter.

Career Guidance: A recap

Things I wish I had known when I started:

- Learn how to negotiate
- Demonstrated curiosity and interest is the most desireable quality in any software engineer
- Soft Skills become more of the job as your career advances
- Develop thick skin against firing and judgement
- Learn how to grow within a company
- Know when to leave: Moving every 2 years early on is pretty good strategy for many
- Know what you like and what you don't. Do the former, don't do the latter.
- Learn how to interview (it changes over time)

Questions

Thanks for your attention! Questions?