



# Onboarding the Right Technical Talent to Build Your Startup

Itzik Paz

Startup Solutions Architect







## Itzik Paz

### **Startup Solutions Architect**

### **AWS**

Startup Solutions Architect. Itzik has 20+ years of experience in development, architecture, and IT. He's been in startups for over 13 years before joining AWS 5 years ago where he's been part of founding teams and a CTO. In his current role he's helping startups with their cloud solutions, mentoring, and running workshops for customers. Itzik is also part of the healthcare and life sciences team.

**Are you ready to make  
your first technical hire?**

# Hiring isn't easy

It takes months to find the right candidate and refine your hiring process.

In my previous startups it took about 2 months to find.



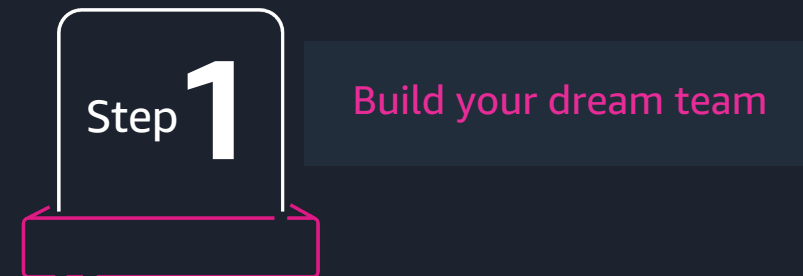
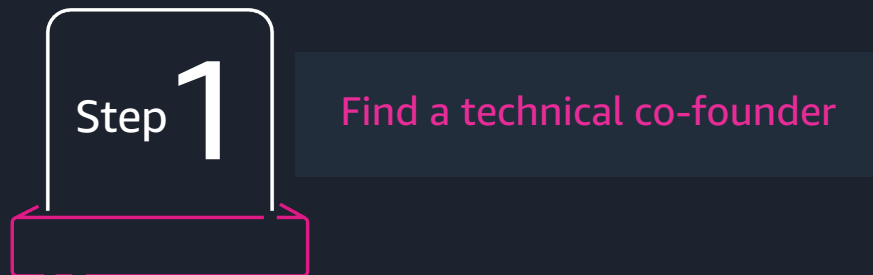
# How much tech do you understand?

## Non-Technical Founder

- Validate your idea
- Boost investor confidence
- De-risk technical contractors

## Technical Founder

- Manage your technical dependencies
- Hit the next big milestone
- Raise the bar
- Graduate from developer to CTO



# What exactly are you developing? Tackle the projects by stage.

## Early Days = Ideation & DIY

- Prototype
- Find Your Technical Co-founder
- Key Milestone Work Back Plan

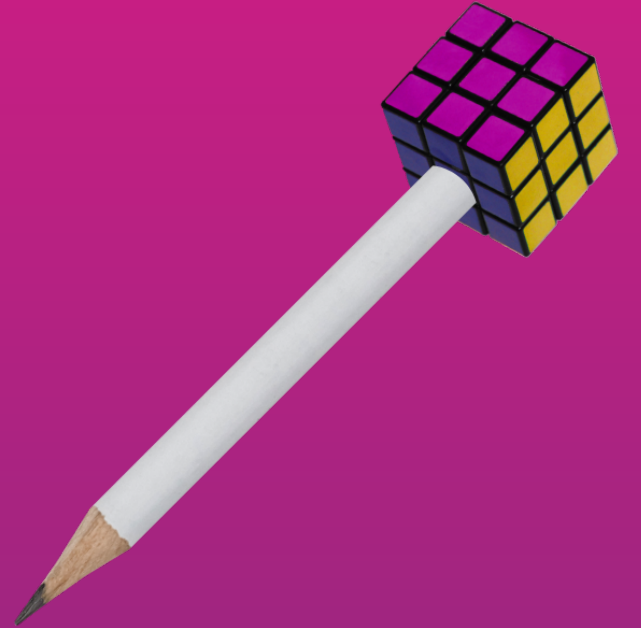
## Present Day = Fundraise & Hands-on

- Iterate Your MVP
- Hands-on Development
- Managing Developers
- Early Customer Development
- Fundraising

## Future Days = Speed & Scale

- Product-Market Fit
- Managing Dependencies
- Customer Management
- Hire & Develop the Best

**Make a list and  
check it twice.**



# 7 tips before you hire, anyone

1. Get the legal part nailed down.
2. Create a hiring roadmap for all roles, not just technical.
3. List the specific skillset and depth of knowledge your dream technical hire should have.
4. Consider the characteristics of who you should bring in that will shape your startup's culture.
5. Document procedures, process, tasks, and goals.
6. Build a pipeline.
7. Create an onboarding doc.
8. Find a technical problem to use an example for talking through development and testing.





# Finding technical talent.

# Go where they're at

1

## Personal Networks & Advisors

2

## Developer Communities

GitHub, Stack Overflow, Women Who Code, Black Girls Code, Meetup Developer groups, AWS Developer User Groups, Hackathons, Indie Hackers, Slack User Groups, Inbound, Hashnode, Dev.to

3

## Early Adopters, Beta Customers, Open Source Contributors

4

## Social Communities

Virtual co-working spaces, Meetups & virtual events, AngelList, LinkedIn

5

## Generate Inbound Requests

Create compelling content that attracts developers - blog, social media, PR influencer marketing

6

## Hiring Marketplaces

A-list, workinstartups, techcofounder.com

# How to approach



# Does location matter?



## Centralized

- Collaboration & hands-on
- Ease of onboarding & training
- Engage & network with
- A place to call home



## Remote

- It's today's reality
- Employee flexibility
- Increased productivity
- Increased savings
- There's still a central place to go to, if needed



## Distributed

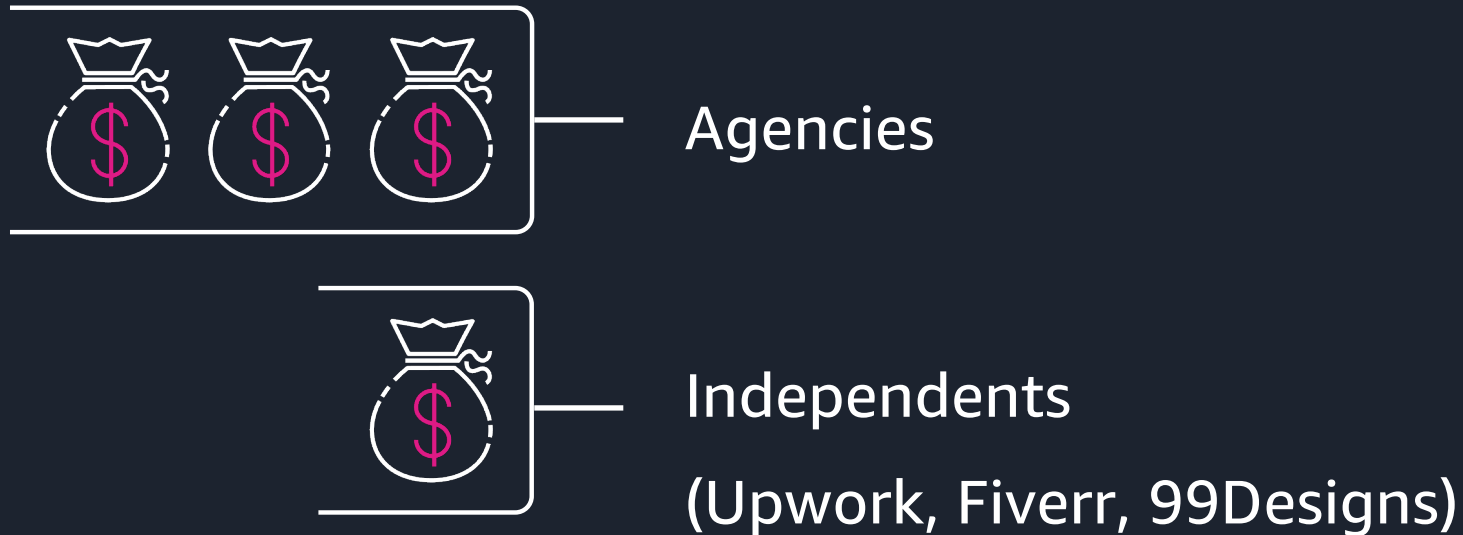
- Wider pool of candidates
- Ensures diversity
- Bigger cost savings
- Improved record keeping
- Better use of internal team technology
- Easier to do earlier, than converting to it later



# What about contractors?

# So, what exactly is a technical contractor?

Tech contractors provide skills to clients on a business-to-business short-term 'contract', rather than becoming traditional employees.



# 10 things to get right with contractors

1. This is a short term fix.
2. For non-technical founders, this is your biggest risk.
3. Have an off boarding plan from the start. Time is money.
4. Do due diligence. Ask the hard questions.
5. Set your ownership terms: code, copyright, account maintenance.
6. Don't give away the farm: projects, equity, money.
7. Only engage when you need specialized skills
8. They're part of your team. Communicate often, hold them accountable.
9. When you find a star developer, hire them long-term.
10. Be realistic. Be selective.



# How to de-risk hiring: the interview process.

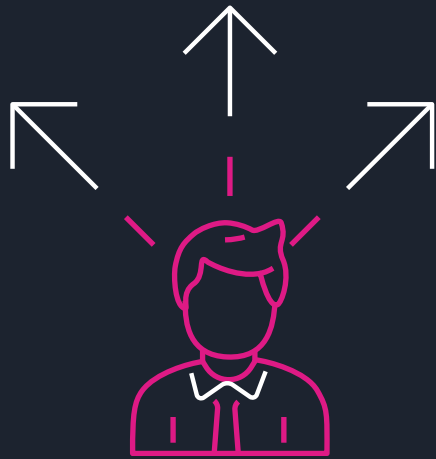
# Questions to ask

1. Non-technical founders, lean on your technical advisor for advice.
2. Ask for lots of examples. Share lots of examples.
3. Find out what they've built, not what their team built.
4. Focus on their last 3 projects.
5. How have they prioritized moving fast vs doing it right?
6. How do they design for failure?
7. Ask them to explain a tech concept like 'polymorphism' to a 10-year-old.
8. Which cloud providers are they most skilled in?
9. Give them a real-world technical problem to solve.
10. Make sure their recent experience is the same as your current technical stack.

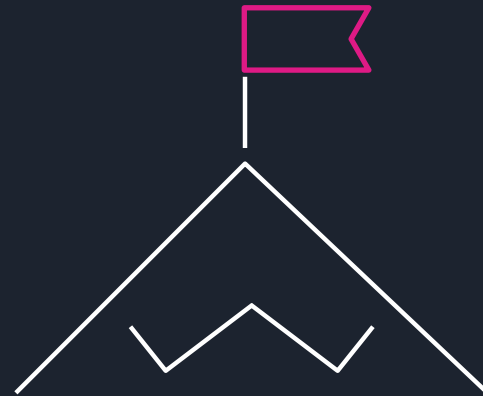


**It's more than compensation.  
Sell a vision and earn trust.**

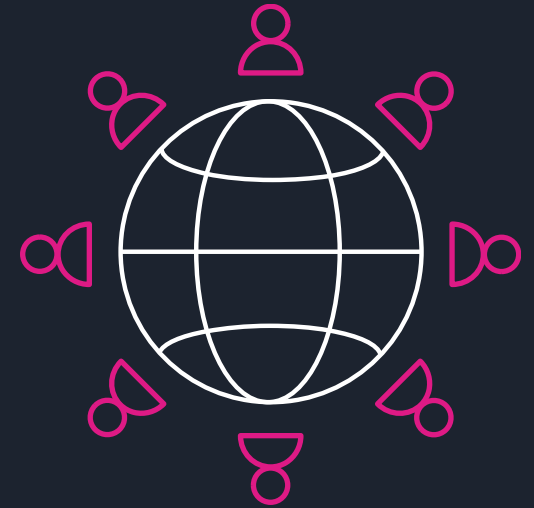
# What are technical hires looking for?



Opportunity for  
Growth & Impact



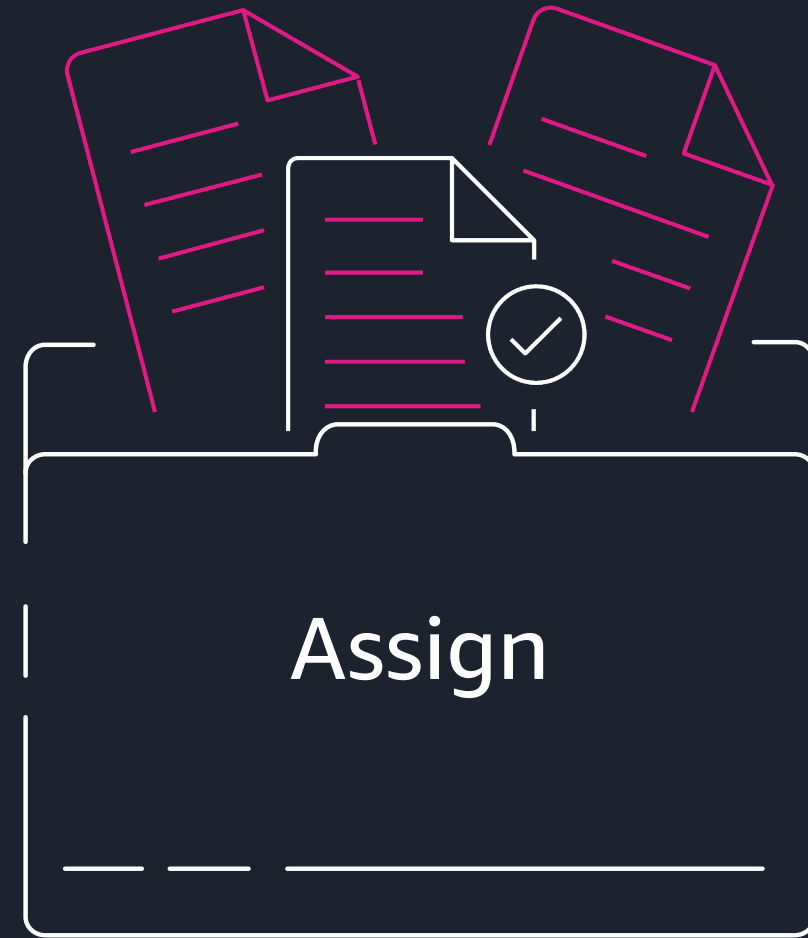
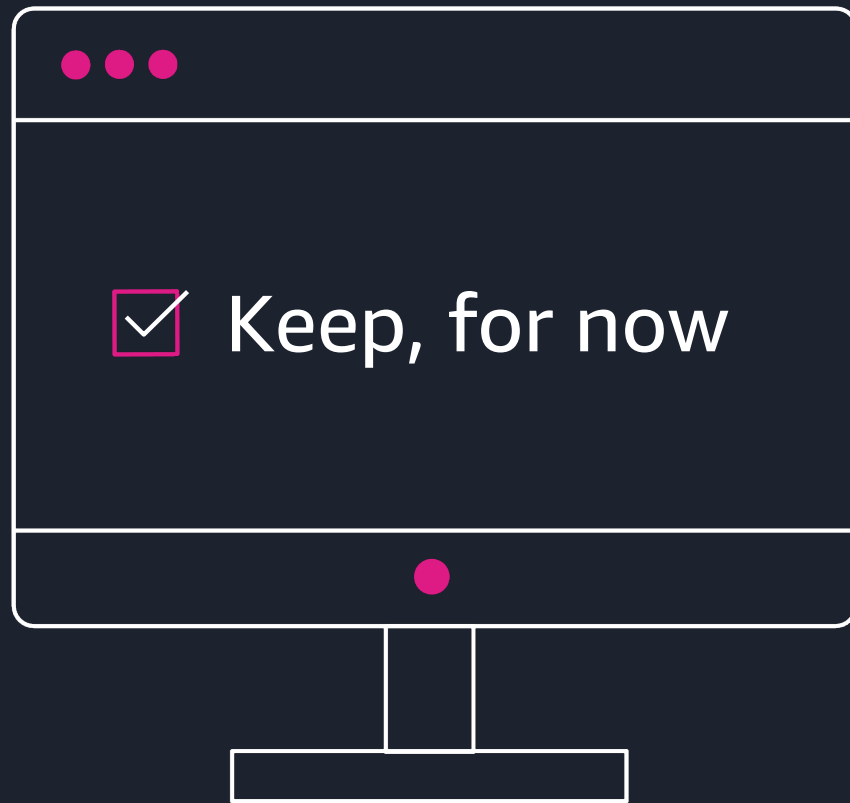
Challenge &  
Tech Stack



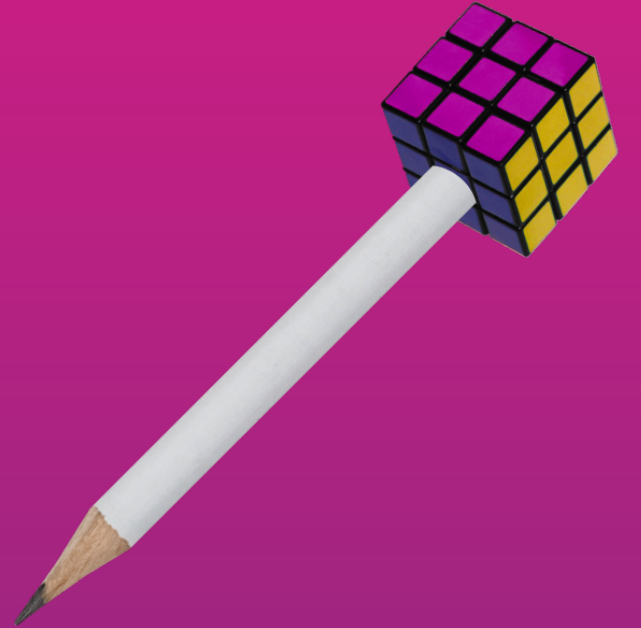
Environment  
& Culture

**What projects do I keep,  
what do I assign?**

# Start with smaller tasks that build up to a full project



**Avoid stumbling blocks.**







- Don't hire too quickly.
- Don't assume tech hires will be productive immediately.
- Avoid the fancy talkers – just because it sounds right, doesn't mean it will work.



- When you find a star, close quickly.
- Do make informed decisions.
- Do Hire & Develop the Best
- Always be recruiting.
- First hires should be generalists, and then graduate to specialists.



# Thank you

Itzik Paz  
Startup Solutions Architect  
AWS

