

NO DEGREE, NO PROBLEM

Your **road map** to getting hired as a software developer in 8 months



JOSH KEMP

No Degree, No Problem

Your **roadmap** to getting hired as a software developer in 8 months

By Josh Kemp



2013, all rights reserved. No Degree, No Problem.

www.joshuakemp.blogspot.com

DEDICATION

This book is dedicated to my amazing wife who supported, encouraged and sacrificed so that I could chase my dream. I Love You.

Also to David Bock who despite being a partner at a very busy and successful software development company: "Code Sherpas" still took the time to reach out and help a young guy who needed a mentor and take me under his wing, teach me the ropes about Ruby on Rails and Software in general. David, thank you.

THANK YOU'S AND SHOUT OUTS

No one is an island, and **NO** one makes it on their own. I am not the exception to that rule. IN fact I would say I needed more help than most since I was coming from such a totally different background than software in general. All the people listed are only names to you but to me they are hours spent learning together and asking questions and figuring out problems. Without them I wouldn't be hired and I certainly wouldn't be writing this book.

First and foremost I want to thank the good Lord for keeping me safe while shoeing thousands of horses over the past 7.5 years and letting me get out of that dangerous career path and into a MUCH safer one. I came out with only having broken a couple fingers and getting kicked about 20 times in my career. I am very grateful to have not had more serious injuries.

To my wife, I could not have studied as much as I did for 9 months without her having to take more than her fair share of taking care of my 2 little boys. She watched them while I was out either working or studying. You are the perfect mate for me :-)

To David Bock for always being "available" to me and not saying no even if it meant meeting early in the morning to work out DNS issues and meet quickly for coffee. David was and is the perfect example of what a mentor should be like and a great ambassador for Ruby on Rails and the whole

software community in general. Without him taking the time and effort baby stepping me through problems and explaining complex concepts in a simple way, I would not have gotten as far as I did. I only hope that whoever reads this book can find a mentor half as good as mine was.

The whole "Code Sherpa" team really took me under their wings and made me feel welcomed. They helped me out when at Meetup groups and all around were nice to a "noob". I admire not only the team's abilities but also their kindness to new up-in-comers in the field.

Thanks to Arlington Ruby for being such a great place to come and learn every 2 weeks. Jim Gay, Chris Sexton, and Sean Marcia. You guys always do a really amazing job of teaching/helping new Ruby guys/girls feel welcomed into the group.

Thanks also to NovaRUG for hosting an awesome Meetup group and hosting "Coderetreat" last year. Gray Herter does a great job helping everyone feel welcomed. Andrew Wagner who used to co-host the group and, has since moved on, was super nice in helping me get my first machine set up right for Git and how to do a 'git clone'. Jason Wieringa, an awesome guy and great developer who helped me many a late Sunday afternoon at Panera answering all my annoying "noob" questions. Jason has a really neat skill of being able to break a large problem into smaller more easy to manage/comprehend sub sets of problems.

Richard Lewis of Wales who helped me with the remake of the www.wellspringlawn.com site I made. He not only is a really skilled talented developer, he also took many hours of his time to EXPLAIN stuff to me and encourage me to ask questions. He helped me not worry if some developers are mean when you ask beginner "noob" questions. I've never been to Wales but I hope to one day and now have at least one friend there. :-)

Kalimar Maia, I am SOOO glad we got lost together in the wrong building and made a connection while riding up and down elevators looking for the right place. I am so grateful for him meeting with me for coffee which eventually led to me getting hired at an awesome company.... ZipList! Thanks Kalimar!

Matthew Gallagher and Colin Macdonald of Arlington Code Shop a cool Meetup group in Arlington. Matt showed me some many cool things I had NO CLUE about like: "Why Javascript can be hacked easily" and things like: "How to make a magnetic rails gun". Colin showed me so many things like how to "grep" something and how everything is virtual e-file, even operating systems!. He didn't quite convert me to Python or Linux but I learned that I want to be a true software "craftsman" and not just an average developer.

The Whole ZipList team which would be too long to mention everyone, just suffice it to say that I am having an awesome time learning from such a great bunch of people. ZipList really does have an development team!

I know, I knowI didn't mention EVERYBODY!....if I did, we would have no book, only a list of thank you's which I don't think most people would enjoy. However there are about 1,000 more peeps who helped me out in some way shape or form and I thank

you....you know who you are:)

TABLE OF CONTENTS

DEDICATION

THANK YOU'S AND SHOUT OUTS

TABLE OF CONTENTS

OVERVIEW

What's the Goal?

MY STORY

SECTION ONE

The secret that most college students don't know about

Why you can beat out college students

5 reasons why a company will hire you

SECTION TWO

What do I have to know, What should I learn?

3 easiest and most common ways to get hired

How to know which language to pick

SECTION THREE

Must have skills

Roadmap Overview

SECTION FOUR

Street Cred

Introverts

Learning Timeline

Networking

Blogging

Resources

SECTION FIVE

The Roadmap

Resources

Final Thoughts

OVERVIEW

Problem: You want to be a software developer but don't have

1. A CS Degree
2. No IT Skills at all.
3. Hardly know what a "program" is
4. A college dropout with lots of debt
5. Have a degree but nothing related to technology
6. Are currently working a "blue collar" job but would like to switch over into the cool world of IT
7. Don't want to go hugely into debt for college and spend 4 years of your life
8. Are married with kids and don't have the time or money to go back to college but want to switch careers.

Solution: This book was written specifically for you.

I myself had 2 kids, a wife, no degree, a very tough job (shoeing horses full time everyday), knew VERY little about IT at all but wanted to make the switch and didn't know how. It took me 9 months and 2 days to go from zero to getting hired and paid \$70K as a junior developer. My goal for you is with this book (and depending on what you know about IT already) you should be able to cut the time down to 5 - **8 months** no problem, if you are willing to do the work.

FEARS

- 1.No one will hire me without a degree.
2. Even if someone DID hire me they would probably pay me hardly anything.
3. I don't know where to start or what I need to learn.
4. Doesn't becoming a "software developer" take like 10 years?
5. All the jobs say they REQUIRE a 4 year degree PLUS experience!
6. What about job security?
7. Without a degree you won't have job stability.

CURIOUS?

Well then how did I get hired against the odds, get several job offers and am making \$70K a year with almost 4 weeks paid time off with no degree?

What's the Goal?

It would be ludicrous to think you could become a full fledged "software developer" in 8 months. Here's the deal, I'll show you how to get **STARTED** in a career as a software developer.

My promise to you: Read this book AND do what it says and within 8 months you **WILL** be hired as "junior developer" making between \$45K - \$75K.

You will not be a full blown "software developer" BUT hopefully once you are hired and working somewhere you will take the initiative to learn more and be more. So then hopefully in 5 - 10 years or so you **will** be a kick-butt awesome developer.

In order for your beautiful career to happen, you have to first get hired at a "junior" position then a cool thing will happen....you will actually get **PAID to LEARN!**

That my friend is a happy day when you stroll into work and make really good money. Once you are ready to head home you walk out to your car satisfied and happy. You spent your day and time doing something you love while making an income for your family & also having learned a TON of new things.

That is my hope for you reading this book, this is just the start. The start of a journey that can lead to great things. I hope you take it :-)

MY STORY

My wife had just given birth to our second and last son Ian, it was a very hard birth and she lost a LOT of blood. I had to take off a week of work which caused two clients to fire me. The clients who fired me amounted to about \$800 a month loss. It was the dead of Winter and not a good time to be losing money or a good time to try and find new work shoeing horses. I pulled up to my last stop of the day, overcast and cold 27 degrees with a good breeze. I started to work on the horse and carry on a conversation with the owner. All of the sudden the horse spooked and kicked, which broke my right thumb. I finished the horse and got paid my \$45 and went home. I remember driving home getting depressed thinking: "This is SO hard with Elisha not doing well and having 2 little kids and not a lot of work and now my thumb being broke".

I took no time off and kept shoeing through the winter. I started thinking of other ways I could make money and be able to quit shoeing. Not a lot of options besides starting your own business when you don't have a degree. I started several different businesses that winter: A mobile brake repair service, drawing Caricatures at events and parties, buying and selling stuff, heck I even snow shoveled people driveways when we had a heavy snow and I couldn't go shoe horses.

It was a long tough discouraging Winter. I was grateful for the spring when I was really busy again with work. I went out in March to a barn that I knew to have some wild crazy horses and got my pinky broken on my left hand. My sweet wife Elisha insisted I get it looked at to see if it would heal back correctly. Both the pinky and thumb bug me from time to time to this day 2+ years later. My younger brother by 1 year Cody has always been the brainiac of the family. The nerd, math whiz, programmer I was listening to him explain to me that in order to do web design you didn't need to know any math (which REALLY got my attention as I sucked at math). I am very creative and artistic and thought it would be fun to make things on the computer.

I downloaded Gimp and tried to resize images. I tried to make a site or a graphic look nice but really didn't have a clue what I was doing or how to do it. I started to read more and found that in order to make better money (and to be really good) a web designer/front end developer should also know how to code, or at least be very handy at it. So I decided to learn the C# .NET Framework since I owned a Windows machine. Why not learn some sort of Windows based software? I signed up for a 6 month .NET \$1,000 online developer bootcamp.

Long story short, the videos I was supposed to watch promised to take you from a noob to a developer in easy steps. They had me designing and making my own database by week 2. To make matters worse the guy who recorded the videos had a THICK accent which made them almost **impossible** to understand. I stuck with the training for 2 months (despite getting stuck) and learned very little. I finally decided to call the whole experience a loss and think about other things I could learn. I had put the \$1,000 bill on my credit card which was rapidly gaining in amount of our debt.

Someone recommended Python online and a beginner's book to go through. I went through half of the book and decided that it wasn't for me. It was about this time that I read on dice.com about this new place in San Francisco called "**Dev Bootcamp**" and that in 12 weeks of training you could get hired as a Software Developer making \$78K a year! I called my wife and told her: " Man I wish I had the money to attend, I know I could do it." I called Cody (my brother, the math whiz) who is very smart but who had only ever been able to land a low paying retail job (despite having a 2 year Electrical Engineering Associate's Degree).

I told him about the school and encouraged him to go and to just sleep on someone's couch if necessary. Cody didn't like the lack of guarantees and the \$12,200 cost for only 12 weeks of training. I was so upset because I felt like he was blowing an awesome opportunity to go learn a skill and then quickly get hired at a really good rate. I kept thinking that if I could somehow get into the school and be able to afford the living expenses for 4 months (till I could get hired) I could do this. I talked to my wife who is always amazingly support in all of my crazy plans and schemes.

Elisha's words kind of shocked me a little, "Well honey, why don't you apply and try to get into the school?". I thought she would say "**NO way JOSE!**...that's too risking dragging our family all the way across the country!" So I thought about it and finally decided to make the 2-3 minute YouTube video which was required, and to fill out the application form. On the form it asked if you were accepted into the program would you like to pay the tuition all full up front or take a 3 year loan. I opted for the 3 year loan. I sent my video and application in and waited for a couple days.

I then heard back from Dev Bootcamp that they liked my video and would be inviting me to do a Skype phone screen/technical logic style test. I clicked the Google Calendar invite to find a convenient time, and to my surprise found that the following Tuesday was open or the other option was over a month later. Since it was Mid May and I wanted to get the ball rolling since the next boot camp started at the end of September so I choose Tuesday.

I learned a lot about interviewing that Tuesday and how defensive the interviewer was (who by the way was a brand new teacher) after having just graduated himself the previous month from the school. I passed the "logic questions" which was tough and very confusing because the interviewer used a very visual concept and on Skype everything is hard to see and the connection disconnected several times. Eventually I got what he was talking about and answered it correctly. I was asked if I had any questions and I had two. He instantly got defensive when I asked him his thoughts on going through the course and how he felt as a developer now. Long story short I was told they would be in contact with me shortly.

A week later I hadn't heard anything so I sent a follow up email to see if I was still in the running to get into Dev Bootcamp. I heard back a few hours later from a different person that replied and told me that they were sorry and had been busy but yes that I had now made it to the next phase of the interviewing process. It had already been almost 3 weeks since I had made my youtube video and sent it in.

I was sent another email soon after that said: "To complete the next phase of the interview process please go through and complete the "Michael Hartl Rails Tutorial" and then put it up on Github and send us the link. This should take you 1-2 weeks. Let me know if you get stuck or need help". They then listed a link to the 547 page "Michael Hartl Rails Tutorial". I opened up the link and started. I read about the terminal and then Googled "how to open terminal in a Mac". I finally opened the terminal and went back to reading the tutorial. I read things like make a "mkdir" and "cd" into this and "cd" into that and "pwd" if lost followed with "ls" to see the contents of your directory.

After an hour of confusion and opening things in the wrong places, making apps in the wrong directory and then not being able to find out where they were I went online and typed in "beginner's guide to using the Mac terminal". I soon found out that there are many different types of shells or terminals and I figured out that my terminal was running the 'bash shell'. I then read forums and articles on bash until I stumbled across the beginner's gold mine for learning the terminal: "Learn

the command line the hard way" by Zed Shaw. I began reading it around 1AM. I fell asleep on my bed and woke up in the middle of the night and put the computer away.

The next month was spent working on passing that tutorial. I ended up getting 4 or 5 of the 11 chapters completed and I would continue to get an error that I had no idea how to figure out. I ended up deleting the whole repo and then starting over from scratch. I finally decided after 3 weeks of staying up into the wee hours of the night (having shod horses all day) feeling depressed thinking that I wouldn't ever get accepted into Dev Bootcamp. I was taking so long to do something they had told me would take a week or 2 tops.

I started getting desperate and kept thinking how to speed things up and finally learned about the magic cheating method of "copying and pasting". I was having a super hard time with Git and Github. I wasn't sure what to write and exactly what was going on. But as I got discouraged I started thinking to myself: "I wonder if there's a way to take the tutorial code and just send it in." Then I thought well maybe I'll just copy and paste the big pieces of code to make sure I have no typos. So before you knew it, I made the 5th or 6th try of the app in like 2 days pretty much all by copying and pasting.

I had spent just over a month from start to finish. I was completely sleep deprived and not enjoying the whole process at all but I was SO glad for it to be over or so I thought. I sent in my email with the Github link and I was immediately told that they would be in contact. One week later I was told: "Congrats on finishing the Michael Hartl Rails Tutorial, we'd like to do another SKYPE interview with you. Please check our Google calendar for a date that works for you." I was SO bummed at this point because their website had said from the very beginning that anyone who applied would have a yes or no decision within 3 days. Here I was almost 2 months into the process and still having no clue what was going on. That wasn't the only problem though. Once again the only 2 dates that were available were a week from now which was next Tuesday or a Wednesday one month from now. Both of the appointments were at 1:30PM.

I now thought to myself: "This is just a mind game test to see who will continue on and make the sacrifice to get into the program." I decided to take the Tuesday at 1:30PM appointment because it was the soonest and because I thought maybe it was a test. If I picked a month later they would think I didn't REALLY want to get in. A week goes by and I'm sitting at my kitchen table at 1:15PM. I got onto SKYPE early in case they wanted to get on earlier. I was sitting nervously and waiting. I'd waited almost 9 weeks for this day and then my Droid phone **-dings** telling me I have a new email. I open it up and it's from an HR person at Dev Bootcamp saying "unfortunately we won't be able to make your appointment today but that I could select another time."

I was SO upset and when I looked on the Google calendar the next available slot was **10 MORE days away**.

Round 2: Waiting by the computer (now over 10 weeks) from my original 3 minute YouTube video and application. The person (who shall be nameless) logs onto SKYPE and says: "Hello". We talked for 30 minutes and he quizzed me on some technical questions on Ruby and Javascript and then a brain teaser. Finally after 30 minutes he said "Any questions?" It took everything in me not to say: "YEAH I have a question, am I in or NOT???" but all I said was: "Thanks for the interview, any idea on when I'll know anymore?" He said: "Uh...it shouldn't be too long."

3 days later I got an email saying: "Congratulations! you've made it into the Fall Bootcamp!...blah blah...blah. Just make sure to send in the **\$12,00 + dollars within 20 days** or else you won't have a

spot. See you this fall!" WHAT??? what about the loan? What about the "possible" scholarships I had read and heard about on their site? I had 5 days to make my decision. I had saved just enough money to be able to live for about 3.5 - 4 months with my family in San Francisco. There was NO WAY I could swing another \$13K on top of it all. I had been VERY clear that I could only go if I could get a loan which was an option on their application when I applied.

I waited 3 days then finally emailed Dev Bootcamp and very politely asked about the possibility of a scholarship. I received a reply about an hour later that said very matter of fact that this opportunity was for people who could pay upfront and if I couldn't pay they could put me on a waitlist and maybe if they had any spaces and if they had any slots left I could try and get one **but** no promises. I agreed to go on the waitlist where they instantly gave 60 hours worth of homework to do and then gave a timeline that of the 21 students on the waitlist, whoever didn't finish in time would be cut from the list.

I was SO depressed and discouraged, It was the end of July now I had spent my entire summer trying to get in, from the end of May till now, and there was no way to keep up with all the work without cheating. I decided what the heck, I'd gone this far why not a little more effort and maybe I would get a scholarship and beat out those other 21 people. All I did the last week in July was go to work, come home and study/cheat to get all 14 sections done in time. Which I did **JUST BARELY!** Whew! I thought, I MADE IT! I had seen that only 14 other people had finished in time so I at least felt good that the pool was getting smaller until the **NEXT** assignment came through. It was even harder with the same promise of one week to get everything done or you'd be cut when in fact they actually gave us only six days. I logged into the student prep-work section and finally decided to give up when I saw that there was not 14 students trying to finish this section (that I didn't even think was going to happen in the first place) there were in fact 31 students.

My heart sunk, I finally knew that I couldn't work hard enough, cheat enough or even pay enough money to get into Dev Bootcamp. I gave up on learning to code and I gave up for two days on virtual anything besides shoeing horses and being depressed. I felt like I had been misled and lied to. The entire Dev Bootcamp site had said **Absolute Beginner's Welcome** but then they were asking how much Javascript I knew even though I told them I didn't know any. In the end it had all come down to money and if you had it, great. If not screw you, keep shoeing horses. I was SO mad. I wanted to get out of shoeing horses but more so now. I also just wanted to learn to code and get hired as a developer just to see if I could do it.

Then God smiled on me. I sat for two days after resolving myself to giving up and my wife looked at me and said: " Don't be sad honey, don't give up, I know you can make it as a developer." My eyes watered up and I just looked at her and said: " I don't know how that's going to happen" My wife came over and gave me a hug and said, "I'm going to find you another school," to which I replied: "There aren't any other schools, I've already looked, I wish there was because I would try to get into them." Elisha looked for about an hour online and then she got excited and said: "How about this one!" I took the computer to look just to show her that she was wrong but in fact she wasn't! Two days before a new school had opened up in Manhattan N.Y. it was called **Flatiron School**. My heart JUMPED. The application process had just started yesterday and there was only one month until the new school opened for the very first time. The **Flatiron School** was being hailed as the "East Coast Dev Bootcamp" only much more affordable. It only cost students \$5,500 for the entire 12 weeks! I felt like I had swallowed lightning. I couldn't type fast enough to get my application in. The next

week for 3 days the school was doing all the interviews on Thursday, Friday and Saturday. You could either show up in person or do a SKYPE interview. One secret about me is that I hate to SKYPE. It just feels weird and I would much rather meet someone in person than on the computer. Elisha and I talked into the wee hours of the morning. Flatiron would only take 20 students. So I told her, "I'm not missing THIS opportunity! I think I should drive the 6 hours to Manhattan so I can interview in person! What do you think?" Elisha always being the awesome, supportive wife that she is, agreed. The trip would also set us back quite a bit in money from taking the trip with the cost of: Gas, Food, Hotel and loss of work. I decided to take my really good friend with me up to New York since neither of us had ever been.

I ended up dropping around \$400 dollars just driving up to NY. We arrived in Manhattan on Thursday at midnight and all I remember was all the cops, homeless people and the smell of garbage. We checked into our hotel which was the cheapest I could find at \$117 for the night. The room was small and smelled very musty and damp. I looked out from the 11th story floor and thought, "Is this where I'll be living in a month?" I couldn't sleep. I was SO excited/nervous for my 8AM interview the next morning so we went down to Times Square. I have to admit it was pretty incredible. There was so much going on even at 2AM. I finally drifted off to sleep that night around 4AM.

I was up at 6:45AM and we hustled to get dressed and quickly eat downstairs. The school was 2 blocks away so I wasn't too worried. I left my friend at the corner Starbucks and told him I'd be back in an hour. I got to my interview 15 minutes early. A small office with no sign to know whether or not I was at the right place. I assumed I was at the right place because downstairs there was a very nerdy looking asian dude who looked like he was in his early 20's. A young girl with thick glasses came downstairs minutes later. I asked her if she was interviewing for Flatiron. She said she was and asked if I was and where I was from. She seemed impressed that I had driven 6 hours for the interview. Just then a pale skinny nerd came down the stairs. The girl nervously wished me good luck and left.

I didn't know it yet but the pale skinny nerd was Avi Flombaum, the dean of Flatiron School. I just stood there not knowing if I should go upstairs or not, finally he said, "Are you here for the interview?" I said that I was and we headed up an old staircase that creaked and seemed very narrow or else I was just too heavy. Avi opened a door on the second floor and when I walked in, to the left of me, I saw the school for the first time. It was a long and narrow hallway and amazingly hot and stuffy for it only being 8AM. Adam Enbar stood up and introduced himself. He apologized for the fact that the a/c wasn't working and that the windows didn't open.

I sat down on a leather love seat. What was immediately odd was the fact that Avi was the founder and developer but Adam who was the untechnical partner was doing **ALL** of the interviewing and questioning. What made it even more awkward was the fact that Adam was sitting on a couch 180 degrees to my left while Avi was sitting on a chair 180 degrees to my right. I couldn't figure out if I should **ONLY** be talking and keeping eye contact with Adam or vice versa. I noticed Avi scribbled down notes in his notebook. Talking only to Adam felt rude but including Avi felt very unnatural. I was craning my neck back and forth talking to each of them at opposite ends of the room.

I was told by Adam that my resume was one of the most fascinating they had gotten and that he wanted to hear how I got into shoeing horses and then of course why I wanted to learn to code. I talked for a few minutes then Adam gave a brief background on himself and the school's plans and direction. Adam then moved into a few word brain teasers followed by Avi giving one brain teaser as well. They were very happy/shocked that I had driven all the way up the night before. They finished the interview

by asking if I had any questions to which I replied: "Only 1, what can I do to help my chances to get into the school?" They said that I had done well and that they were optimistic about my opportunity at Flatiron and told me they would let everyone know in one week who had made it and who hadn't.

I walked outside, after saying goodbye, to find my friend. I was sweating from being nervous/excited and also from the stuffy room. I spent the remainder of the day getting lost in Manhattan trying to see a few sites and finishing up the day by driving in rush hour the whole way home. The only consolation was stopping in at New Jersey Fried Chicken which was amazingly good:-). I then called a few clients I had scheduled for the next day and move them to later that week because I was utterly exhausted.

One week went by then 10 days. It was now almost September and class started on October 2nd at 9AM in Manhattan on 26th street. Here I was still shoeing horses and living in Northern VA. Elisha was (rightly so) getting VERY fed up with these schools and just the overall uncertainty of where we'd be living and what I'd be doing. I decided to send Adam an email politely asking if he had any idea if I was still in the running or not. He immediately emailed back apologizing about the delay and reassuring me that I was **DEFINITELY** in the running and that things were looking very promising and that he would give everyone an answer by the end of the week. I said okay no problem. Elisha was thoroughly freaked out at this point and I couldn't say I blamed her although I was feeling the weight of all the stress. I didn't care how tough things got, just as long as I got into Flatiron and had the chance to become a developer.

The end of the week passed, and still no word. I emailed him and politely told him my dilemma. I needed to know because I still had to get a place to live in New York and move my family. He assured me that I was still in the running and that I would know soon and that they had just had WAY more applications than they had planned on. Things were getting desperate at home because our lease was almost up and I didn't know what to do. So we started packing stuff up and giving away a ton of stuff we didn't need anymore so that we would be ready to move the instant we knew.

Finally I said that we should take a gamble and risk it and go up to New York and just find a place tomorrow. Elisha agreed and said that she would feel better if we had a place. We left at 3:30AM. You know the ending to this story without giving any more details about all the heartache and money we lost on the \$1,700 deposit (that we never got back) or the fact that I was told the day **AFTER** I signed a lease. Which was also one week before our lease in our current home was up that unfortunately I had made it down to the top 30 finalist but didn't make it into the final 20 people selected for Flatiron. I can't describe the panic/fear/desperation I had. I sent several emails practically begging to get into the program and even told them I had ended up getting a place all to no avail. October 2nd came around and I wasn't in the group. The only thing that was different this time was that I wasn't nearly as depressed as I had been before. Now I was **MAD**. I spent the next month and a half trying to get into different schools all the way from Boston to Denver to London. All of the schools were the same basically without all the money \$10-15K up front, we don't care about helping you.

I spent the next several weeks re-grouping and getting my business caught back up. I had kind of neglected it for the past 5 months and had lost some clients. I also moved my family to a nice 2 bedroom apartment in a small town near where most of my clients lived. Finally one day I came home from shoeing all day and getting kicked pretty bad by a horse and with my game face on told Elisha: "don't give a #@*^! I am just going to teach myself. Screw these schools, I don't care if it takes forever I am going to teach myself and get hired!" My wife was equally enthusiastic and said " Go for it! In

fact I think you should start a blog so other people can see what you're doing and maybe be encouraged to learn on their own.”

Well you know the rest of the story. I am currently writing this on my lunch break from the 7th story building at ZipList where I was hired just 5 weeks ago for \$70K as the **Partner Integration Specialist**. Yes, dreams do come true if you work hard at them. This is the very roadmap I wish I had when I was trying to learn to code. I hope to **inspire, encourage, challenge**, and above all **help you** become AND get hired as a developer as soon as possible!

Section One

THE SECRET THAT MOST COLLEGE STUDENTS DON'T KNOW ABOUT HAVING A DEGREE

1. The one thing that everyone must have and cannot get hired without, is EXPERIENCE! (disagree or not)
2. Even with a degree (when you are looking to get hired) EVERY job description asks for at *least* 6 months - 2 years of relevant skills/experience. There is NO getting around that requirement with or without a degree.
3. Most college students don't get taught very much (or any) "version control", like Git, and most college students learn very little **if any** Html/Css.
4. Most college students think they are "GUARANTEED" a job just because they have a degree. That may have been true years ago but certainly not true since late 2008- early 2009 as the economy tanked and employers wanted to pay for skills and not your degree.
5. Once you are hired you still have to pay back a massive amount of school loan debt. With my "roadmap" you can get hired for well under \$1,000 total investment.

WHY YOU CAN BEAT OUT COLLEGE STUDENTS

(THE NEW WAY TO GET HIRED WITHOUT A DEGREE)

In Software, remember the Golden Rule: "**He who has the Gold RULES!**" but in our case just replace the word "Gold" with "Skills", and then it's true.

1. Skills are the great equalizer for everyone.
2. Colleges are several years behind when it comes to newer technologies. Take Ruby on Rails for example, I couldn't find a "college course" to take to learn RoR here in Virginia. The closest one I could find at the time was in Texas.
3. If a company needs a newer/emerging technology, recent college grads are not typically the best choice. So as far as "newer" tech skills, a college degree really doesn't matter.

As a rule: the newer the skill the less important a need for degree is required.

4. Real IT skills are better than degrees and certifications. Most certifications you can buy all the test answers for \$69 online and can be easily memorized with some simple memory tricks.
5. Being truly "remarkable" is your secret weapon to getting hired. If you are still trying to find security in degrees then this book is not for you. You have a choice to make. You can either be a "lemming" and go back to school like everyone else or decide that you will put in the "sweat equity" work and do the things and take the steps that are necessary to stand out as a junior developer. Remember skills and projects are the new currency. Your security comes from knowing a few skills

well and not in having a generic degree.

6. Your blog is your living breathing constant resume at all times. Contribute and add to it every week to make it constantly evolving and changing. It should be reflecting exactly what is happening to your own self.

5 REASONS WHY A COMPANY WILL HIRE YOU AS A JUNIOR DEVELOPER

(EVEN WITH NO DEGREE)

1. The secret that no one will tell you or admit is that most junior/entry level IT jobs can be learned in the first 6-9 weeks and don't require high level knowledge or a degree.
2. Most junior positions will be boring (at least at first). You will usually have to do some of everything like: Support (answering emails), Testing, Html/Css, General bug/trouble shooting and very little actually making of cool features and coding.
3. A lot of junior developers (that many will not admit) end up doing far more emailing/marketing for the company than actual coding. When things are slow you will probably be called upon to do tasks like: "Check the printer ink.", "Oh hey we're out of batteries..do you mind running down to Staples?" and "Hey, we're out of Cokes in the fridge, let me show you how to order those."
4. Employers would rather have: "**A hard working C student**" than: "**MIT Grad who is bored in 5 minutes and wants to know when he/she can get a raise**".
5. Employers want people who are: Loyal, hardworking, enthusiastic, easy to get along with and who really want to learn.

Section Two

WHAT DO I HAVE TO KNOW, WHAT SHOULD I LEARN?

Here's the Secret: You can't learn enough in 8 months to really be good at everything. With that being said you most certainly CAN learn enough to get hired in 8 months.

So what should you focus on?

You could spend two years just learning HTML/Css stuff and then after two years be able to design and make a beautiful custom website from scratch. The problem is do you want to be a

Web Designer or Web Developer?

You could spend years just learning a programming language, like Ruby, and get to a solid understanding. But what about Rails? The sad fact is without Rails most people don't want Ruby. You could learn Back End stuff, design and make your own relational database like I tried using MySQL. What about Git? What about Testing? Agile? Scrum? Javascript? Holy crap there is NO WAY you can learn all these languages and databases and get hired in 8 months.

AH...but the beautiful part is you don't have to. That's right. Our goal is to get hired in a **junior position**, whatever that might be.

3 easiest and most common ways to get hired in an Entry Level Developer position

1. Become a Q&A (**quality and assurance tester**). Most of these jobs don't pay that amazing but the one I was offered paid around \$40 - \$45K. If you want to get one of these jobs the barrier is low. You need to learn some of the basics about software but focus on learning a testing framework like: Cucumber, Rspec, or something like that. **Drawbacks = Lower pay** and you can get stuck in this position for a long time and not have a very good way out.

2. If you know some HTML/Css you can be a support guy/web designer/bug fixer/tester/fill_in_the_blank developer. If you are somewhat artsy and can make stuff move around a page you can get hired to fill a need the project has at the moment. **Drawbacks = You won't be truly "coding" at least not right away.** If you have ZERO artistic design talent and hate Css then this is probably not a good path for you.

3. You can really be hired as a legit developer and start writing code on day 1. However, typically even a developer will spend the first 3 months or more doing very boring small tasks and testing/bug fixes and think: "When do I actually get to write code". **Drawbacks = If you are weak at all in programming logic** you will have an enormous amount of stress/pressure on day one. If it is your strong suit go this route.

The **ROADMAP** outlined in this book will show you EXACTLY what you need to learn so that you can land a job in 8 months. It's good to know that there are a couple slightly different ways to get hired and to see which fits you best.

HOW TO KNOW WHICH LANGUAGE TO PICK

My **roadmap** will work for Ruby on Rails and it will work with other languages as well. Here's the caveat, you need to pick a language which is fairly in demand but also a language that can be learned fairly quickly.

Don't pick a language that only wants to hire senior developers. Don't pick a language that was written before time began. Don't learn the language assembly and then be surprised when there are no junior positions available.

I strongly recommend Ruby on Rails for the following reasons:

1. It's one of **THE fastest growing languages** and one of the most IN demand languages that there is
2. Ruby is fairly straight forward to read and is a "high level" language. Which just means it's easy and you can do more faster.
3. The framework Rails is super easy to get started with and there are lots great tutorials on RoR.
4. The RoR community is the best software community for "noobs" bar none. Most RoR developers are VERY nice and helpful to beginners.
5. RoR jobs pay really well right now. I don't suggest you do something JUST for the money BUT if you have a family (like I do) and need to switch careers to a field that will pay the bills, RoR is a good one.

Follow the ROADMAP and you won't regret it!

Section Three

THINGS YOU ABSOLUTELY MUST HAVE

1. A computer.

NOTE: If you only have a windows machine and are really tight on money, I TOTALLY recommend getting a \$300- \$400 used Mac on Craigslist versus a Linux or Windows.

Ruby on Rails works best with Mac.

2. Internet connection. (**preferably fast**)

3. Time. (**3 hours per day or 21 hours per week**)

4. \$49 for the "One Month Rails Course" by Mattan Griffel.

5. \$25 for 1 month of Codeschool.

6. \$25 for "Learn To Program" by Chris Pine. (**you may be able to get it used cheaper**)

Recommended: You MAY be able to get by with some OTHER Ruby book however this one is VERY good. **Beginner to Professional Ruby** by Peter Cooper pretty pricey, I believe it is \$50, but good.

Those are the only items that you **CANNOT** do without. Sure there are other books but these you cannot do without.

Roadmap Overview

Networking

Go to every Ruby on Rails Meetup group. I don't like the word networking. For me it conjures up images of aggressive people only on the lookout for themselves and not

caring about others. What I mean by networking is: **BE NICE!** There that simple. Don't only be nice to certain people who are hiring managers, be nice to **EVERYONE**, don't be rude to "noobs" like yourself. DO help everyone you can. Be genuine. Offer to help in any way you can whether it be helping other beginners with code or by getting a chair if the

room is full. It could even be agreeing to go stand and do door watch. The whole point of going is to get to know developers, build rapport and to start to hear and pick up the lingo. By just hanging around developers you'll pick up and learn a lot of information.

Speaking 3-4 times

At 4-5 months you want to start giving small lightning talks that last 5-10 minutes long. Do at least one per month, all though I would do as many as possible. This is how you get hired. This is exactly how I landed my job. Speaking is very hard and you'll feel like you're about to pass out, **speak anyway**. Instead of going around the room trying to meet everyone, shake their hands and try to make an impression you can speak for 5 to 10 minutes. Afterwards everyone will want to know who you are what you're about and what you're doing.

Blogging 3x Weekly

Please note, blogging and tweeting don't count as studying-sorry. Lunch breaks are now your blog

writing time and red lights are twitter time. You don't have to blog about all the nitty gritty. Just blog about something you learned, like or hate about how things are going. **3 Keys, be honest, open and consistent.** I didn't start getting lots of comments and emails until I had been studying for almost 178 hours. After 178 hours it

seemed like someone flipped the switch and people started to take me seriously. They began to check in on me and ask questions and email. That's also when I first met my mentor, David Bock, who only found out about me from his co-founder of the company, Karen, who just happened across the blog one day. She sent David an email to "keep an eye on this guy". You never know how it will turn out. I also was contacted by a company soon after I met David about possible opportunities. Which although it wasn't a good fit then, I did get to meet with their senior developer for coffee for an hour at Wegmans. He gave me some awesome advice. I have probably had well over a hundred emails from around the world and probably 20-30+ people asking to pair program with me, **all because of my blog.** I also was offered two jobs on day 1 of my own personal website launch. Which was exactly at months.

Studying 21 hours Per Week (827 total)

Study 3 hours per day minimum. If you do as outlined here in this book there is NO reason you shouldn't be hired by or before the magic number of **827 hours!** You must keep a log in order to be honest and not skip your studying. If you don't put in the day, forget it. You will never learn how to code if you don't actually code. So many people just want to spend their "studying" time liking stuff on Twitter and think they are somehow going to magically be good at coding. There is this huge movement of "learn to code, change the world" which is great but it also paints this picture that after you type in the commands 'rails new app' you know how to code and that someone will

now hire you. Learning to code is a lifelong journey. You have to love learning and be willing to put the time. Lots of these **learn to code startups** try to convince you that you can never learn on your own and that you need to pay \$3,500 for a 45 hour online course. Which basically teaches you how to make a VERY basic rails app. Learn the hard

way which is also the best way. You will actually **OWN** your knowledge after learning on your own. Make your own **coding bootcamp.** Pair program with people all over the world every week and keep yourself accountable by putting all your code up on Github or Bitbucket. Blog about your success and struggles and before you know it you'll be snatched up by a company as a junior developer making great money!

Remember: Don't do this for the money. The goal is not to get a high paying job one day and then to shut your brain off and learn/do as little as possible. NO! My goal was to get out of my current tough, dangerous, VERY physical job and make enough money to take care of my family then get paid to keep on learning. If one day I'm an expert I'm sure money will follow. Don't chase the money. Chase the knowledge and coding skills :-)

Why Hours?

I met a guy at last year's Coderetreat who said he had been programming in Ruby for 6 months. At the time I had been learning Ruby for about 2-3 weeks. I was shocked by his lack of skill. I asked him how many hours he studied per night in the past 6 months and he replied with probably 45 minutes-1 hour most nights. That's only roughly 2 months of training, not nearly as impressive. That's why I

recommend and use hours as a kind of bench mark. When job postings say 3 years experience, what exactly does that mean? Working 40 hour weeks that means thats roughly 6,000 hours. I find that hard to believe. I myself got hired after 827 hours, you can do the same.

Pair Programming 2x Per Week

Pair program as soon as you can. The earlier on you can do it in your learning journey the quicker you'll get the basics down and not waste too much time stuck on errors. If you can't get anyone put it up on your blog and twitter that you would like to pair program

either remotely or in person. The good news is before long you will have a ton of people asking **YOU** to pair program a couple of times per week. Don't pair program **TOO** much, no more than 3 times per week, unless you're programming with a **VERY** senior developer. If you pair program too much you won't have time to specifically study and learn on your own. Pair programming is a great way to get unstuck and make sites together. You can go through a book together, especially when you are brand new. But with programming at a certain point you just need to start to learn [How to Learn](#) and not rely too heavily on others. Learn to read basic documentation.

How to Pair Program?

So how do you pair program. What do you do? What are the tools you need, who leads? what if I suck? Answer? Who cares!! Agree what it is you want to: **work on, go through, or make and then stick to it**. Don't let one person do **ALL** the work. Be nice, don't make fun of the other person but let each other struggle a little bit so that you will both walk away with some new knowledge that **YOU** actually now know. If you just want the other person to do all the work then just go and clone a repo off of someone else's repository.

What to Study?

My main question I get asked in emails is: What resources I recommend and what should I study or work on. I admit that I wasted a **TON** of time flipping between resources and looking for the definitive **gold mine** resource that would get me hired tomorrow. The truth is yes some are better than others but not doing any is no good. Learning to code takes a long time. I am just trying to get you ready to get your first junior entry level position somewhere you can then get paid to grow your skills. I have a roadmap for successfully getting you hired as a junior developer as long as you stick to the program and do the work. This roadmap will take out **ALL** the guess work and wasted time of googling for new and better courses and who thinks I should take this or that and whether or not it is any good. I am living proof that you can go from **ZERO** programming knowledge, working your butt off, and 827 hours later land a great paying junior developer job. I used to shoe horses everyday for a living. I barely used email. Trust me if I can do it, you can do it. All it takes is **COMMITTING** to the plan and a lot of diligent hard work. If you think all you need is to go back to school and get heavily in debt to get hired somewhere, this book is not for you.

How to Study

Usually when you don't want to and are tired :-)) You #1 probably want to find an accountability partner or connect with a study buddy who will hold your feet to the fire when it comes to consistently studying everyday. No days off. You can't stop for the whole football season and then wonder why you are learning at a snails pace.

How to Study in a Nutshell

sample content of No Degree, No Problem

- [read *Without Mercy \(Sean Dillon, Book 13\)* pdf, azw \(kindle\)](#)
- [download *Vengeance of Orion*](#)
- [click **Cathedral, Forge and Waterwheel: Technology and Invention in the Middle Ages** pdf, azw \(kindle\)](#)
- [download *Wordsworth: A Life* pdf](#)

- <http://tuscalaural.com/library/The-Forbidden-Forest.pdf>
- <http://creativebeard.ru/freebooks/The-Untold-History-of-Japanese-Game-Developers--Gold-Edition-.pdf>
- <http://yachtwebsitedemo.com/books/Chinchillas--Animal-Planet-Pet-Care-Library-.pdf>
- <http://honareavalmusic.com/?books/Memoria-de-mis-putas-tristes.pdf>