PREFACE

It's 3:02 AM

You wake up in a cold sweat.

Again.

You can't sleep. Again.

The thought of it all is eating you up inside. It's keeping you up at night and it's getting worse. You remember that viral video of that robot that does backflips. Then the one about that surgeon robot that can sew a grape skin with inhuman precision. And then it all comes back. The thought of those robots replacing millions of jobs, including your own.

You love your job and you're good at it. But it's one of those low-tech jobs that is at risk of being automated and replaced by cheaper, more performant, intelligent machines.

You walk over to your kitchen and pour yourself a glass of water. The children are sound asleep. Peacefully dreaming of sweet nothings, the kind only children can dream.

What if?

What if you do lose your job one day? What would you do? What could you do? How would you provide for your loved ones? And what kind of world would your little ones grow up to experience?

A machine world?

You know, deep down in your gut, that your family and millions of other families across this round world of ours - are all at risk. You gulp that lonely glass of water, in that dimly lit kitchen and slowly tiptoe your way back to bed. Tomorrow is another day. Maybe you're wrong. Maybe there's nothing to worry about.

Welcome To The Machine Era

Unfortunately, there is, my friend. Studies show that hundreds of millions of jobs are at risk in the next decade and a half. It's not science fiction anymore. Artificial Intelligence has grown by leaps and bounds and we can now expect intelligent

robots to enter the workforce - en masse. Every human task that can be automated, will be automated. And absolutely every low-tech job is at risk. You have sound reasons to worry. Whether we want to admit it or not, the Machine Era is here and it's here to stay.

But hey, cheer up. There's good news after all.

There is a plethora of jobs that will come with this global transition to the Digital Economy. In fact, every business is demanding Digital Innovation Solutions - as in yesterday. If businesses lack an online presence, a mobile app and an overall digital strategy, they're in trouble. And they know it. The demand for quality high tech work is skyrocketing. Even as demand for low-tech jobs is declining.

I hear you. Software Development is for those young Silicon Valley kids. You're in your forties. You've got kids of your own. A mortgage. And a sore lower back that gets horribly stiff when it rains. Or maybe you're a newly landed immigrant trying to make ends meet, with no discretionary time for learning anything new. I hear you.

And yet - listen to me. If we stand any chance whatsoever, even a tiny hope, of living in a human-centered world, in the near and distant future - then we can't do it without you. Listen to me carefully.

The Machine Era will keep marching on – with or without our consent. But if you, me, your children, my children, and the rest of humanity stand any chance of inhabiting a world that is human friendly and worth enjoying, we have to get our act together and start becoming Tech Builders, and not just Tech Consumers.

If you have an online account, you use technology. If you send emails, you use technology. If you read news online, you use technology. We all use technology. But how many of us really create technology? How many of us build? I'll tell you how many. Less than 30 million of us. That's how many. That's how many Software Developers open their laptops every day and build software. Versus billions of other human beings on this planet who use the software the rest of us build.

Redefining Human Work

The only way we stand a chance to get some sleep at night in this brave new world, especially you, my low-tech worker friend, is if we have, not 30 million, not 300 million, but a billion Software Developers out there. People like you and me who boss machines around and command them to act in our interests. People like you and me who tap into our authentic human intelligence to create human-

centric Software Products that make our world more human. More worthy of our children and our children's children to enjoy.

Can anyone learn to code? Can anyone actually become a Software Developer? No, not even close. But the ones that want it badly enough - can. And here's the fun part. If you are willing to embark on this insane journey of becoming a Software Developer, here's my promise to you. Not only will you be able to finally get some sleep at night and stop worrying, but you will get to meet others like yourself on this journey.

We're building a movement and we're calling it Carmel.

A movement of committed people like you who are willing to level up their skills and become tech creators, builders, makers.

And I, and many others like me, will be there to cheer you along, to support you and to do whatever it takes to help you make it. And you will. If you want it, you will make it. You will work as a Professional Software Developer. And you will build amazing products that the rest of the world will love to use.

This book is my gift to you as you begin this journey.

I will share my personal Developer Journey with you. And I will open up my entire heart and my whole mind in the pages that follow with one single purpose. One single purpose. For you to decide, truly decide, before you reach the last page, that you will do this. That you will commit to this. And that nothing will stop you until you start your new career as a brand new Software Developer.

I promise you this. If you make that commitment to yourself - the commitment to learn - I will meet you halfway and I promise you on behalf of the entire Carmel Community that we will all pitch in, our energy, our knowledge and our time, to support you - start to finish. And beyond.

Come. We've got a long journey ahead of us. But one that will be worth every single minute you spend learning. It's time to flip the page in your career.

Let's do this.

CHAPTER 1

I Could Have Ended Up A Street Hustler

We had settled in Toronto, Canada for about five years at that point - the year was 1999. This was our first trip back to Romania since we had left and it was an emotional one to say the least. My parents, my brother and I. We had travelled back to where we grew up, in Bucharest, and even reconnected with some of the people we grew up with. One of those people was my best friend growing up in Bucharest when we were kids. We used to do everything together, even fight together, get in trouble together, be scared together and just literally spend hours of our day roaming the streets of Bucharest in the 80s and do what kids used to do back then - nothing and everything.

This time, our worlds were so far apart it was as if we lived on different planets, not just different continents. My friend ended up stealing my dad's cigarettes when we weren't looking and then ran away. That was the extent of our bonding after not seeing each other for years. That was it. Him stealing from us and running away. He had apparently dropped out of school and gotten into a lot more trouble than we used to get into as kids. He was a bonafide Bucharest street thug and a pretty famous one at that - as we found out later on. Apparently, we were lucky we just lost our cigarettes and didn't get stabbed.

I Could Have Ended Up A Thief

A few years before, I was a young twelve-year old playing ping pong in a refuge camp in a small town in Belgium, called Hastiere. I was just learning ping pong, but I was quickly rising through the informal - but nonetheless meaningful - ranks of the refuge camp ping pong tournament. I was the only kid in the tournament, but I had a secret weapon. My secret weapon was my teacher. He was a soft-hearted mean-looking African dude in his thirties, who was a ping pong master. He used to live off hustling people and betting - and winning every time of course. My ping pong teacher saw me hanging around the ping pong table a lot and took me under his wing, teaching me his own flavor of street wisdom - that I still remember to this day. He was the one who introduced me to the concept of full mental concentration on the task at hand. He was so focused when playing ping pong that you'd think he went into a trance. I was fascinated. My brother and I and two African kids younger than us were the only children in a large refuge camp of Eastern European and African refugees. We had no friends our own age. Whatever we learned, we learned from adults twice or triple our age. And living

a refuge camp, a lot of those things weren't always about pretty unicorns flying over cute rainbows, fluttering their mythical little wings. No, it was mostly about figuring out creative ways to stay alive and not starve.

My parents had been living in that camp way before my brother and I joined them, later on. I was twelve and my brother was a year younger than me, when we had travelled for days all the way from Romania to Belgium, forging our identities and illegally running away with two brave strangers, a Belgian man from Hastiere and his wife, who pretended to be our parents. I will be forever indebted to them for taking that wild chance on us and going at insane great lengths to reunite my brother and I to our parents. We had been apart for two years. They had to flee the country and we stayed behind - hoping to see each other again one day.

I remember we didn't speak a word of French and our pretend parents didn't speak a word of Romanian. So for days we would spend time together as we travelled from country to country on our journey, pretending to be a family and simply gesticulating and pointing at things and trying to bond through improvised sign language. We eventually made it to the Belgian refuge camp where our real parents were waiting anxiously and when we made it, none of us could even believe it was real. We didn't last long in that refuge camp though. The more time you spend in a place like, the easier it is to fall into a life of regrets. I was tempted and even taught to steal by my middle aged "camp friends" who lived by stealing left and right. But I never did it. I never stole a thing. But if I had to I sure did have the training for it.

I Could Have Ended Up Hateful, Angry And Resentful

My brother and I ended up going to a school 13km away, in a town called Dinant. When we started out, we didn't speak a word of French so we just tried to adapt as much as we could. It was tough at school because none of the kids in the school wanted to play with us. Except one kid. But when the popular kids reached out to him he dumped my brother and I and chose the cooler crowd. So we spent our days and nights studying. It was either that or getting into some sort of trouble. The only thing that kept us on the radar of the other kids, the only reason they knew we even existed is because I was good at drawing so I entered a competition with a cartoon I had invented and became kinda nerd-famous for it in the school. Other than that, we were simply not the kids you'd wanna hang out. We were simply uncool immigrants with horrible accents and a more mature outlook on life than kids our age.

We were too young to walk to school so we used to tell our teacher that our parents were dropping us off every morning but that was a blatant lie of course. I think she knew it all along but was a big softie and didn't call us out on it and never reported us. We walked a few kilometers every day from Hastiere to Dinant to and from the bus station and took the bus most of the way. In the evenings, back in the camp we used to run a little French class for the immigrants in the camp and it was funny seeing grown men come up to my brother and I and sitting down through a good old lesson of "Je m'appelle Gheorghe et je veux du boulot" - that was the most popular word they all wanted to learn to use well - "boulot". Meaning job. For months, our tiny little room in that camp had become a career development centre and I had invented an innovative way of getting those guys (the camp was primarily housing men) - to speak French. I created a phonetic alphabet, for Romanian immigrants at least, so that they could read entire French phrases using Romanian sounds. It seemed to have worked wonders because we had more customers and we could handle. We would work for free but in turn they would all give back by teaching us whatever they knew and taking care of us if anyone bothered us. In addition to learning ping pong, basketball, and swimming, the camp taught us lessons in street hustling I would never have otherwise been exposed to and in a way I'm glad I went through that

I Could Have Ended Up A Bully

During the day, at school, we hated recess because for 90 minutes we had to walk around the big schoolyard full of hundreds of kids playing and having fun, and my brother and I we used to just talk to each other and think of the things we would do when we'd grow up and we didn't have to put up with snotty kids like the ones that kept making fun of us on a daily basis. Most of our clothes were hand-me-downs we used to get out of pure mercy and sure, some of the shoes didn't have all the laces and stuff but hey, at least we had shoes. They also made fun of our lunches. We used to pack some of the food we used to get in the camp back in Hastiere and bring it over to school for lunch. Of course, the kids in Dinant were not impressed by our refuge camp food. And they had to make sure we knew how they felt.

We kinda beat the crap out of one of them one day. I have to admit it felt good. My brother and I had enough towards the end of the year and when the biggest, baddest bully got aggressive, my brother jumped him and then of course his friends jumped by brother and of course, I jumped the friends. Hey, we won. And the funny thing is that after we beat the crap out of the them - because you can't even imagine the anger two bullied immigrant kids can unleash on five or six first-world bullies - every single one of them wanted to be our friends and we

suddenly become wildly popular. If I had known, I would've punched them right in the mouth the first day we entered that schoolyard and we would've enjoyed a full year of friendship. But hey, who knew.

So for about a week we felt kinda popular - felt good too - but unfortunately it was the end of the school year and we had just found out we were getting deported back to Romania and we saw our newly-made friends - you know, the ex-bullies - one more time at the awards ceremony at the end of the year. Somehow, my brother and I won the best students of the year award that was traditionally given out to one student every year but the school administration decided to award both my brother and I.

But Education Saved My Life

I could've ended up a thief. Like my refuge camp friends or like my Bucharest street thug friend. I could've ended up a street hustler or even a bully. And like so many other friends I had before and after. If I learned something through those years growing up is that most people who slip into that kind of life - most, if not all - never wanted that life in the first place. But they never had the opportunity for a better life. They just kinda struggled and hustled their way into it and once you end up there and have nothing, it's hard to hustle your way out.

When I grew up in communist Romania we literally had nothing. Every single thing - every single toy, every single clothing item, every single chair, every single cup of water, every single piece of bread - everything - everything - had to be fought over. You had to fight to survive. God bless my parents. I can't imagine raising children in such poverty. Friends, if you haven't lived through that - watching a movie or even reading a story like this isn't enough to explain it. If you did have a tough life - then welcome to the club, you know exactly what I'm talking about.

But the purpose of me telling you all this is not to make you understand if you haven't been through it or not to remind you of it if you did go through. The purpose of it is to make you believe this: Education saved my life.

This is it. I'm convinced that it's absolutely the key, the answer, the secret to everything. Education saved my life and I know it saved my brother's life and I know it saved a lot of lives. And so, as I look at my four beautiful children and the beautiful life they have, and as I enjoy every second of my life, I can't help to wonder, what would my life have been without education. What would my parallel life look like? Would I be as happy? Would I have a happy marriage, happy

children? Would I enjoy the things I enjoy now? Would I be able to think, speak, feel, the way I do now?

I'm absolutely certain that the lessons I learned living the childhood I've lived helped me become stronger. But, I am equally certain that without education, I would've ended up in more trouble than I could even imagine. It was all setup for me to fall into it. And yet, I didn't. Thank God for mentors, thank God for people who believe in you when you don't and thank God for education.

I Have A Gift I Want To Share With You

I've spent the first two decades of my career learning and becoming as good as I possibly can at what I do. I brought my hunger, my memories, and most of all, my heart - to the table. Every. Single. Time. Always. And I'm humbled by the opportunities I have received. But now, my life would be meaningless if I would keep all that to myself. I heard someone say something once that made me change my course in life in a heartbeat: Pride is unshared knowledge.

Education saved my life and I've learned my share. Now, it's time to pass along some of that knowledge. The first 40 years of my life have been about learning a lot and I know that learning will be a big part of my life moving forward because I'm more starved to learn now than when I was younger. But, it's time to pass it forward. It's time to give. My commitment to you, is to help you grow in your very own Developer Journey.

This is my Education Story and this is the beginning of the Carmel Story - and the beginning of a new chapter in my life but more importantly - the beginning of a revolution in education, that is all heart. I call it All-Heart-Education and it's what I've vowed to spend the rest of my life doing. Giving back to those who need it most. And giving what matters - education. From the heart. If you feel the way I do about education and if this story is one you'd like to be involved in - simply stay tuned and let me know what you think along the way.

There's a lot of people who are making the Carmel Story the movement that it has become and I am going to introduce them all to the world as this story progresses. I can't wait for you all to meet them. But for now, I want to personally welcome you to our RoadTo1B lives changed through education. That's the dream, and I know that if it can happen to one poor kid from Romania, it could happen to anyone. It can happen to you too.

Let’s do this. Together.

CHAPTER 2

Poor Kids Deserve A Shot Too

"Hey listen up! Whoever ordered their books from me last week, come and get them quickly before class begins", I yelled at the top of my lungs. I had to yell. That lecture hall was full of first year statistics students who were having a hundred simultaneous conversations. I dropped my big box of freshly printed textbooks on one of the desks somewhere in the middle of the room and I made sure everyone not only heard me yelling but that they also heard the loud thud of the big box being intentionally dropped on that wooden desk. All of a sudden, I had the entire room's attention for a few seconds and since I knew the professor was about to walk in any minute, I got right down to it.

"I got your textbooks here, the ones who ordered them, come and get them - you know who you are. I printed a few extra ones in case anyone else is interested. Just so you all know, if you bought your Statistics textbook at the library you can return it within 30 days and get your money back. This right here, is an exact copy of that textbook, except it's all black and white. If you care about pretty colored pictures in your books, then maybe university is not your thing, kids. The good news is that it's only $30 instead of $125. Taxes included, of course. Unfortunately, I only made a few extra copies but if you want one come give me your name and phone number and if you pay in advance you'll be at the top of the list for next week's drop."

This was stupid and illegal, of course. I was literally copying books at a print shop downtown Toronto and I would sell them back to the students on my campus for a sweet profit. Everyone and their mother knew it and they absolutely loved me for it. I used to save them loads of previous cash. I'd walk into my classes and I'd see my colored textbooks on most desks. I mean, you had to be stupid to pay 4-5 times more for a textbook you were only going to use once.

Once A Hustler ...

I was in my second year at the University of Toronto and it was just after we had returned from our first trip back to Romania. I had spent most of my money on that trip and I was pretty angry at the system that literally abused young naive students of their incredibly precious little money. The previous year I had paid $900 for all my textbooks in Year One of my Computer Science undergraduate

program at the University of Toronto. I mean, my first car cost me $1,300 and that was a car I bought together with my brother. And so, imagine what $900 meant to me. And it wasn't like I could ask my parents for that kind of money. God bless them - they were not in a position to pay for my school and I am forever grateful for that. Because I was forced to learn to survive

And oh boy, did I survive those days. Those instincts I had developed growing up in that refuge camp, and in many other similar situations growing up, all of a sudden started flooding my mind and I just started going for it, man. I made cash during the weekends as a waiter and I used to use some of that money to make more money.

I had devised a plan find a local print shop downtown Toronto, where they would copy any book I'd give them at three cents a page and would reprint subsequent copies at a cent per page. And it worked. The highest margin textbooks were the ones starting at a couple of hundred pages and up, because they usually went for over a hundred bucks at the library and it would cost me about twenty bucks to make them. Cover and binding included. I not only printed all my textbooks for Years Two, Three and Four - but I also printed them for all my friends and for their friends and for everyone else in school who had a brain. I used to have it all color coded and I would call my awesome old asian lady who ran the print shop and all I had to say was "30 more of the red ones please and 25 blue ones. I'll pick them up tomorrow."

I felt like a secret spy giving coded messages over my fancy Nokia 3310. If you haven't held a Nokia 3310 in your teenage hands during the 90s, you haven't lived, my privileged millennial friends. Let's say the word "snake" for you means - well - who knows what it means. For us 90s kids, to this day, "snake" means a brand new world of opportunities. Meaning not that I ended up a snake oil salesmen, although that would've been an option. No, meaning that technology showed me a world beyond my wildest dreams. It's like it opened up my mind to a million "what ifs". More on this a bit later on in this story.

Raging Against The Educational System

I'm not saying what I did was pretty or would I ever advocate for anyone to do anything of the sort. But I tell you this. It's absolutely criminal for schools to charge so much for educating young minds. I desperately wanted to learn everything I could about technology. I understood education would keep me sane but I knew that technology was the stuff my future was to be made of. No, I understood that technology was the stuff everyone's future was to be made of.

And I wanted to rock at it. But it was expensive. And boring. But mostly expensive. And boring.

Schools used to be about education - but somewhere during the 20th century, they mutated into Big Business. Nothing wrong with business. Except when the so-called-customers are poor immigrant kids who just want the opportunity to learn valuable skills that would enable them to add value to society and feed their families one day. I mean, $900 for a bunch of books - some of which were not even used? Yeah, you know what I'm talking about if you went through that. You were supposed to pay another $15 or $20 for the professor's own little personal unpublished textbook they scrapped together and they would use that to teach the course. Oh yeah. Hey, at least my textbooks had better covers than theirs.

First year statistics was amazing. Everyone hated that course but the ones that had to take it - well, had to take it. I didn't. I was in the Software Engineering stream. But that didn't stop me from pretending that I did. And it didn't stop me from lying about the fact that I had previous orders on those text books. That's right. When I dropped that big box on that wooden desk and got the entire room's attention, I knew perfectly well no one knew who I was and no one had ever spoken to me before, let alone order illegal textbooks from me. But they didn't know that. And they fell for it. I made a killing. Selling every single book in a matter of minutes. They were begging me to take their $30 on the spot just to make sure they get a book the following week. "Alright, alright. I guess I can print another batch. Oh, what I wouldn't do for you guys."

The Educational System Is Setup To Favor The Privileged

Lying, copying books - which is effectively called stealing, and selling on school property. I did worse things than those. But let's stop right there for now. You get the idea. I never did become a street hustler but it took me years to civilize that hustling beast that was always buried deep inside of me. I felt I could sell anything to anyone, anytime, anywhere. I even sold my "coding services" to other students. I hated school but my code was pretty good. Ok, it was usually so sick that I wanted to simply hug it. My comments were tight, the structure was simply beautiful, it would read like a book. I don't remember not getting an A on a coding assignment. Well, I do remember one time. That's a funny story. I might tell you all about it another time. But I'd charge about $100 per assignment and I always guaranteed an A. And I always got it for them.

I didn't end up hustling on the streets of Bucharest in the 80s, nor on the streets of Belgium in the early 90s - but I couldn't help not letting it all out on that U of T campus in the late 90s. Look, I'm not proud of it. In fact, I spent the rest of my

career defending artists rights, left and right. To this day, I don't ever rip movies off the internet. I would rather rent them online or straight out buy them because I know the creative process behind anything is excruciating hard work and creators deserve to get paid. But back then, I was still a raw kid from Bucharest trying to survive in a first world country and there was no manual for that sort of thing.

That was two decades ago. Since then, I've heard stories of parents going into massive debt just to be able to provide a so-called-good-education for their children. And it's usually those families that come from an unprivileged background. See, if you're rich, you can afford stuff and you can afford textbooks that you'll never open. And you can afford to fail courses several times because - hey, daddy will pay for me to take that course as many times as I need to, before I pass. But if you're not one of those privileged kids who get a brand new car when they graduate grade eleven in high school - true story at my high school in Toronto and at many high schools in first-world countries I'm sure - then you have to hustle your way through education. You just gotta. It's just not setup for you. All my parents could afford as a high school graduation gift was chocolate bar. True story. I loved it and treasured it. Because it was a symbol of so much more.

Everyone Deserves An Equal Chance

It's almost as someone somewhere sat on a big fancy victorian high chair and said "Yeah, Dan from Romania - nope, not him. The other kid, you know that overweight one from that country, yeah - not him either. Oh - John Fitzgerald Smith The Third - yes sir, step right through - here's a life discount for the rest of your life - to everything." I hear that if you're white there's some sort of privilege cow somewhere - I dunno where - but I hear that you can just milk that cow and that beautiful white milk is the stuff easy dreams are made of. Well, I'm still looking for that cow and no - actually not anymore. If any of you find it, let me know, maybe I can get me a fresh jug of that white foamy milk.

No siree, no white foamy milk for this kid from Bucharest. I had to work my Eastern European butt off ever since I can remember. But listen. I want to make sure we're on the same page. I am not belittling the centuries of oppression against so many classes of people out there. But there's one class, called poor Romanian kids, that I guess I'm advocating for and there's nothing anyone anywhere can say or argue about - Romania is the poorest country in Europe and yes it's made of mostly white, poor people. Poor being the operative word. Here's what I'm saying, my friends. I'm saying that I believe in privilege - yes. But I think that the privilege we need to be pointing out is money, above all other kinds of

privileges. If you're poor, the world is setup against you and you have to work so much harder than those who are not poor and oh - so much harder than the ones who are not only not poor, but they are filthy rich. Be them white, black, blue or short or overweight.

And I think we all deserve the same chance. Don't get me wrong. I don't advocate for a world of equal outcomes. No, I don't believe in equality of outcome as much as I don't believe in communism - where that kind of mentality was forced upon us, upon our parents and upon generations of poor families who didn't stand a chance. No, I don't believe Universal Basic Income will solve our problems. That's an equality-of-outcome solution and those end up in someone taking over the whole country with a hammer in their right hand, a sickle in the other. Millennials - that's a reference to communism. I'm not picking on you. I love you. My wife's a millennial. I know, I'm such a lucky Gen X-er.

But I digress. Listen, there's nothing wrong with having money, but if you didn't earn it - I don't respect that. Who am I to disrespect your unearned money? I'm a nobody. But trust you me - no one respects someone who hasn't worked hard for their money. We're just afraid to tell you that most of the time. I'm not afraid anymore. I'm gonna say it and I don't care if you get offended. You need to work. Work hard. And everyone should get a chance at working hard for their money. That's what I call equality of opportunity. Everyone should get their fair shot at working hard to make something of themselves in this life and it all starts with a fair shot at education.

The Future Of Education Will Be Open To Everyone

The way I envision the future of education is radically different than the present and the past. I see a world where we don't speak of color or race or any of that stuff, but rather of equality of opportunity and of pure competence and hard work. This is what the Carmel Dream is all about. Building an Educational System that's completely out of anyone's reach in terms of controlling it and owning it and dominating it. A system that can't be hacked, tricked or rigged. An incorruptible system. Where poor kids from Romania and poor kids from Africa and hey, poor kids from America and France, and Italy and - I don't know about the Canadians - ok fine - we'll let them in too. Definitely not the Swiss. There's no poor kids there. Alright, fine, we'll let the Swiss in too. A system that welcomes anyone who wants a fair shot at Quality Education in a technological world that's so far away removed from that Nokia 3310 I was sporting during my campus hustling days.

This is what Carmel stands for. This is why I want to spend the rest of my life dedicated to it. It's more than a product or a business. It's almost as if we have have the opportunity - through Blockchain Technology - to create something that resembles more of a country than a startup. A digital community let's call it - where anyone hungry enough to make it - will make it. And where no one can pull out their rich privilege card because we won't have those kind of card readers built into Carmel. But I'll tell you what kind of readers we will build into it. And they are already in. Mind readers. Settle down, that's just a stupid joke. Maybe in a few years. But for now, Carmel is built on Heart Readers.

I absolutely don't care how cheesy that sounds. I'm after people who go all in - heart first. Because it's those people that we all want to lead this world of ours and it is those people that will go all heart and no fluff. They should be the ones that get the opportunity to become true leaders. Imagine what kind of world this would've been if the ones who deserved to lead us where actually the ones who did lead us - but somewhere along the way maybe they were too poor and too tired to hustle their way through an educational system setup for the privileged. And maybe they simply gave up and ended up anonymous - just living - just surviving - somewhere.

Carmel is for you my anonymous, unprivileged future leaders. You - the big-hearted ones - you the crazy hard working ones - you the ones who deserve it the most. You - all the Chris' out there. I invite all of those of you who believe this, to join us and to build Carmel into a universe that is 100% ours and ours to shape, form and mold. Come and see what we've been working on and get involved. Make it yours. Make Carmel yours. Because it already is. It's time everyone got the quality education they deserve and it's time one of you invent the next Nokia 3310 of the 21st century.

Or at least learn to code for it.

CHAPTER 3

Education For The Sake Of Education Is Not Enough

I had never seen more poverty in my entire life. And I didn't grow up with warm water, let me tell you. But this was different. A few hundred people, close to the city garbage dump, crammed together all around that dump, in improvised carton boxes. You can't honestly call them houses, but they call them homes nonetheless. My heart was absolutely torn apart seeing young children playing in the ankle-deep mud and holding a piece of dirty bread in their unwashed hands, while smiling - yes - smiling - and playing and running around in that nasty mud, without the slightest sense of worry on their beautiful little minds. I was accompanying a friend who worked with the children and tried to connect with them through music and games. The entire experience is something I am not even going to attempt to lay down in writing because there are no words to describe what I felt. One little story stands out above the rest though.

I met a young man, a fifteen year old who looked and sounded more mature than many adults I know. He had recently become a father and he was exceedingly proud of being one. My mind and heart were all over the place when I approached this young - old - man and started talking to him. I felt as if I wanted to help him with all my might but at the same time, there was an unconscious desire that kept tugging at my heart, urging me to ask him questions and somehow, someway, learn from him. So I uttered, perhaps the dumbest question I could muster under the circumstances - but I said: If you could have anything in the world, anything at all, what would that be?

Poor Education Kills Dreams And Breeds Poverty

I don't know why I asked him that. I just thought that if you wanna help someone, you might as well ask them how they might envision that help and kinda start from there. His answer is still striking a continuous deep chord in the depths of my soul, even to this day, as I recount this - and

every single time I remember that encounter. It's something you don't want to forget. Maybe because it's unforgettable. Here I was, standing before this incredibly unprivileged young man, who had a family to support, who was almost young enough to be my son, surrounded by a mountain of garbage and disease and foul smells, cradling his tiny, innocent baby, and here's what he said: I would want a horse and a carriage so I could haul out more garbage.

The reason why they settled there is because they go through the garbage dump several times a week and they have a first-come-first-grab kind of system of sorting through the garbage and finding items worth selling at nearby recycling centers. The garbage truck pulls in and the strongest and most agile get their first pick of fresh garbage. But the toughest part of it all is carrying the garbage back. And the wealthiest garbage collectors of them all in that garbage dump, are the ones that have a horse and carriage at their disposal, so they can collect as much as they want.

My young friend's deepest dream in life, was to be the richest garbage collector of that settlement. I still find myself either weeping and frowning and clinching my fists in anger at the thought of it all. See, as a poor unprivileged person, you can't even dream of the good life. The misery of your situation has robbed you - most of all - of your dignity and that loss of dignity is no where as visible as in one's inability to dream great dreams. I left that place with a heavy heart and to this day, some of us are still looking for ways to solve that particular crisis.

We Need A New Kind Of Educational Model

Romania is the poorest country in Europe. Statistically, a quarter of Romanians live in poverty and that includes half of all Romanian children. And poverty is relative. You see, not only is Romania the poorest, but here's the really sad statistic. The at-risk-of-poverty threshold is calculated based on the median income and Romania is the second last in terms of the median income. So that means the average Romanian earnings puts Romanians second from last in terms of income and in addition to that, a quarter of Romanians are well below that income. This, after half a century of communist oppression that left the entire nation absolutely crippled for generations to come - in more ways than one.

Why does all this even matter? Here's why this matters and why it continues to boggle my mind. Because when it comes to serious international educational competitions, Romanian youth consistently rank high. Podium high. Young Romanians from all over the country keep on bringing back home coveted medals in all sorts of key subjects, especially Math and Computer Science. Go find an international Math competition and you'll be sure to find a Romanian presence near the top ranking students. I'm not even going to mention the fact that Romania is killing it in terms of the growth of the IT industry. The talent is simply phenomenal.

I don't mean that everyone is going to be a top Math Olympian. No, but here's what I'm saying. Those Math prodigies and those Computer Science genius kids, should be given the opportunity to build sustainable economical structures that create enduring positive outcomes for others. Like for my fifteen old dad raising his new family in that garbage dump. And that's where education breaks down. When education exists for the sake of education, we all pay for it one way or another. If education's sole purpose is winning medals and obtaining diplomas and paying through the roof for degrees, we're all in trouble my friends. Deep trouble. It can't be just about that. There's gotta be more to this story. What is education all about?

True Education Must Lead To Economic Growth

Education was never meant to lead to pure academic achievements. Education was all about preparing young minds to become a valuable part of the economy. To explore your gifts and unique abilities to contribute to existing value markets or even create new markets. Education removed from economy is not only useless but down right dangerous. It turns education into a financial black hole that burdens the students and their families with disproportionately massive debt. And the worse part is that when education becomes a business, obsessed about the bottom line - profits first and students second - or third, or fourth - that means the entire curriculum is not designed with the best interests of students in mind. But rather, it's optimized to increase shareholder ROI. The fruit of that misalignment of interest is truly and clearly visible when those students enter the workforce. They're completely unprepared for the real world. Sure, I'm generalizing, but this is becoming a worldwide epidemic.

Romania is the poorest country in Europe but the effects of a sterile educational system are well observed throughout Europe, North America and every continent of this great world of ours. If we are ever to hope for a less impoverished world - globally, we have to redesign our global educational system. And the main flaw is its misalignment with real economical needs. If we are to get to the bottom of it all, we need to empower new generations of hungry minds with an education that can not only put food on the table and ensure basic material needs but an education that can contribute true value to existing markets and even create brand new markets. If we can put our minds together and get around to building the right educational structures for this to happen, we might just be able to be alright after all. And even the poorest of the poor might find themselves in the midst of networks of productivity, enough to pull them out of their current poverty and misery and propel them towards living authentically dignified lives. For themselves and their loved ones.

As much as we might want to believe that going to school and getting a high school diploma might help that fifteen year old get out of that slum, it's nothing but a pipe dream. It won't work. We live in a world right now where we simply don't care about diplomas anymore as a society - not as much as we did a few decades ago anyways. What we need in the 21st century, are people skilled enough to get the job done, diploma or not. And as much as those Math prodigies might help the national brand get ahead in the world, a medal is not going to solve real customer problems and a diploma is not going to cut it out in the real world. As much as I believe education can save lives and yes - I do believe everyone deserves their shot at a good education - I don't believe that's the end of the story. No, education for the sake of education is not enough. We need to give people a chance at making a living and we need the sort of education that gets them there.

What we need to do is measure the effectiveness of education based on how many real jobs it creates and how many lives it truly saves in the process. We're beyond measuring the efficiency of the educational system. What good is efficiency if it's ineffective? And when I mean jobs, I don't mean academic jobs - as much as we need those, the bulk of the economy requires more than professors. So what does our economy really need in

these difficult times of ours? What do we desperately need as far as skills, valuable, marketable, in-demand skills?

Every Job Will Be A Tech Job

Well, before we answer that question, it's well worth stepping back a bit and reminding ourselves about where it is that we find ourselves at this particular point in human history. We are at the very cusp of a new era. One where non-human entities are poised to yield so much economical power that humanity will be forced into an existential fight or flight mode - unlike anything else we've ever seen in the history of our species. Think for a moment, that we now have more intelligent machines than we ever had before and that these machines can do things that humans cannot even dream of attempting. You know how you could never compete at arithmetic with a calculator? Well, multiply that by a trillion and we're not even close to what's happening today.

Analysts predict that roughly a third of US jobs are at risk in the next fifteen years, because of these incredible advances in machine intelligence. The UK is approximately in the same boat, as well as most developed countries. Developing economies are even more in trouble. So in addition to fighting poverty, the next decade and beyond, will see us fighting machines as well. At least competing with them for our jobs. We're not talking factory jobs only - but every job. Every profession is at risk, including lawyers, accountants, sales people, cooks, even surgeons. If we are to make a future for ourselves, we need the sort of education that truly is job-creating and is effective at doing so.

If every profession is at risk because of intelligent machines, then perhaps the last profession on that list would have to be the profession that got us in here in the first place - Software Development. As the Digital Era unfolds before our eyes and eats up every industry known to man, digital skills are going to be more and more in demand. Put it this way, someone has to write the software that runs those machines and someone has to maintain them and communicate with them. That's where Software Development comes in. It's clear that non-technical jobs are disappearing but what's also crystal clear is that tech jobs are booming.

We Need To And We Should Do Better

In this Digital Economy, if we are to design an education system that stand apart from the crowded sterile educational landscape, we'd have to build one that's primarily focused on the creation of tech jobs. Jobs that are future-proof and jobs that are in high demand. For us to get there, we need to teach technical skills that are relevant today and that will still be relevant as we all grow through this transitional period towards a dominant machine-driven economy. As scary as that sounds, we might as well embrace the reality as fast as we can and get onboard this train before it's too late and we're left behind scratching our heads.

If we are to hope at a human-oriented future, where personal dignity is flourishing and where young fifteen year olds dream good dreams, our global course of action is plain and clear. We have to arm current and future generations with relevant technical skills and send them out in the marketplace, ready to not only survive the digital economy, but face it fearlessly and courageously - ready to thrive. We need to make Tech Education our priority and we must move fast, while we still have time to learn and acquire the right skills.

The challenge is that current Tech Education options are limited to say the least and at best, they are highly unscalable to the massive number of people that will flock to such solutions once they begin to face the reality of the new machine revolution that's upon us. Tech Education has grown from a theory-first educational system and it's not helping us achieve the job-creation goals we need to achieve if we are to transition the global workforce to new employment opportunities. Attending expensive lectures or watching video courses online is not going to cut it. We need to do better if we are to stand a chance at providing a future for ourselves and for our families. We need to improve Tech Education.

And we need to do it now.

CHAPTER 4

Is Software Development For Everyone?

Hugo and his pet dog Uno, were two of my best friends during my teenage years. They were two cartoon characters I had dreamed up during classes I couldn't stand, mostly English classes, in grades 10 and 11. Hugo was indifferent, always laid back and chill, and he couldn't care less about anything and anyone. Including Uno. He could never bring himself to show his poor dog any love. Well, Uno on the other hand, was this exuberant, always happy, always high energy kinda dog that just always got Hugo off his game and eventually always got him to play or do something fun.

I'm pretty sure Hugo was inspired by Meursault, the protagonist in Albert Camus' "The Stranger". A book I devoured back then and a book that I spent years trying to unpack and understand. I marveled at Camus' ability to paint a vivid picture of apathy, a feeling well to known to teenagers since time immemorial. Especially teenagers with a tendency towards absurdism and existentialism. Looking back, I'd most likely not recommend Camus to my younger self but in the absence of a working time-machine, other than good old memory lane, I'll just have to live with what that younger self turned out in the end.

In between drawing, reading philosophy and writing poetry, my life was nothing short of an amalgam of poking at life and trying to express in words or pictures or rhymes, what life was saying back to me. Needless to say, my parents and my high school guidance counsellor were equally confused about what to do with a young artist with an old soul who'd spend hours writing prose about the way a window drape moved in the wind.

"How will you feed your family with your drawings and your poetry?"

"You can't be a dreamer all your life, young man."

"How about you learn something that's useful to society? Something that makes money, you know."

I ignored it all of course, the parents, the counsellors, the well-meaning adults in my life who laughed at my jokes, cried at my poems and sighed and nodded their heads approvingly at my cartoons. Still, they insisted I picked up a real career, and they made sure I heard them that they rather see me hold down a good job, raise

a good family and earn a living doing something "real" with my time - not just make people laugh, cry or sigh. I ignored them all.

Traditional Tech Education Is Fundamentally Broken

Until my best friend, my poetry buddy and philosophy maniac who read more than I did, and daydreamed more than I drew, said to me one day: "Hey, how about we enroll in that Computer Programming course next term?"

"What?", I said. "You too?"

"No man, it will be interesting. Let's see what this computer stuff is all about. Why not?"

Why not, says he. Well, that's an intriguing way of putting it. And that's how it all began. Back in grade 10, in 1995, the year of Windows 95, floppy disks and the Pentium Pro. How hard could computer programming be? Right? Well, little did I know. Let's just say I had never gotten a lower grade in my entire life up until that point. In fact, I had pretty good grades.

That course - Grade 10 Intro to Computer Programming, I almost flunked it. I barely got a passing grade. That's how my lifelong journey into Computer Programming actually started. By almost flunking my first programming course. What I experienced during those few months that seemed like never ending years to my fresh 16-year old mind, would change my life forever and it would fundamentally shape my relationship to technology, and especially to Tech Education.

To create great software products, you need much more than a Computer Science degree. The science of computing is absolutely necessary as it is foundational, but it simply isn't enough. To be a great Software Developer you need to be much more than a computer scientist. You need to be an artist and you need to be a philosopher. That is because Software Development is part philosophy, part art and part science. This simple understanding of the nature of Software Development opens you up to a whole new level as a Software Developer.

What I experienced in that Grade 10 course, was a typical scientific approach to software development. An approach devoid of art. And devoid of philosophy. The young 16 year old philosopher-artist me, was immediately turned off by the software-development-as-science method of teaching and my mind - and heart - completely tuned out.

The Science-First Approach Turns Most People Away

"Can someone tell me how many dimensions an array can have?", the teacher asked during one of the soul-sucking, never-ending, dry lectures that drove me nuts to no end. How many dimensions? Uhm, how many more than 3 can you have? Well, 4 if you count time and that's still a stretch. What kind of question is that, I remember thinking. Oh, and this "array" thing, what does that mean? I know she mentioned that before, right after words like "variables" and "constants" and I honestly blanked out and went straight into drawing a fresh new episode of my beloved Hugo and Uno, while deep diving into a whirlwind of Meursault-esque teenage apathy. Thanks Camus. Thanks.

"An infinite amount", replied the class genius. This was Toronto but somehow the dumbest guy in class - me - and the smartest one - the one who answered, we were both Romanians. Go figure. I guess we're everywhere, eh? That's maybe for another story - another time. As soon as the teacher congratulated him - once again - for being such a smart, awesome nerd, I just literally lost it - silently of course.

My mind was racing like a hungry stallion on steroids chased by an infinite army of array-shaped monsters. Yeah - don't get me started with what my imagination can concoct when utterly confused by a concept that my mind fails to grasp. "Infinite dimensions? What is he talking about now?", I whispered to my equally confused philosopher buddy who was just taking it all in with a ridiculous smirk as if he enjoyed flunking the course. Arrays. Infinite ones too. Variables. Constants. If-statements. While loops. Coordinates. Functions. Algorithms. Arrrgh. Stop!

I can just imagine Meursault lighting a new cigarette and blowing smoke in the teacher's face as her eyes pop out with rage, red cheeks, crisp mouth, yelling "Get out of my class!" Alas, 16-year old me was not as assertive as Camus' apathetic hero - or actually, was he assertive, after all? Oh well. I sure wasn't. And so I strolled through that confusion as if nothing could touch me and I nonchalantly survived weeks and weeks of mental torture until that dreadful day when I got my final mark and I swore - I swore - I'll never touch computer programming with a ten foot pole as long as I live.

Software Development Is A Lifelong Story

That promise to myself lasted about a month. I just couldn't figure something out. And shortly after that course ended, during the '95 winter holidays and well into the new year, I tasked myself with getting to the bottom of it. A crime had been committed. Not against another human being, as in a flat Sherlock Homes novel, but a graver crime, a full blown crime against humanity and against the

noble institution we collectively call "Education". I had been through a programming course and not only could I not write a line of code but I hated it with an irrational passion. I certainly did not feel "educated". And that bugged me. The nagging feeling wouldn't stop.

As a philosophy enthusiast, I spent a few months analyzing my utter failure in that computer programming class. I kept thinking of some of those few happy souls in the class who had so much fun programming and I couldn't understand how someone can actually have fun with variables, arrays and even worse, n-dimensional arrays.

I remembered some guy in class who was so way ahead of everyone else that he would actually code small computer games in between classes. Once in a while he would give the rest of us mere mortals a sneak preview of what he was currently working on. I once saw him demonstrating a shooting game where you had a cowboy character and you had to shoot some stuff up and you would even get points and stuff. That really intrigued me.

I just couldn't make the connection between the theory and the practice of it all. Right before that '95 winter holiday, I put myself out of my misery and I went and found that smart guy from my computer class. I asked him to give me a copy of the programming language we were supposed to learn in class. That language was Turing. I took the floppy disk home and plugged it into my very old 386 PC (those were the days) and loaded Turing up.

Science Is Not Even Half Of The Story

Floppy disk. Programming Language on a floppy disk. Wow. Let me say that again. This wasn't just before Facebook. Mark Zuckerberg was 11. Enough said. No, this was before Google. Can you imagine a world before Google? Now, this was before Java was cool and before the Internet was widely available. Meaning, I couldn't just go online and read tutorials or anything like that. There was no "online" back then. That means, I just went and tried different things to see what would happen.

Right about that time I had a passion for Tetris so I set for myself this insane goal of writing a Tetris-like computer game and I told myself I won't give up until I did it. Guess what. I did it. It took me a few long weeks of trial and error but I eventually did it. It was around Christmas time. I remember it still. I ran over to my parents and told them I wrote my first computer game. Me, the heads-up-in-the-cloud-artist-philosopher teenager. I suddenly had become a programmer — and no one forced me into it. I even enjoyed it.

I was ecstatic. I felt alive. I felt on top of the world. You know that feeling when you do something insanely awesome and it's something you thought you could never - ever - ever do? A few months before, I had almost flunked the class that was teaching us basic stuff, and now I had written an entire game, including levels and a scoring system, complete with graphics (all-be-it very primitive graphics) and all.

The lesson to be learned in all of this is that computer programming is more than mere science. But how did that happen? What happened? How did I become a programmer all of a sudden, when just a few months before I was convinced programming was definitely not for me?

Sure, I did end up studying Computer Science and I did earn a Bachelor of Science but Software Development is so much more than science. The stuff that explains how computers works - the science - is crucial and fundamental. I eventually had to learn how to use variables and arrays and all that technical stuff. That was all necessary in order for me to create my first ever functional game. Absolutely necessary. But it's not all that was required for me to create that game. Knowing the nitty gritty of programming - the science - that's not the whole story. In fact, it's not even half the story. At best, it's a third of the story.

Traditional Tech Education leaves out the other two thirds. What they didn't try teaching me in that class was the other two parts — the really important ones. They started with the "how" of Software Development. That's the go-to pedagogical approach in Tech Education, whether in world-class Computer Science university programs, online coding courses or programming bootcamps. It's all about getting aspiring Software Developers to chew on variables and constants and arrays until they choke the life out what Software Development is all about.

The Philosophy of Software Development

Software Development is all about using technology to serve humanity. Period. It's about getting machines to do things that make our lives better. And we do that by create useful, valuable, beautiful Software Products that human beings enjoy - and sometimes even love to use. We need to start with that when we teach technology.

We need to start with the "why" of software development. Effective Tech Education has to begin with The Philosophy of Software Development. With the unpacking of the reasons why anyone should even care to sit in front of a monitor and spend hours thinking and typing and fixing bugs and creating new value out

of (almost) nothing. If we do that well - if we give newcomers a reason to learn to code - then they will.

Because once the philosophy of it all sinks in and you understand the immense value of possessing the ability to boss inert machines around and get them to do things that add value to the Human Experience - well that - at least to me - is mind blowing. If that teacher would've sat me down and said, young man, let me tell you why going through this course is going to change your life and your outlook on humanity - I would've thrown Hugo and Uno (maybe not Uno) right under the bus and I would've perked up my ears to make sure I absorb every next work that would come out of that woman's mouth.

If she would've told me that a computer is an extension of the Human Mind - I mean c'mon. Even now, 23 years later, I still get goosebumps every time I think of a computer as an extension of the human mind. And here's the absolutely outta this world part - if the computer is an extension of the human mind - what's programming then? It's the key to telling the computer what to do, it's a window into the computer's mind so to speak. Meaning, if one is to master computer programming - in some mysterious way - they could also - even if partially, peek into the depths of the human mind. Honestly, does anyone need more than this to at least flip the page and say ok, I'm intrigued - tell me more?

Oh, how I wish we taught young hungry minds - and maybe not so young but still hungry - that computer programming is a deeply philosophical subject matter and that it hides layers of beautiful and meaningful mysterious truths about human nature itself, if only we had the proper guidance and patience to dig deep enough. Philosophy first. It just makes sense.

The Art of Software Development

Next - before getting to the science - right after the philosophy and the unpacking of the "why", I would've loved to understand what it is that I'm supposed to build, before I actually get around to building it. The creative component that makes computer programming so appealing to me and many others. If only they would've told me that coding is an art.

An art that would allow me to create amazing things. An art that would allow me to express my imagination. Then that would've been incredibly appealing to me. I would've learned the science behind it — gladly — in order to practice the art. In order to create. If you give me a computer today, I see an infinite blank canvas, waiting for me to fill it with the expressions of my imagination.

Back then I saw nothing but a lifeless machine incapable of understanding me and unwavering in its lack of empathy at my frustration in trying to communicate with it. What if I would've been shown, step by step, to see that machine as a beautiful instrument of immense creative potential, much like my pencil - oh, much more than my pencil. If someone would've opened my eyes to that simple notion of using a computer as a tool for unleashing my creativity, that would've opened up an entire creative universe before my eyes.

That's precisely what I did when I pushed myself to go home and try again, but with building a game in mind. Having that creative goal as a motivating and guiding light, fired me up and gave me a reason to go on whenever I felt like the syntax of the programming language was just ridiculous. But I put up with it because I knew it was simply a means to a higher goal.

And so, looking at the Art of Software Development, understanding the "what" of the process, seeing - even if slightly - the object of your creation in the back of your mind, before starting to write a single line of code - that is powerful. Because it takes no technical skill whatsoever and it doesn't take a math genius to fire up the brain's imagination power. Imagine it first and build it later.

If only we would teach those hungry minds eager to become Software Developers, this simple lesson - imagine it first. If philosophy opens up your mind to understand, then art opens up your heart to imagine and peek into the future - at least the potential future. It gets you fired up to act in the present in such a way as to somehow make that image more and more real with every new line of code you write.

Towards A Comprehensive Tech Education Approach

Let's change Tech Education by teaching the Philosophy of Software Development first. Then let's take it a step further and teach the Art of Software Development. And finally, when our students understand the "why", and when they see the "what" - then, and I will add, only then - will they be ready, even more than ready - yes, willing and hungry, to learn the "how" of making it all come to life.

Philosophy. Art. And then Science. This has worked in so many areas of life. This has worked wonders in my professional career as a Software Developer. And I'm convinced this will work in your life and in your future career as a Software Developer.

But first, some context.

CHAPTER 5

The Twinkling Of A Dream

"Should I take an online course and learn Java? Maybe I could build a game." Why is it that computers invariably lead to gaming? Maybe it's because a game literally tells a story and invites you in to "play" a part in that story. Nonetheless, my friend Manu, was debating whether he should get into coding and he reached out to me, asking for advice on how to get started. The year was 2014 and it was right about when I was deep into running an Innovation Lab in Cluj, Romania, with a small but fiery group of developers who where crazy enough to have embarked with me on a journey to launch several new Software Products in a short span of 18 months. We did and it was a thrill.

But it ate up most of working time and I had no time whatsoever to think about coding advice to my old friend. Who, by the way, lived in a different city – Timisoara. So I simply said "Sure, but Java?" Helpful. I know.

The Good Old Days

I remembered the first time I encountered the beast. Sorry, Java. I was still in high school and it was a couple of years after my first stint with programming and that old computer programming language, called Turing. The next couple of years after I had written that Tetris-like game - see it's all about the games I'm telling you - I pursued programming and took a second course, this time, learning another programming language, called Pascal. I loved it, because of course, Pascal was a brilliant philosopher so my masterplan was to explore Pascal because it must be a brilliant programming language. Brilliant. Well it actually was, and that Pascal course in high school was the course that literally sealed my fate as a future programmer.

After I saw the immense creative potential of programming I decided to get serious about it. In my final year of high school, I applied to the University of Toronto's Computer Science undergraduate program and I started reading everything I could about how to be a good programmer. Oh, so this was 1997 and this was really the first year I got online. Everything was so new. What a different world. I started surfing. Oh yeah. "Surfing the net" back in '97 would be like us playing with a time machine today. It was incredible. There wasn't much

out there. As of December 2018, there were 1.94 billion websites according to "Internet Live Stats". But 21 years ago, in 1997, there were just 1.1 million websites.

A million websites seemed like the whole world back then. Especially since four years prior to that, there only 130 websites. Can you believe it – there were 130 websites in 1993. Crazy. So I would log into my dial up Internet after school and I'd start exploring this new wonderful world called the Internet. I read everything I could get my eyes on. This was about the time when I was a book fanatic and I'd run to the public library to check out books and read – and read – and read. Now, I felt like I had the whole world at my fingertips, all the libraries put together, on one screen. And I started reading. And learning. And never stopped since.

When Java Was Cool, Fun And Exciting

One day, my CD tray opened by itself. I - of course, freaked out. And I thought my computer had gone mad. But it turned out I had been hacked. That's how I got introduced to a world of so called "script kiddies" - youngsters, some younger than myself back then – who would find "scripts" - pieces of code written by someone else – and they would use them to hack into people's computers. Why? Oh, just for fun, that's why.

I stayed up nights and kept digging into it and found out that others experienced the same thing – the beauty of online communities – and they posted details about how the hack worked. My mind exploded. Wow. Really? This stuff is possible? I mean, you can login to your computer and run some "scripts" and stuff and click a few buttons and you can open someone's CD tray across the ocean? What? Seriously?

Oh, I had to dig deeper. And that's how I encountered Java. It turned out that this was a new programming language, freshly invented by some genius programmers back then, and it was a programming language that was free to use and anyone could try it out. You know what happened next, of course. I tried it out. There were online tutorials on how to get started and even more advanced tutorials on how to do stuff that to this day gets me so excited that I could drop everything and write me some scripts just for fun. I learned to write scripts using Java, scripts – as in small programs that do one small thing – and I've learned to transfer data between computers using the Internet. I was fascinated.

See, those "script kiddies" who hacked my computer were not programmers. They were kids who got a kick out of hacking people and bragging to their friends

about it. What I was after was to learn what the people who created the original scripts knew - I was after the skills needed to build the original programs that the script kiddies would use. Because those scripts where written by real programmers, and usually for good reasons. But of course the malicious users found ways to utilize them for horrible purposes.

So I became a so called "white hat hacker" - one of the good guys. I started using Java to write "good scripts" that would do good things like protect your computer or alert you when someone is trying to hack your computer or scripts that would discover security flaws and showed you how to make your computer more secure. Oh, it was fun.

The Traditional Educational System Strikes Again

I loved Java. Back then. Then I finally got into University of Toronto and began my formal education as a Software Developer. Guess what the first course - of the first semester - of the first year of that formal education was? Java, of course. In fact, when I started, in '98, that was the first year that Java was introduced as the main programming language for beginners in U of T. They used Turing before. The irony.

And so I thought, oh, I got this. I had been learning Java for the past year, hacking – white hat hacking – and doing pretty advanced stuff. This will be fun and easy. But no. The good old traditional educational system managed to ruin it for me once again. And so we started out with objects, classes and interfaces. Yeah, we started with those. I still remember it. What's a class? What's an object? And what's an interface? Oh, and inheritance? You're in university now, you have to know what inheritance means. Yeah, types – types too. You have to define the types. And on, and on, and on.

When do we get to build cool stuff? When do we get to actually use Java? Not just learn about Java. That's when I got the ominous feeling that I won't be going to too many lectures for the next few years. And I was right. I skipped lectures like they were the plague. I literally lived in the Computer Lab. The ultimate computer nerd sanctuary, open dawn to dusk and I was the first one to get in – in fact I'd wait at the door for the lab to open – and the last one to get out - "wait, wait, almost done, one more line of code – wait". That was me pretty much every day for years in a row.

The good news was that most of the marks were based on programming assignments. You'd have to go to lectures where they would supposedly teach you what to do – same with the tutorials – and then you'd have to go home or in

the computer lab and actually do the work – write the code. I'd go online and learn what I needed to learn and I'd spend my entire day locked in the lab trying different things and mastering my craft. I honed, and honed, and honed. I refused to let the educational system suck the life out of my passion for programming and I learned to love my craft by practicing. By creating. And learning all along. Not by talking about it and about the theory – with no real feeling of the nitty gritty of actually doing it and applying the knowledge to real problems.

I learned the theory while practicing. I could tell you everything you wanted to know about classes, objects, interfaces, inheritance, types and even garbage collection. My favorite theoretical concepts were the ones that professors would not even teach in class but I'd learn by using them hands on and by digging deep, poking and seeing how they work. I ended up doing a lot of Java advanced work after I graduated and started my professional career as a Software Developer. I used Java for a number of years developing desktop, mobile and web apps in the early 2000's and I know that what I have applied in my work were the lessons I learned in that computer lab on that U of T campus in the late 90's. Not the theory I learned during the lectures or tutorials. Not that.

Java Just Isn't What It Once Was

So when my friend Manu asked me in 2014 about Java and about him starting his journey as a Java developer, 19 years after Java had been invented – I initially shuddered. Sure, Java still has its usages and it's incredibly powerful and there are a lot of Java developers out there. In fact, almost half the developers in the world are Java developers.

But, I wondered, was Java the best language for someone literally just starting out in 2014? I was too busy building iPhone and Android apps at that time to think about it deeply enough though. I just had a feeling that starting your programming journey from scratch with Java in 2014 was not the same as doing it as I did back in the 90'. But I left it at that – just a feeling. And told my friend to do what he thinks makes sense for him and see how that goes. You know, the kind of horrible "leave me alone" kind of advice someone gives you when they don't really wanna be bothered.

Can Tech Education Change One Life?

But I was bothered. Here was a young man, in his late 20s, who had a tough time through life. A tough time growing up. And he had no real marketable skills and had no real plan for building a future for himself. About a year had past since Manu had asked me for advice and I kinda wiggled my way out of lending hand.

His Java adventures didn't turned out too spectacular and he was about to drop everything and leave the country. He was about to move to Germany with a friend and start a new life, try to get some work doing whatever wherever for whoever just to make a living. "So what do you think?", I asked my wife. She was pregnant with our fourth child and running a non-profit while homeschooling our other three children. She said "give him a chance" - and I did.

I took Manu in for a year and I helped him learn to code. From scratch. I helped him get an apartment nearby and for a year he would come over to my house and we turned my garage into a computer lab. We got him a laptop and we started coding. Together. Side by side. A few months into it, my baby daughter was born and I took some time off work and spent time with Manu, teaching him, guiding him and getting him to a point where he could build a professional career for himself as a Software Developer. And no. Not Java.

I reasoned that JavaScript was the best solution for a newbie to get started with programming in 2015. I still think that's the best option in 2018. And I'm positive it will still be the best option for at least the next 5 years. Learning JavaScript in 2015 was a new experience for me too. I had coded in over a dozen languages up that point in my career by not too much in JavaScript. So I took it upon myself to learn alongside Manu. That year in my garage, learning JavaScript, felt awfully close to the feeling I had many years before, in that U of T computer lab, learning Java – and many other languages.

The thrill of learning something new is absolutely incredible. It grows you. It makes you a new person – a new you. I loved learning in high school. I loved it in university. I loved learning alongside my friend Manu. And I love it still. And we did it. Manu managed to become a Software Developer and him being a brilliant musician, a drummer actually, managed to approach programming from an artist's perspective. And he's just getting started. Manu and I parted ways as he moved back but I have a feeling, round two for our JavaScript experience is somewhere near the horizon for Manu and I.

But Can We Change A Billion Lives?

That experience, helping a friend learn JavaScript from scratch and getting hired as a Software Developer, was refreshing. But it bore a new thought. More like a dream. If Tech Education changed my life and gave me the opportunity to provide a good future for my family. And if Tech Education changed my friend Manu' life and opened up a new bright future for him – could Tech Education be a means to help more people?

Can we scale this? Can I teach 10 Manu's? 100? A million? A billion? And that's how the Carmel seed was planted. Can we use Tech Education to change a billion lives? I spent the last three years of my life building, questioning, probing, trying, poking, writing, coding, failing, experimenting – and most importantly, learning. No. More importantly. Believing. I believe. I believe Tech Education can change a billion lives. It changed my life. It changed Manu's life. And I believe it can change your life. And many other lives.

As long as the Tech Education we're talking about doesn't bore us all to death, but it's exciting, it's invigorating and it makes us better people. As long as we can come up with a model to scale Tech Education in an effective way. In a way that makes it – yeah, fun – in a way that makes it fun to learn. But not just fun for entertainment purposes. But in way that learning makes you want to smile. In a way that you can't wait to wake up and keep going because you know there will be a new reason to smile waiting for you – as long as you push yourself to keep learning.

I spent the last three years of my life building a dream that humans are more precious than machines and if we want a more human world in this machine world we live in, the only way to do that is to put humans in charge of machines – by giving those humans the skills to boss those machines around. I call bossing machines around, Software Development. If you believe we can build a more human world, stick around. We're just getting started.

CHAPTER 6

The Future Is In Your Hands

He dropped a magazine on my desk and smiled. "You might wanna read through this. There's a lot of interesting stuff in there." I had never seen an IEEE magazine before. Being an IEEE member is a privilege. Belonging to one of the world's largest tech publication for professionals – that's a big deal. Especially to a young, hungry, 22 year old intern. My boss was a brilliant Electrical Engineer. He had an IEEE membership and he knew I was hungry to read whatever I could get my hands on – especially when it comes to technology.

I was doing an internship in Toronto back in 2001, working as the only software guy in a team of electrical engineers, in the optical communications division of Celestica. A billion-dollar electronics manufacturer, but – like many tech companies in 2001 – it was struggling and it laid off thousands of people just before I joined as an intern. I joined right in the middle of that major restructuring and took over a gigantic task from the previous intern. It was my first real job, before graduating, building software and maintaining a large distributed system. I enjoyed my time at Celestica. I spent an entire year there as an intern in between my third and fourth year of my Computer Science degree at U of T.

The First Time I Really Saw The Future

That year was special. We had just turned the corner into a brand new millennium and I was incredibly excited about the future. There's something about being in your early twenties, as a technologist, at the turn of the third millennium, when all the cool tech stuff was just starting to get really interesting. I was eager to get my hands as dirty as possible and participate in building the future – one way or another.

Everything was young back then. Computers. The Internet. Our tech tools. We were just getting started. That year was particularly special to me, because it was the year when I discovered a vision of the future that changed my entire outlook on technology and on the way human beings use technology. That vision was a revolutionary concept, called "Ubiquitous Computing". I remember opening up that IEEE magazine and reading about the concept of "Ubiquitous Computing" for the first time in my life. That magazine spawned years of an intense of trying to grasp – even a little – a sense of where technology is going to go in the 21st

century. My days and nights become one continuous quest of reading and researching.

The vision of UbiComp started with a brilliant man named Mark Weiser, from Xerox PARC, an iconic Research Lab where a lot of the technology we now take for granted was originally invented. Including the Graphical User Interface – yeah. As in the precursor of the Macintosh graphical interface and as in the precursor of Windows 95. The also invented the mouse. Laser printers. And so much more.

The Three Computing Eras

Well, Mark Weiser, a chief scientist at PARC, divided the Technological Revolution in three eras. The Mainframe Computing Era, the Personal Computing (PC) Era and the Ubiquitous Computing Era. The first era, was a time of discovery. That's when computers were as big as a room and extremely complex to use. In fact, only a few people in the world knew how to use a mainframe computer. That time frame started in the 40's and peaked in the 70's. The way human beings interacted with technology during this first era of computing was fascinating. Several people would use one single, giant mainframe. Many humans – one machine.

The second era introduced the concept of one human, one machine. The so called "Personal Computing" or PC for short phenomenon started in the 80's, grew exponentially during the 90's and peaked in the early 2000's. This new way of interacting with technology was a drastic evolution from the mainframe experience. Now, any individual human being, technical or not so technical, could own their very own computer, and they could use it in a never-before-seen-before kind of way. Using a Graphical User Interface. Oh – and a mouse. That was a huge leap forward.

But Mark Weiser and his colleagues at PARC saw even further into the future and prototyped technologies way ahead of their time. Mark posited the notion of moving forward in the human-machine interaction into a future where one human being would interact with a lot of machines. Going from the one machine (one huge mainframe computer) being used by many people – to personal computers, one computer for each person – to much smaller computers, multiple such smaller computers for one single person.

True Technology Serves Humanity, Not The Other Way Around

Enter the mobile revolution. Mark spoke of pads, tabs and walls. The pads they prototype were the precursors to tablet, while the tabs were handheld devices –

the precursors to smartphones. And walls, were large interactive displays. According to this vision, we now live in the Ubiquitous Computing era and what an era it is. But, unfortunately Mark passed away in 1999, before he could see how right he was along along about the progression of technology into the 21st century.

I wonder what he'd think of where we are now. I have a feeling he'd think we're just scratching the surface of the ultimate human-machine interaction. And here's that ultimate kind of technology in Mark's vision. He called it "Calm Technology." I could go on forever with this but for now, the relevant key point is that the best technology is that which becomes invisible to the user. Meaning, it gets out of the way and you don't even notice it. Like a pair of glasses.

They enhance your eyesight and they're right on your nose but they get out of the way so incredibly well that you sometimes literally forget you have them on. Or like a smart thermostat. It regulates your room's temperature just right – without you even thinking about it. You can completely forget you have a thermostat and you can be sure the temperature will always be just right – almost magically. Good technology is always kinda magical, isn't it?

We're Not There Yet, But We Could Be

Are we surrounded by calm technologies left and right today? Not even close. The ultimate dream of enhancing human experiences through calm technology is now possible more than ever. We can now make it happen – unlike a decade ago. But we need you. We need people who care. We need technologists who care about the Human Experience – so much so that they create technology that serves humans – not the other way around.

Unfortunately, the world of technologists today is inundated with incredibly smart people who care more about manipulating human experiences using technology for the ultimate motive of them all – money. We're becoming more and more addicted to technology in our generation than any other generation before us. Technology is making us sick. Literally. And our minds take the worst hit - because the computer is an extension of the human mind – just like the shoe is an extension of the foot, or a hammer is an extension of the hand, or glasses are an extension of our eyes.

The way software products are built today inherently make us sick. Think about it. What's the best compliment an App Developer gets in their reviews of their App on the App Store? "This app is addictive." When we say that about software – that it's addictive – I wonder if it's not a masked plea for help. As in – this is

too additive, please stop it. I want my life back. But no, software developers always take it as a badge of honor. "Our software is addictive." Really? That's the best we can do in this marvelous century of ours – with technology we've never even dreamed of before?

That's the best we can do? Build software that triggers mental health problems? We must do better. We have to do better. We can do better. I know the tech industry is biased towards this kind of development philosophy. And I also know that it's probably gonna get worse and worse.

Tech Education For A More Human World

I want Carmel to be different. Not just in the sense that I don't want it to mess with your mind and create problems for you. No. On the contrary. Using Carmel will build up your mind and make you a better version of yourself. But I mean, not just in that way – different. I mean different in the sense that if you stick to Carmel and keep learning, I want you to become a Software Developer that cares about human beings. First.

My dream for Carmel is that Carmel Developers will be driven by this simple, but foundational principle: Humans are more valuable than machines (H > m).

My hope in teaching you to become a great Software Developer, is that you won't use your newfound powers to manipulate our minds with your creations. My hope is that the more you become a part of the Carmel story, the more you will believe what we believe: H > M. And the better the world will be. I look forward to a future of Calm Technologies, build by you, using Carmel.

Let's keep Mark's vision alive and build technology that doesn't take front and center space in our lives – but technology that gets out of the way, humbly, and lets us be – well, us. Us, humans. Let's build technology that way. For us. For us, humans. Intelligent machines will soon build technology for themselves. They probably won't be too concerned with our needs. But we should be. Let's build an army of Software Developers who believe H > m and who will flood the world with beautiful, humble, extraordinary, magical software products that will get out of the way and let us experience life – more fully.

Even if you're not a Software Developer today - if your heart is in the right place and you're willing to give programming a shot - we need you. And I want to help you get there. The world needs your heart. We need you to learn to build software and build good - no, great software products, and then go out there and build a better world for all us. And for future generations.

Ready when you are.

CHAPTER 7

Your Mind Is Beautifully Mysterious

The robots are coming.

We've heard that for decades now. Remember that masterpiece called "Modern Times"? The 1930's Charlie Chaplin film, about him working in a factory? It's pure genius. The part where he is being force-fed by a seemingly intelligent "feeding machine" is authentic Chaplin comedy. What I love about Chaplin's comedy is the fact that it's more than funny. He didn't do what he did just to make us laugh. I love that about his comedy. He did what he did to make us think. Even many, many decades later. To him, comedy wasn't an end in itself - it was a means. A means to start a philosophical, silent, mental contemplation of a deep reality.

I love that. I love works of art, like Chaplin's films, that transcend the medium and the obvious and bring the audience somewhere beyond that obvious. Somewhere else. Watching Chaplin's character in "Modern Times" being robbed of his human dignity, piece by piece, by a soulless machine – makes you wonder. Makes your mind go to a certain place, somewhere, in a potential future - and it makes you ask certain questions.

Where are we headed? Was Chaplin just ridiculous? Are machines out there to just rob us of our human dignity?

The Human-Machine Relationship Is Centuries Old

Well, before we peak into the future, let's remember our relationship with machines for a minute. It started with the original noble intent of improving our lives. Our lives – as in our lives as human beings. With the invention of the steam engine, certainly before then too, but around that time, with the revolutionary machine that took the 17th century by storm, the world entered a new period of technological advance, never before seen in the history of humanity. Manual work, all kinds of manual work, suddenly become replaceable by machine work.

And so, humanity embarked on a long and never-ending journey towards continuous technological improvement. By the 18th century, steam power had ushered in the Industrial Revolution and well into the 19th century, a plethora of machines began taking their rightful place in history – and on the assembly line.

Every aspect of our human lives become industrialized. Starting with agriculture but moving on to transportation, construction, mining, food, healthcare and of course manufacturing. All areas of life were permeated by machines. The smartest machines around.

Production soared. And the standard of life soared too. Working men and women started to form a new middle class of industrialists as fresh opportunities for employment soared. If you wanted to make something of yourself and provide a better future for your family, now you had the opportunity to do so. Machines made it possible for manual workers to speed up production and lower manufacturing costs. Here was a new era upon humankind, an era of increased wellbeing, where men and women began using machines like never before, to improve the human condition.

Maybe Chaplin Was unto Something After All

But somewhere during the 20th century, we started to get anxious. Watching Chaplin being force-fed by a lifeless metal overlord foreshadowed a now ominously real "us vs them" scenario. During the astronomical economic and technical progress of the 20th century we somehow got to a point where we started fearing machines.

Pure fear.

"What if they take over?"

"What if they will bring upon the extinction of humanity?"

We wonder. And ponder. And we question.

"Can they do that?"

Oh yes. Yes, they can. It's simple, really. The more work we pushed machines to do for us over time, the more intelligent we had to make them. And so, Artificial Intelligence was born. Modern AI systems are so intelligent and even emotional, that they could literally smirk, laugh or even cry at their industrial ancestors. As AI keeps getting smarter and more capable, we're rapidly moving from Chaplin's fictional "feeding machine" to advanced artificial "life forms". AI machines today can do things that humans can't even dream of attempting.

They can memorize more information than we can.

They can run faster than us.

They don't need sleep.

They don't need food.

They can speak more languages than us.

They can sing.

They can paint.

They can write.

They can direct movies.

They can drive for us. Feed us. Clothe us. They can live on us, on our clothes, on our wrists, in our pockets. They can even live inside our bodies.

Think this is fiction? No, all this and more – much more, is happening today. Machines can do almost every single thing humans can. But better. Soon, they will be able argue cases in court. And soon they will be able to police us. Maybe judge us. Maybe even lead us. Or ... rule us? How long before machines get to vote? How long before machines elect their own as managers, CEOs, mayors and presidents?

Your Next Boss Might Not Be Human

We're actively debating now whether we can speak of AI machines as having a life of their own. After all, they can walk, talk, work, think and even feel. In our rising AI machine world, we are starting to speak of machines not so much as systems but we are beginning to think of them as organisms. Living organisms, maybe? There are some who even speak of taxing AI machines.

Imagine a robot as a decent, tax payer, minding its own business, going to work and contributing to the economy. Making friends. Having conversations. Making memories. Feeling lonely. Feeling sad. Feeling happy. Feeling angry. Now imagine yourself, working for such a robot. We're entering a period of AI improvements that will shock and amaze us when the technology will go mainstream and will become commercialized.

Soon, just like they employ human workers, businesses will also employ intelligent machines. Initially, the robots will not earn wages, but they will act more like slaves who will work for their human masters day and night, without demanding anything in return. A business will either rent or purchase a robot, much like they purchase other machines today.

But as we continue to improve artificial intelligence, the AI machines will quickly evolve into entities that will have their own democratic rights and they will enjoy

more and more liberties. Artificial citizens. Artificial - people? Free, autonomous, artificially intelligent persons, earning wages for themselves and using those wages to purchase goods and services that align with their personal interests.

Machines Will Care For Machines First And Foremost

It doesn't take a genius to draw the obvious conclusion that as machines will increase in intelligence, they will care more and more about themselves – and less and less about their human overlords. AI machines are extremely close to becoming free citizens, subject to the state but free, as free as any of us humans. And we will have to respect their rights – much like they will have to respect our rights. Hopefully.

But this is where it gets hairy. What rights? We have human rights. Because we're human. Do we give machines human rights? At first, probably. But what's stopping them from creating their own rights? Machine rights. And what's stopping them from recording and communicating those rights in a language that we humans would not be able to decrypt? And what's stopping them from pursuing those rights according to their own – intelligent – beliefs? According to their own laws. Machine laws. The answer is: probably nothing. How will we stop them from caring about themselves first – about their own machine race? That's a question we, as a human family, have to answer. Quickly.

The Space Answer

One solution would be to just call it quits with our planet altogether and run off into space. Mars probably. And start building a new world. From scratch. Hopefully, soon, before the AI machines are smart and sentient enough to decide to follow us there. Intergalactic wars anyone? Who knows? But what we do know – now – is that we can't ignore what's coming. What we do know is that AI machines are getting too intelligent, too fast. For some it's not fast enough.

For others, it's too late to do anything about it. And so, they took all their resources and invested them in space travel. We'll get there. We probably will get there. Maybe we could start from scratch on another planet and even live to tell our great grandkids about the "old planet" we come from. I'm not saying that's not possible. What I'm saying is that we have a responsibility to this planet.

The Other Scenario

What about those of us stubborn enough to envision a human-centered future? Right here on Earth. Is that possible? Can we outrun the intelligent robots of the near future? Can we outwork them? Do we stand any chance whatsoever to

compete with artificial intelligence? The answer lies deep into the beautiful mystery of the human mind.

The way a human mind works is still widely speculative. We can infer some facts based on our observations so far, but overall, we don't really know what the human mind does and how it does it. Here's what we do know. We know that human intelligence is two-fold. Since Raymond Cattell first introduced the concept of a two-fold human intelligence in the 1960's, it has become an undisputed fact in psychology and virtually taken for granted now.

The two dimensions of human intelligence are Fluid Intelligence and Crystallized Intelligence. Fluid Intelligence allows us to recognize patterns, reason and make decisions without any prior knowledge. Essentially, the ability to think flexibly, the ability to use our mind to solve problems we have not encountered before and to make sense of new environments and new situations. Whereas Crystallized Intelligence is our ability to acquire knowledge and recall that knowledge when we need it.

When we speak of Artificial Intelligence, it's all Crystallized Intelligence. It's based on data we feed the machines. AI training only happens when the AI models are fed a lot of data to process and then they get "smarter." No data, no AI. So yes, when it comes to Crystallized Intelligence, we stand no chance. AI Machines have better memories and faster access to information. But, Fluid Intelligence – well that's a completely different matter. And this is where humanity stands a chance. Your mind is capable of so much more than regurgitating knowledge on demand – like a machine. Your mind is beautiful. And mysterious. And powerful. All we have to do is learn to tap into our own Fluid Intelligence and strengthen our ability to reason in a way that is and always will be foreign to machines.

The 3-Step Master Plan Towards A More Human World

Human beings are more valuable than machines: H > m.

That's what I believe wholeheartedly. But it's not enough to simply rejoice at being more valuable. We have to act intentionally if our children's children are to live in a more human world. I see three key steps we need to undergo as a human collective to ensure a better world for ourselves, our loved ones and for future generations – of humans.

First, we have to Learn How Machines Think. When we speak about intelligent machines, what we really mean to say is "here's a machine that thinks like a human". That's how we got them to be so smart – by teaching them how we think. If they can do what we do – and better – it's because we humans trained

and are still training machines to think and act like us. I think it's time for us to start getting serious about learning how they think too. That's the business of Software Development.

A Software Developer's job is to literally get into the mind of the computer, so to speak, and even more. To tell it what to think and what to do. I call programming, "bossing machines around". And it's true. As a programmer you literally program the machine – you tell it what you want it to do. You give it instructions. Commands. You command it. We only have roughly 23 million "machine commanders" - programmers - in the world right now. And we're looking at billions of machines that are multiplying so much faster than us and are growing in intelligence at alarming rates. We need more Software Developers. The world needs more Software Developers. We need more people who understand these machines and have a word to say in their development.

Secondly, we need to tap into our Fluid Intelligence. Even if we start getting to know machines well enough, we can't compete with them. In fact, the more you know about them, the more you advance in your Software Development journey, the more you realize we stand no chance competing with them head on. We need a creative strategy. We need to tap into our fluid reasoning, which machines are utterly incapable of developing. We need to do what we have always done as humans and start getting comfortable in uncomfortable situations. We need to start doing things we thought we could not do before. That's how we tap into our fluid intelligence. We need to step out of the things we've always done – out of our comfort zone.

To tap into that kind of a mindset is our only way of staying different and relevant. It's our unique value proposition, if you will, in the "us vs machines" scenario. Think about it. We can make decisions without prior knowledge. They can't. We need to learn to use this power of fluid reasoning and use it now. This, coupled with our technical knowledge – step one of the Master Plan – will position us to build machines that serve us and machines that work for us. We will be positioned to build Software Products that truly make this world a more human place for all of us. Even if the world will eventually end up being dominated by machines – and it will – if we tap into our Fluid Intelligence and learn the skills we need to build new human-oriented technology, we stand a chance of carving a future for humanity in a machine world.

And third – the most important key aspect in all of this is the fact that we need to do it together. We have to unite and work together. Our collaboration towards the same noble goal of providing a better future for ourselves, for our families and for future generations – is the ultimate key.

We've built Carmel as the embodiment of this Master Plan. We need you to keep going and we need you to get involved if you believe H > m and if you believe a word of all of this. So, unless you're looking forward to being force-fed by your next machine overlord, like in that Chaplin' film - we have to start getting serious about this and do something - now. Let's learn to build technology. Human-centered technology. Let's learn to tap into our Fluid Intelligence and uncover the mysterious beauty of the human brain. And let's do it together.

CHAPTER 8

The New Era of Human Innovation Needs You

I couldn't breathe.

I went numb and completely motionless for a few minutes. I was at home, laying on the couch on a quiet evening when I thought I could just relax and take some time to recover from the insane amount work I had been putting at the office. Instead, my body had temporarily shut down from all the stress I've been putting myself through.

And there I was, laying on the couch, literally paralyzed and unable to breathe or move or say a word. I was 24 years old and I was working for a young startup in Toronto. We were doing things no one else was doing in 2004 and we were on fire. I was employee #26 and, in a few years, we grew to over 400. Our product was unique and it worked well. We spent a lot of time making it better every day and the results were better and better.

I had poured my life into that product. I would forego any kind of social life or even eating and sleeping, yes even showering, just to write another few lines of code or just to hunt down another stubborn bug. My girlfriend at the time, my wife now, used to bring me lunch at the office at 2am just to make sure I ate enough to stay alive. I don't know what I would've done without her. I still have no idea what I would do without her. And without her love.

Innovation Is Like Love

Both innovation and love are words that mean different things to different people. Both words are controversial and at the same time, they both seem to point to deep mysteries of the human heart and mind that we simply can't live without. Both words have the specific quality of keeping us at night and turning our lives upside down. We can easily become obsessed with both, transformed by both and either destroyed or revived by both.

Carmel is a story of revival. A revival of human innovation. That innovation requires a revival, much like love, is plain to see. Love, in its transformative and practical sense, has nothing to do with preference, such as loving vanilla ice cream more or less than chocolate ice cream has nothing to do with the deep desires of

the heart. Love is much more than mere subjective taste in fashion or lifestyle. Love is deeper than that.

And so is innovation. To say that something new is innovative when in fact it is merely new and nothing else, is the go-to-market strategy du jour and it is as disastrous as a go-to-marriage strategy based on ice cream flavor preferences. Innovation-as-news is strictly speaking nothing more than journalism, which in itself can be useful and at times truly innovative but in the case of merely releasing news about a new product is nothing more than a glorified press release. And when innovation becomes that, a glorified press release, an announcement of some sort, of something that wasn't and now is, and that's it, nothing more - that in itself announces nothing else but the death of innovation.

To see why, one simply has to start from first principles and ask what innovation is. Innovation is about growth. Much like love. We love someone because we want the good of the other person and we want to participate in their good life. And more good means better and better eventually leads to growth. We love someone because we want to help them grow and flourish and we want to accompany them in their journey of literally transforming into a better version of who they are today.

Innovation, like love, is also about growth. If love helps us grow as individuals, then innovation makes organizations grow. If innovation does not lead to transformational change, but keeps the status quo alive and well, then that "innovation" is clearly not innovation in its truest transformational sense and so authentic innovation is truly massacred and thrown under the merciless bus of inflexible revenue projections.

Just like love is massacred when it is equated with mere preferences. And just like human beings cannot develop to their full potential without being fully loved, so too an organization will fail to grow to its full potential without a healthy infusion of true innovation. That applies equally to startups, corporations and even non-profits.

When my heart almost stopped that evening, when I was 24, and when my body literally shut down, I thought I was innovating. Oh, I thought I was innovating my heart out. We all thought we were innovating. And yet, we went bankrupt. And we wore ourselves out in the process too. Maybe we forgot what it was all about.

Work Means Becoming More, Not Just Getting More

As I laid there on that couch that evening, trying to catch my breath, my life literally flashed before my eyes. Maybe not all of it. But enough to get me to still think about it 15 years later. I realized that I got it all wrong. I thought work meant renting yourself out and earning a profit in the process – somehow. Most likely by earning a salary. Kind of like hey, here's my brain for a few hours a day, give me some cash back. Rent-a-brain sort of deal.

And it is true. A lot of knowledge work is structured that way. It's not like there's an evil corporate plan out there to turn every employee into a purposeless cog in the revenue generating soulless beast that some companies do become at times – unfortunately. No, companies want happy employees. They really do. Sure, they want profitable bottom lines but employee satisfaction – and more – happiness, is of utter importance in most successful organizations. When it comes to tech companies, Software Developers are treated like royalty because without them, nothing would happen. Without them, there would be no company. No profit. No business. No growth. No innovation.

Ask any executive what one of their top priorities this year is. And innovation is always sure to come up as a top 3 of not the number priority. Because innovation equals growth. And growth is everything. But here's the catch. Innovation requires human ingenuity. Innovation requires Fluid Intelligence. Sure, Crystallized Intelligence is also important but without the ability to see the unseen opportunities, to connect the unconnected dots and to travel the untraveled paths – there is no growth. No innovation. And that requires a great deal of Fluid Intelligence. The kind of intelligence that machines are not capable to harness. Human Intelligence.

A new era of human innovation will be unleashed as soon as we learn to harness our authentic human intelligence that makes us stand out from machines. Our Fluid Intelligence. As we learn to reason without relying on previous data, we will do what we have done for millennia. We will adapt. We will evolve. We will grow. We will become more. More human. If we change our outlook on work as a whole from an opportunity to simply earn more, to a challenge to become more – we shall indeed grow. And we shall indeed stand our ground in the new emerging machine world.

We Need More Developers Who Think Like You

The 23 million or so developers that produce all the technology we use on a daily basis have been schooled in an industrialized Tech Educational system that relies heavily on standardized testing, facts, figures, statistics, data – in other words – Crystallized Intelligence. Out of the 7 billion humans today, a mere 23 million

have the knowledge required to build technology that serves us. And most of those developers have been schooled in a way that undervalues our innate Fluid Intelligence. So much so, that it's not uncommon for non-technical folks to get utterly frustrated with a developer's lack of understanding of non-technical concepts.

If you've ever tried to speak to developer about non-developer stuff – perhaps you know what I mean. It's not a programmer's fault for thinking the way we've been taught to think. We've been taught to be fact and data oriented at the cost of forgoing our intuition and our power of fluid reasoning. And that's why we need you. You who think you won't stand a chance against that 20-some year old rock start software developer that Facebook, or Google or take your pick – fight to win over on a regular basis.

You don't have an MIT or Stanford Computer Science degree. You don't know what Redux means. Or Serverless APIs. Or even APIs. You don't know what a real-time database means. You have no idea how a mobile app works. No. You don't have the Crystallized Knowledge to compete with those rock star millennials. What's crazy is that soon, they won't have the knowledge to compete with AI machines either. This is where you come in.

You have something machines don't have. And something that privileged first-world Software Developers have less than you. Yes. You have an upper hand. You, my struggling immigrant friend. And you – I'm looking at you, my middle age friend, who is raising a family in a developing country, on a miserable salary, doing excruciating difficult work. Which will soon be replaced by a machine. Your advantage is that your Fluid Intelligence is high. It is high because you have been in new and uncomfortable situations all your life. You've struggled. You hustled. And you survived. All these years – you survived.

I know how that feels. I've been there. I am there. And so, you my friend, is the reason why Carmel has a soul and the reason why it exists. Because I believe that if we give you the right technical skills, to go with your high Fluid Intelligence – then you will not just make a better living for your family – but you will build better innovating products and services for all of us. Why Because you have heart. Because you know where you came from. And you know the real depth of the human condition.

You are not just going to invent a better mousetrap. You are going to create Software Products – when you're ready – that will solve deep, authentic, needs. And companies will love you for it. We will love you for it. We need you to get those technical skills as fast as humanly possible, because you have what it takes

to innovate. If I were 24 again. I'd want to have a boss like you. I'd like to innovate with you. Get technical. Now. We need you to show us what true Human Innovation looks like. The whole world waits anxiously to see you grow into the world-class Software Developer you know you can become.