aws startups

This Is Not That MVP Session

Mike Apted
Principal SA, Startups

aws startups

"The ability to learn faster than your competitors may be the only sustainable competitive advantage."

Cindy Alvarez
Lean Customer Development

Mike Apted

Principal SA, Startups

Toronto, ON, Canada

🇨🇦 🐶 🏊 🚴 🏃 ⛷️ 🏒 🏳️‍🌈

@mikeapted
ln/mikeapted

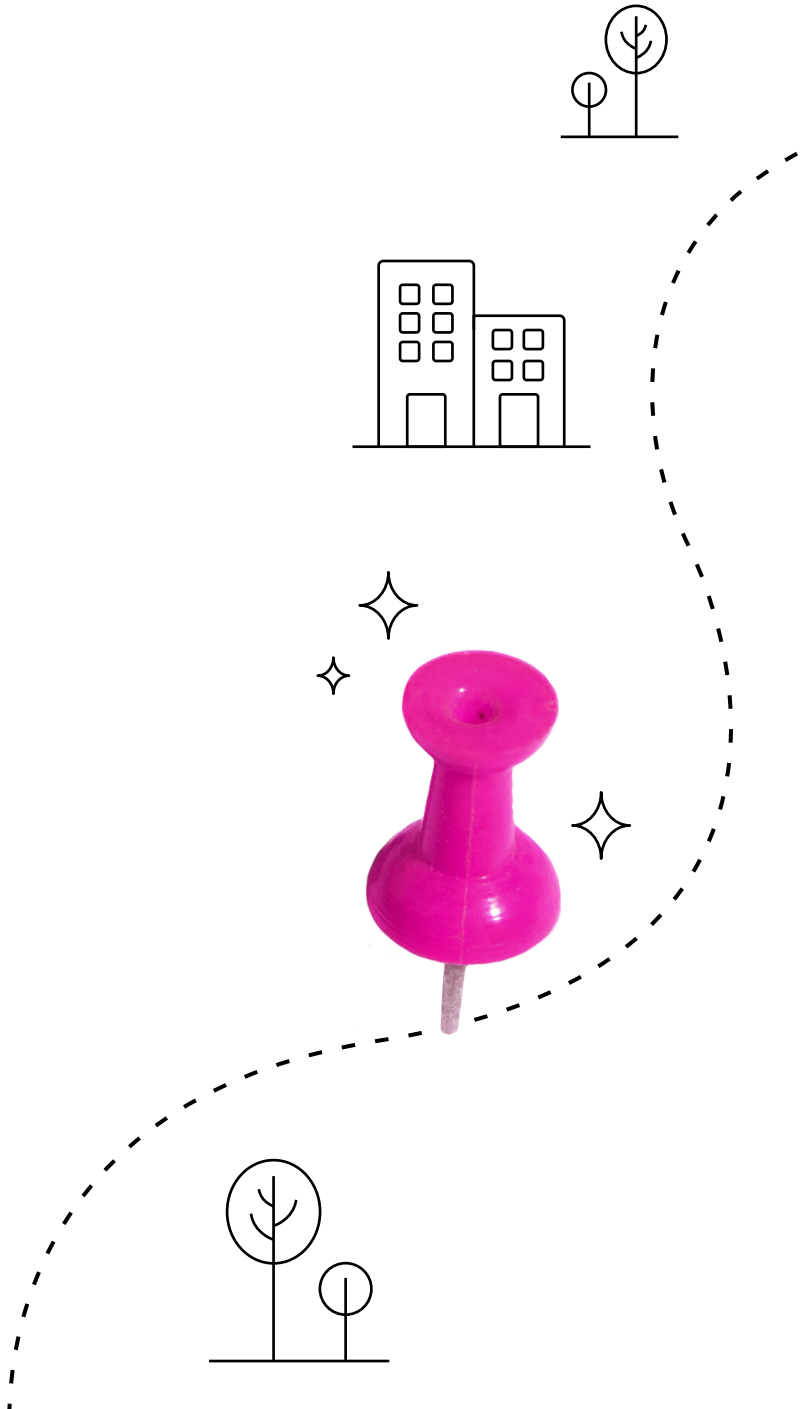
Founded, scaled, failed, and grown companies in the tech, sports, and the media & entertainment spaces. Currently enjoys working on the Startup Team at AWS, and is focused on helping startups live their best cloud life.

Our Focus Today

- Building an MVP from the perspective of a technical founder
- Personal experience and retrospective
- Lessons learned from working with hundreds of early stage startups
- What I would want to know starting fresh today
- What to prioritize and what can be deferred
- Tactical examples and first steps to get you started

Agenda

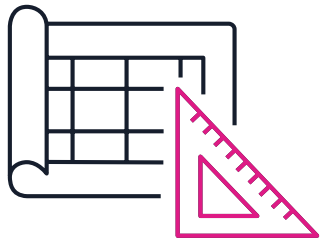
1. What is an MVP?
2. Before You Build
3. Leverage Frameworks
4. Flexible Architecture
5. Building Guardrails, Not Gates
6. Surviving the Unexpected
7. Understanding Your Users
8. Integrating Offline Tasks



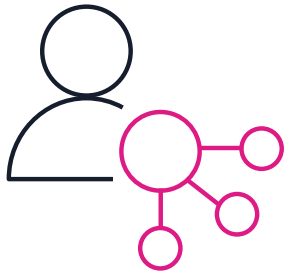
aws startups

What Is An MVP?

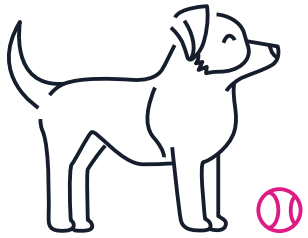
Minimum {X} Product



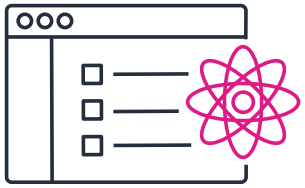
Viable



Usable



Loveable

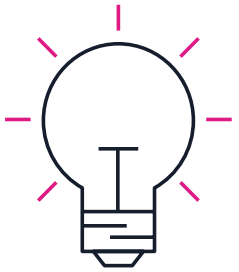


Testable

”Left to Right” vs “Bottom to Top”

Functionality

Reliability



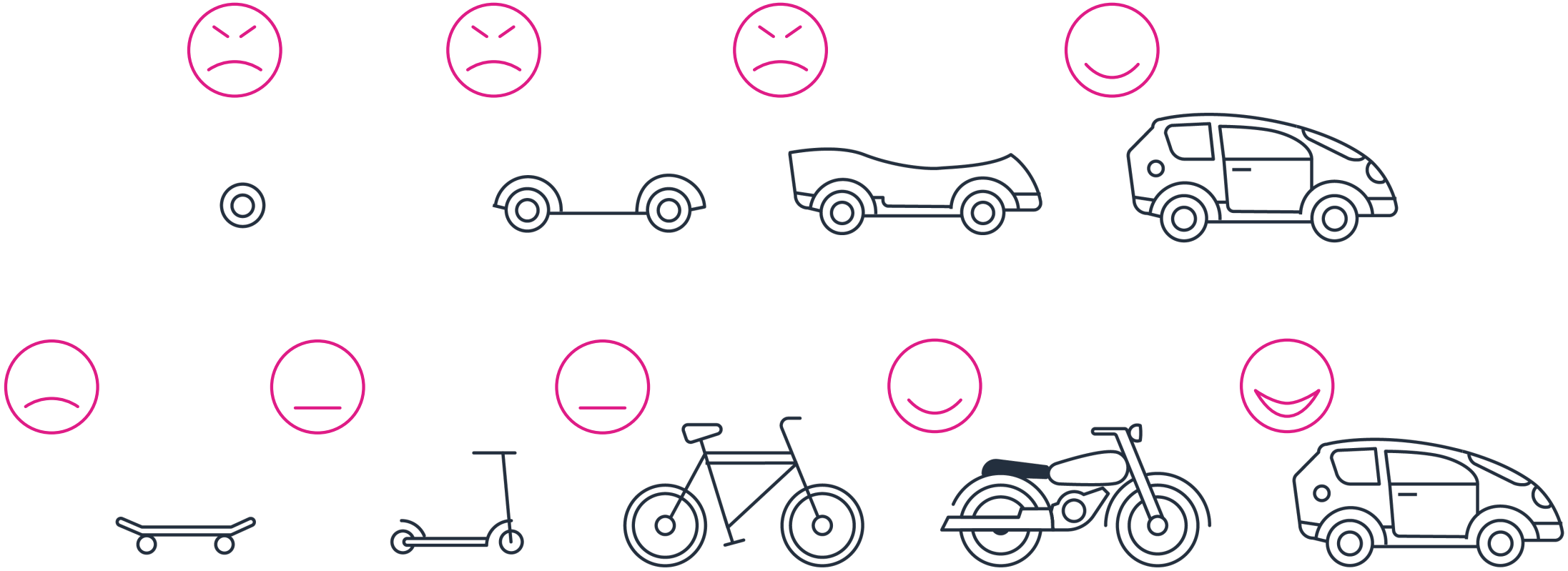
Usability

UX & Design

"If you aren't embarrassed by the first version of your product, you shipped too late."

Reid Hoffman
Co-Founder LinkedIn

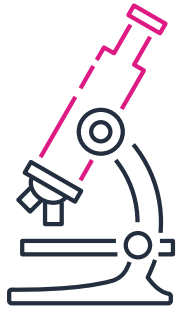
Product Iteration



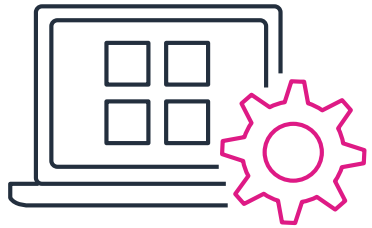
aws startups

Optimize For Learning

Learn

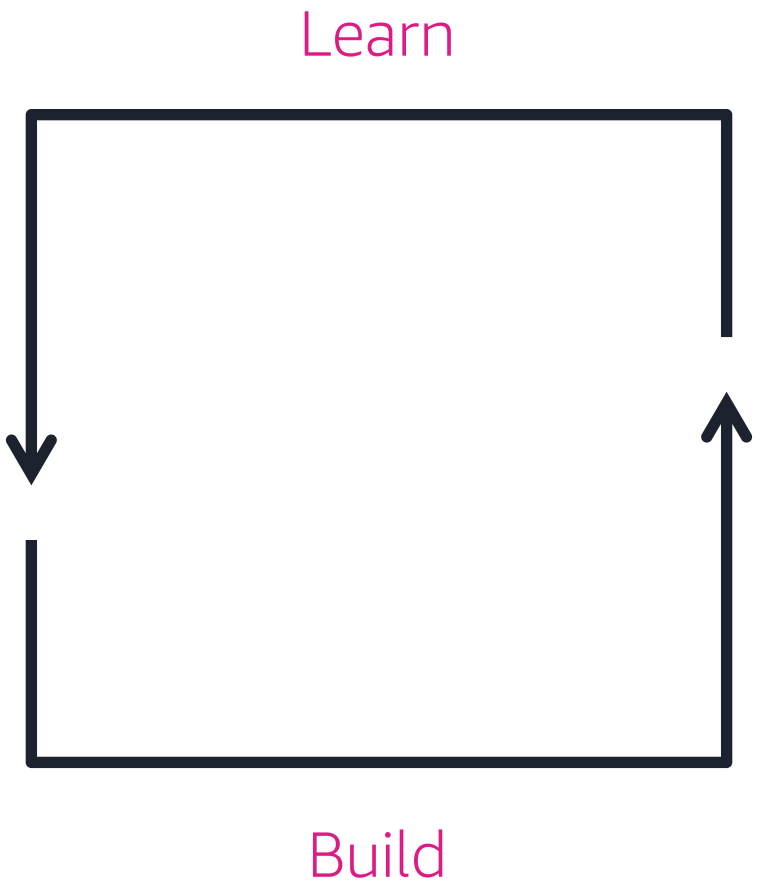


Experiment



Prototype

The Learn-Build-Learn Cycle



Where are you in your MVP journey today?

aws startups

Before You Build

aws startups

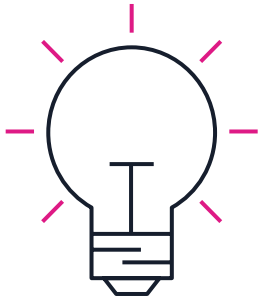
Are You Ready To Build an MVP?

Problem Definition

A clear understanding of the problem you are solving will ease communication of your idea with both stakeholders and users.

Problem Metrics

The number of people that have this problem, and the impact of the problem to those users.



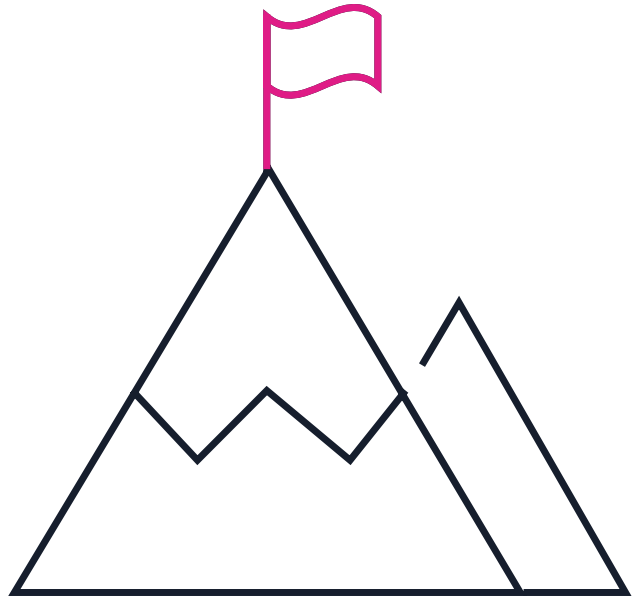
Solution Hypothesis

Initial hypothesis of how the problem will be solved for your users.

aws startups

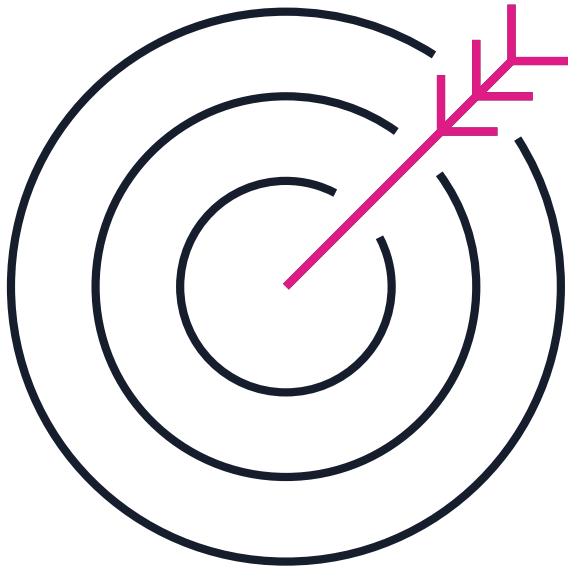
MVP Is a State of Mind

- Process not product
- Speed is paramount
- Set the right focus (10-100 customers)
- Customer feedback is critical
- Expect to change/pivot/adapt
- Weigh technical debt vs other kinds of debt



MVP Feature Prioritization & Scope

- Focus on building and validating the strongest use case first
- In most cases any building should take < 3 person months
- Distinguish between being busy and creating value
- Minimize friction wherever possible for your users



aws startups

“Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away.”

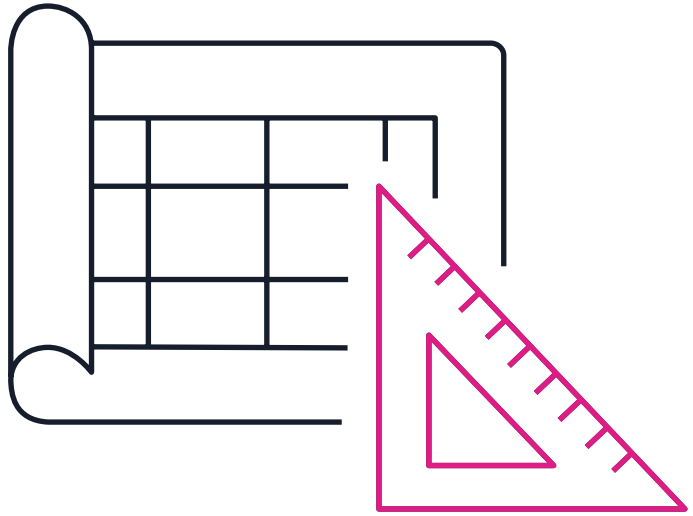
Antoine de Saint-Exupéry
Airman's Odyssey

Leverage Frameworks

What Architecture Is Right For Me?

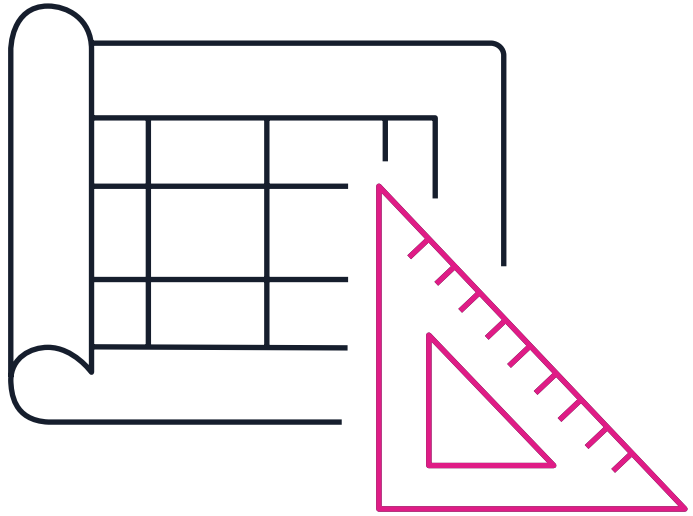
Ask yourself some important questions:

- What are you building?
- What are you optimizing for?
- What are your existing skill sets? What skill sets are available to you in the market?

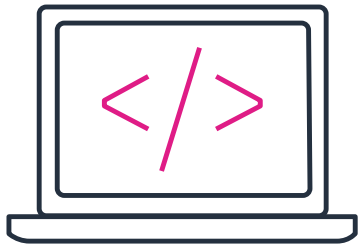


What Architecture Is Right For Me?

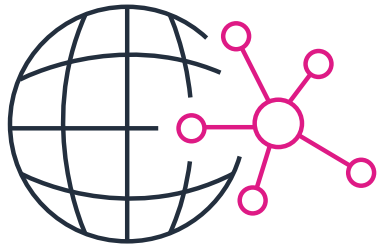
- The monolith is a viable option
- Leverage frameworks to increase your speed and confidence
- Very little of what you build in your MVP will be recognizable a year later
- Whatever you choose: leverage infrastructure as code!



The Code Continuum



Consider where
no code, and low code
solutions fit in

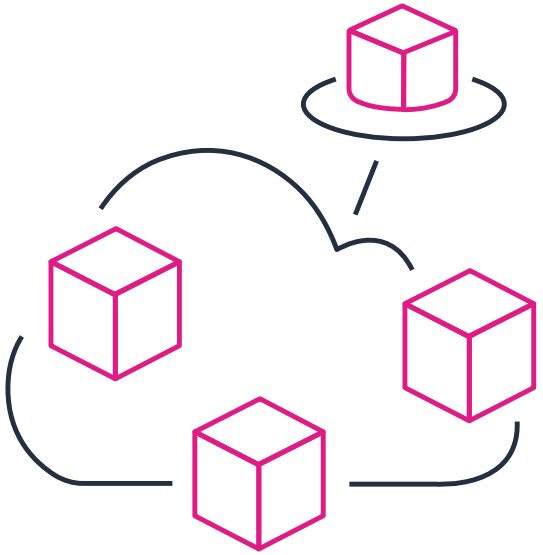


Abstraction
vs
flexibility tradeoffs

AWS Amplify

AWS Amplify is a set of tools and services that can be used together or on their own, to help front-end web and mobile developers build scalable full stack applications, powered by AWS.

- Configure and deploy backends quickly
- Seamlessly connect frontends
- Streamlined deployment of frontends
- Easily manage your users and content



aws startups

AWS Copilot

AWS Copilot is a command line interface (CLI) that enables customers to quickly launch and easily manage containerized applications on AWS.

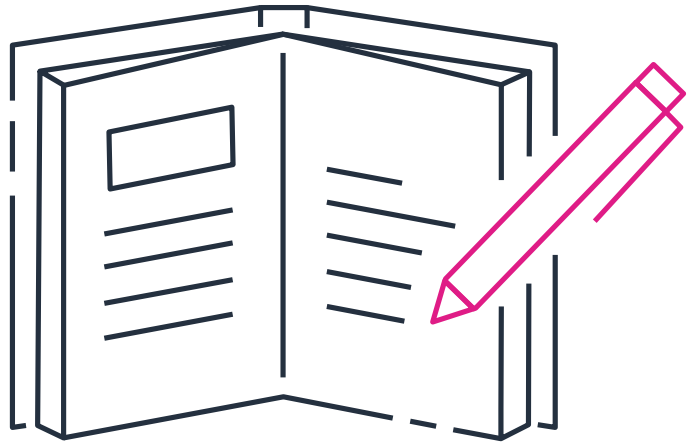
- Architecture, not infrastructure
- Simple and powerful config
- Develop/Release/Operate

aws startups

AWS Solutions Library

The AWS Solutions Library offers a collection of cloud-based solutions for dozens of technical and business problems, vetted for you by AWS.

- Automatically deploys directly into your AWS account
- Library of AWS-vetted architecture diagrams



aws startups

AWS Cloud Development Kit

The AWS Cloud Development Kit (AWS CDK) is an open source software development framework to define your cloud application resources using familiar programming languages.

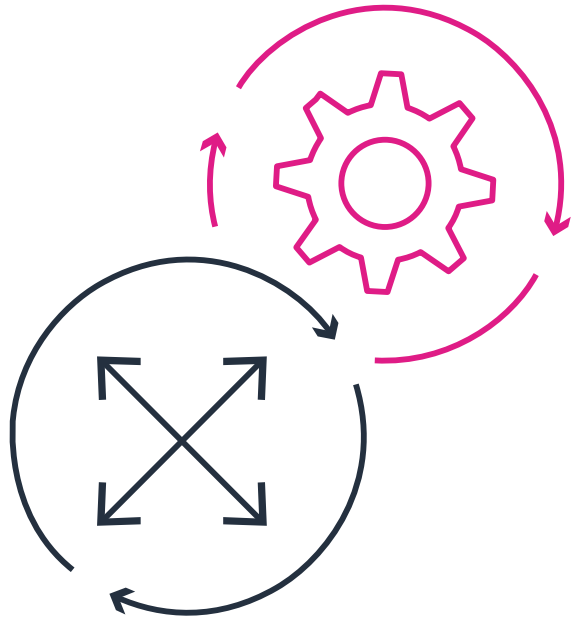
- Easier cloud onboarding
- Faster development process
- Customizable and shareable
- No context switching

aws startups

AWS Solutions Constructs

AWS Solutions Constructs are vetted architecture patterns, available as an open-source extension of the AWS Cloud Development Kit, that can be easily assembled to create a production-ready workload.

- Infrastructure-as-code
- Pre-built, well-architected multi-service patterns
- Combine to create your own solutions
- Speed up your development cycle
- Consistently deliver Well-Architected apps



For those of you that are building today, what frameworks are you using, if any?

aws startups

Flexible Architecture

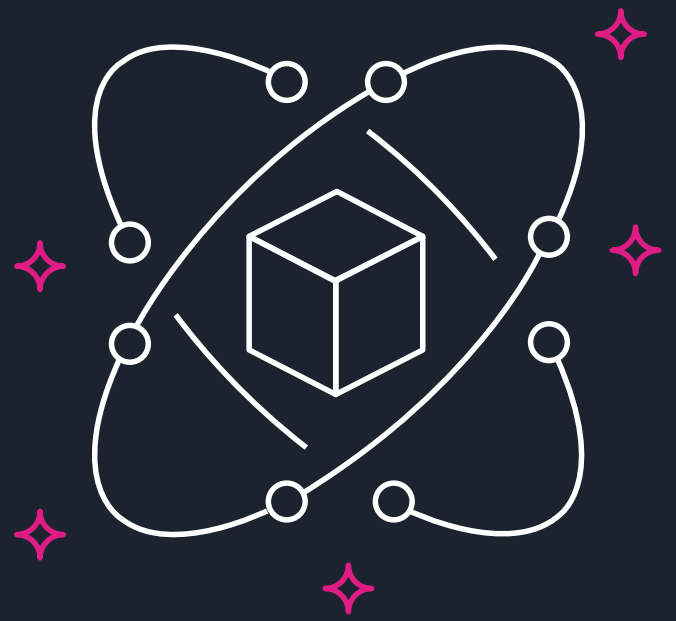
aws startups

Technology/Architecture Philosophy

- Avoid “undifferentiated heavy lifting” at the beginning
- Resist the temptation to add complexity unnecessarily
- Don’t introduce endogenous churn
- De-risk by transferring that risk to your provider
- Don’t be intimidated by the unknown

Prioritize Managed Services

- Choose the highest level of abstraction available to you that gets the job done
- You can always choose to de-abstract down the road, the reverse is much harder
- Non-functional requirements are baked in to managed services
- Operations burden is significantly reduced
- Security risks are significantly (but not completely) reduced



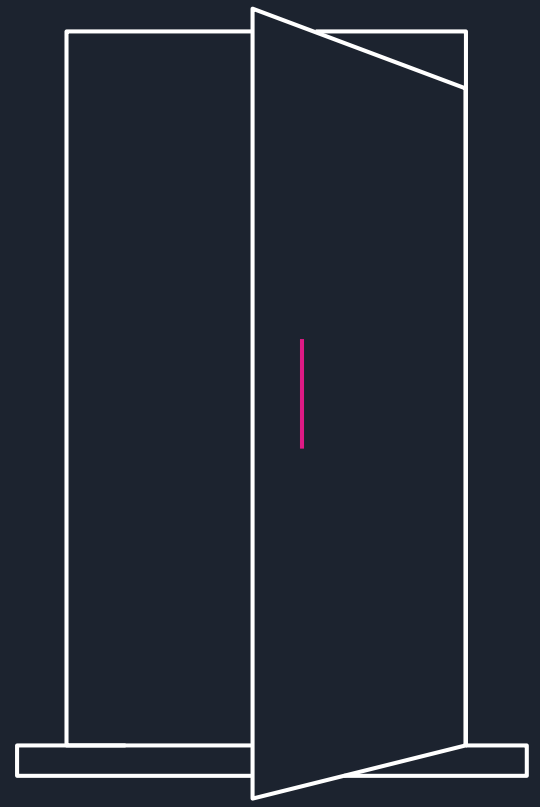
Minimize Throwaway Work

- Minimize, but don’t expect to eliminate re-work
- PMF can be described as a point where your product is breaking under growth
- Let your user feedback and desired outcomes drive changes, not technical debt
- The iterative learning cycle can mean evolution over revolution

aws startups

Make Reversible Decisions Quickly

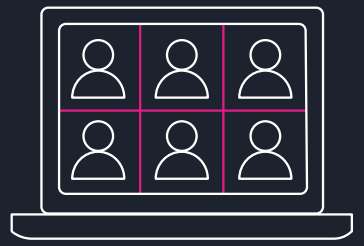
- At Amazon we call these decisions “two way doors”
- What is the cost/difficulty of reversing the decision?
- How do I measure the outcome?
- How do revert the decision?
- Be prepared to be wrong a lot initially



Externalize State

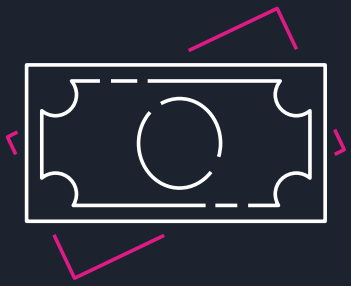
- Separating state from logic gives you optionality, scaling, and flexibility
- It also introduces new challenges and risks you need to manage

Externalize Authentication



User management is the pinnacle of “undifferentiated heavy lifting”

One of the highest risk components in the architecture



Payments is another prime example of this

For those of you that are building today, what frameworks are you using, if any?

aws startups

Building Guardrails

aws startups

Build Guardrails, Not Gates

- Includes but not limited to both security and operations
- Teams should spend time solving problems that move the business ahead
- Experimentation should be encouraged
- Seeing the results of a change should be immediate
- Time spent firefighting, waiting on people or processes, or doing rework is waste
- Hiring talent is hard enough

Onboarding

- As you grow your team the ease and safety of onboarding new developers is critical
- You will have varying levels of trust
- A single source of truth for identity will make your life much easier
- Start documenting expected standards and processes early

Onboarding

Consider the differences in:

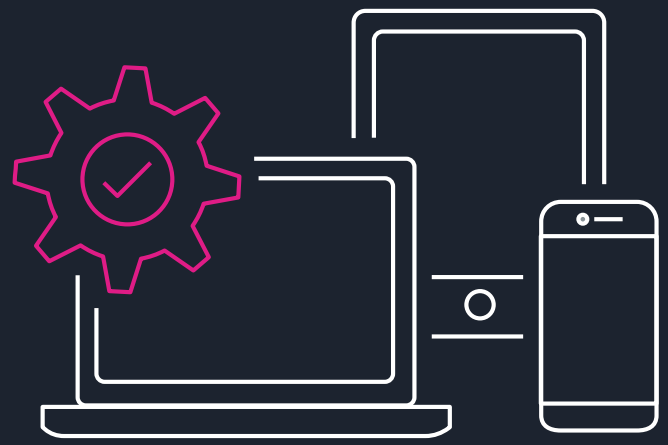
the ability to push code

access production systems

access production data

CI/CD

- Leverage the benefits of small, fast units of deployment
- If you aren't shipping those updates, you are missing the primary benefit
- Working on an MVP without CI/CD is like trying to "roll a cube"
- How much CI/CD is right initially?
- Does anyone really write tests?

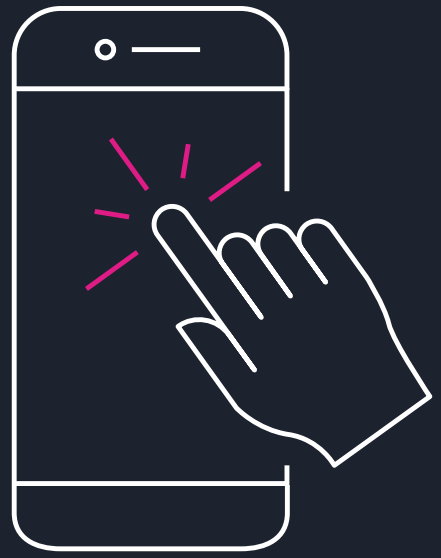


Observability

- Observability is what brings confidence to move quickly
- How do you know the impact of your changes?
- When do you need to rollback quickly?
- Key components:
  - Metrics
  - Logging
  - Tracing

Synthetics

- You want to find issues **before** your customers/users do
- Continually verify your customer experience (even when you don't have any customer traffic)



What level of confidence do you have in quickly making changes to your application today?

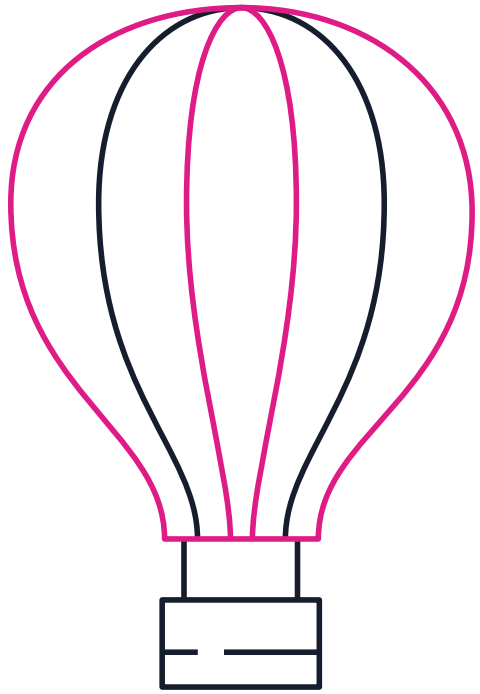
aws startups

Surviving The Unexpected

aws startups

Every Once In Awhile, Look Up

- It’s easy to get tunnel vision when focused on the product
- There are important ways to minimize existential risk in your environment
- Don’t wait until the loss would be catastrophic
- **Minimize, not eliminate**



Managed Backups

- Managed services help guard against failure
- They don’t protect you from accidents, errors, and malice
- Centralize and automate data protection
- Covers storage volumes, databases, and filesystems
- Validate and test the process before you need it

Cost Visibility

- Ensuring you have visibility into your costs is critical to planning process
- Understanding your costs helps you reason about feature viability
- Unexpected spikes are indicators of a problem that you want to be aware of quickly
- Track your AWS credit usage
- Use budget alarms

aws startups

How confident are you in understanding your cloud costs, and avoiding unexpected surprises?

aws startups

Understanding Your Users

aws startups

Validated Learning Drives MVP Iteration

- Learning what resonates with your users is a core purpose of your MVP
- Determine how users are actually using your MVP
- The only users that have value are active users
- User metrics are different than operational metrics
- Investors will expect you to be an expert on your users and their experience with your solution

Measuring User Activity

- What are unexpected high use features?
- Where do you see unexpected drop off/abandonment?
- How do you need to update your hypothesis?
- If you can log it, you can build metrics around it.

Integrating Offline Tasks

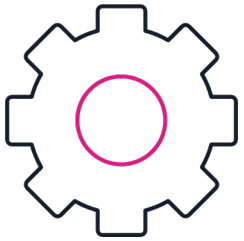
Fake It Until You Make It

- Your MVP make look different to your users than from the inside
- Manual processes are normal and to be expected
- You can still track them to ensure good customer experience
- Lay the groundwork for future automation

Tracking Manual Work with Step Functions

- You can leverage workflow co-ordination between systems
- You can loop in humans with yes/no decisions via email
- Applies to any back office process

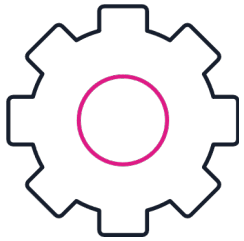
Key Takeaways



Speed

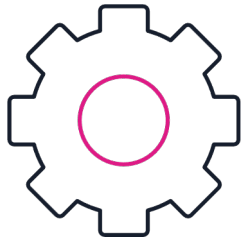
Forgo perfection and make progress.

“Move fast and make things!"



Learning

Every day should generate new questions, based on yesterday’s activity.



Risk Reduction

Have awareness of, and externalize or mitigate risk where possible.

"Make something people want."

Paul Graham
YC

Join Us August 11!

Get it Right:

10 Mistakes to Avoid When Starting on AWS

Register Here!

aws startups

aws startups

Thank you

Mike Apted
Principal SA, Startups

@mikeapted

ln/mikeapted