

Vercel-like CLI Tool

Effortless Deployment to AWS

This presentation highlights an innovative CLI tool

Aimed at streamlining the frontend deployment process to AWS, making it easier for developers to manage their applications efficiently.

Priyesh Rai



Why Is Frontend Deployment Still Hard?

Exploring the challenges of deploying frontend applications on AWS efficiently



Meet deploy-tool CLI — Your Ultimate AWS Deployment Buddy

Simplify your deployment process with just a few commands and no AWS Console needed.

CLI tool for seamless deployment.



This tool auto-builds, auto-deploys, and auto-monitors your applications effortlessly.

Simple commands for quick setup.



Use just a few commands: `init` → `config` → `deploy` → `status` → `rollback`.

Compatible with React, Node.js, static & other websites.



Works perfectly with popular frameworks, making it versatile for developers.

No need to use the AWS Console at all.



Focus on your code instead of navigating complex interfaces.

5 command to rule your deployment.



Experience the simplicity of managing your deployments with just: 5 commands.

Old Way vs New Way: How It Makes Life Easier

Transforming manual processes into seamless, automated infra deployment with ease.



Old Way

- Manual S3/ECS setup
- CloudWatch monitoring is complex
- Rollbacks are panic-driven
- Different configs per environment
- Console clicking and guesswork



New Way with CLI

- One-line infra automation
- Grafana monitoring preconfigured
- Safe rollback in 1 command
- Unified config files
- Stress-free deployment

Who Can Benefit from This Tool?

Discover the diverse users who can leverage this powerful deployment tool for their projects

Interns and beginners

This tool is perfect for **newcomers** looking to gain essential deployment experience and skills quickly.



Teams needing agility

Quick project setup and deploy — under 10 minutes

No time wasted configuring AWS manually

Supports rapid rollbacks and versioning in case of last-minute bugs.

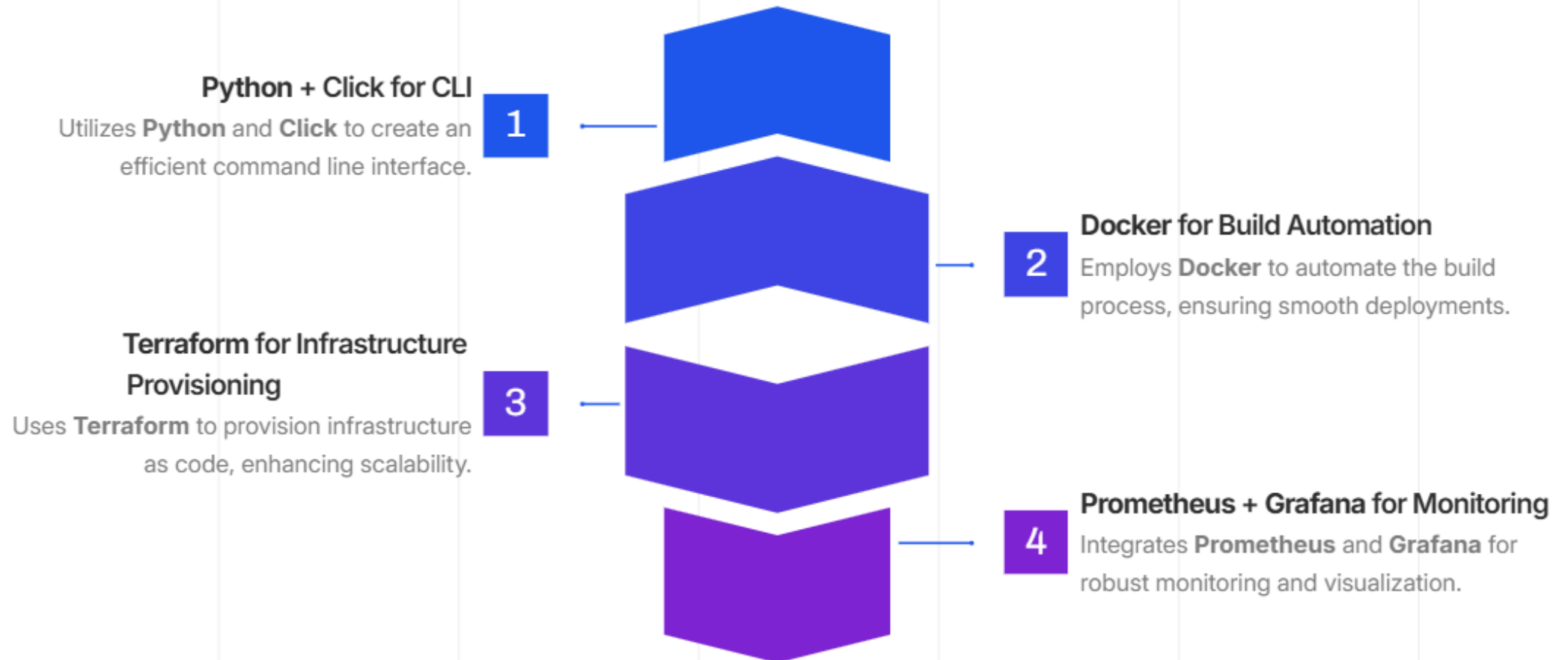
No time wasted configuring AWS manually or fixing CI pipelines.

Eg:- Hackathons



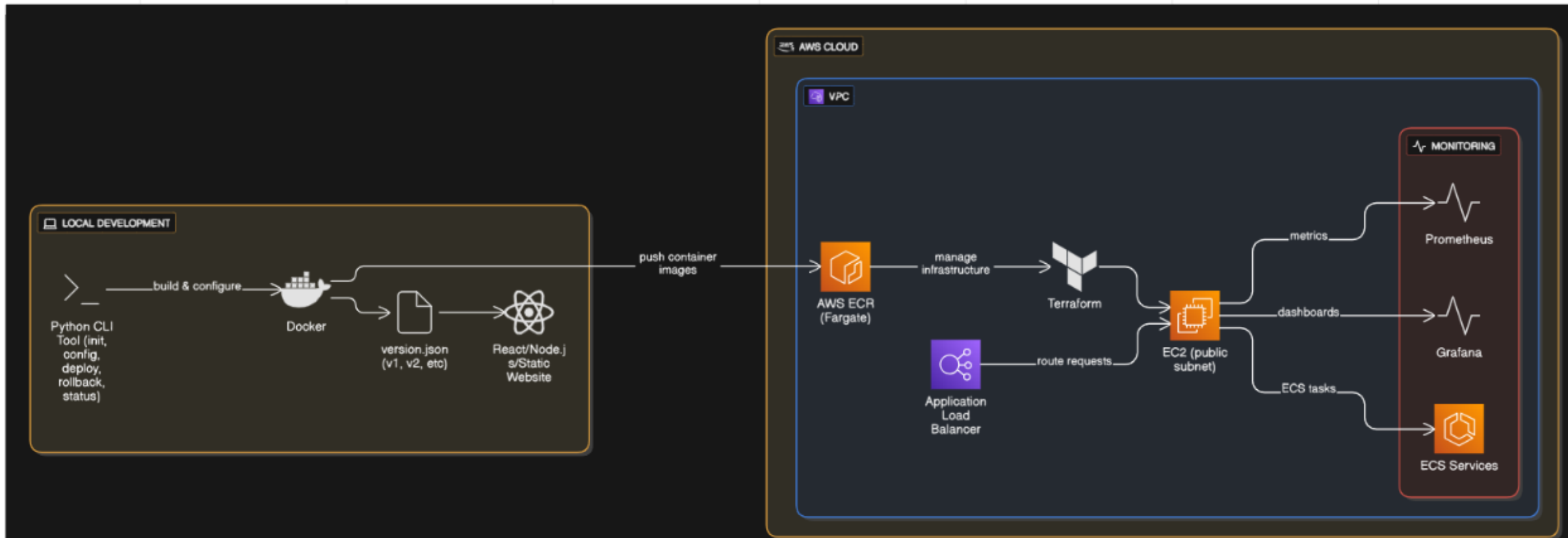
Powering the Vercel-like CLI Tool with Modern Tech

Discover the technologies that drive our innovative frontend deployment solution on AWS.



Understanding Project Architecture Diagrams

A comprehensive overview of project architecture and its key components



Overview of Current CLI Features for Deployment

Explore the automated tools for seamless app deployment and management

Command	Description
deploy-tool init	Automatically detects what kind of frontend app you're using (React, Next.js, or static) so you don't have to configure it manually.
deploy-tool config	Lets you choose whether you're deploying to a development or production environment. It sets up everything behind the scenes without you writing any configuration.
deploy-tool deploy	With just one command, your app gets packaged, uploaded, and deployed to the cloud. After it's done, you instantly get a public URL to view your live app.
deploy-tool rollback --version vX	If a new version breaks something, you can quickly go back to a previous working version of your app — no need to troubleshoot or rebuild manually.
deploy-tool status	Helps you check if your app is running properly. It also gives you a link to the monitoring dashboard (Grafana) to view performance and uptime.

Technical Overview of the CLI Tool: deploy-tool

Explore the main features and commands of the deploy-tool CLI for efficient deployment

Command	Function
<code>deploy-tool init</code>	Detects frontend framework (React/Next.js/static), generates <code>.deployconfig.json</code>
<code>deploy-tool config</code>	Prompts for environment (dev/prod), creates <code>.awsconfig.json</code>
<code>deploy-tool deploy --version vX</code>	Builds Docker image, pushes to ECR, deploys to ECS Fargate via Terraform
<code>deploy-tool rollback --version vX</code>	Restores an older ECS task definition using version history (<code>version.json</code>)
<code>deploy-tool status</code>	Shows ECS service health, returns ALB URL & Grafana dashboard link
<code>deploy-tool populate-efs</code>	Uploads monitoring configs to EFS using EC2 instance + SSM

Achievements in Deployment and Automation

Fast, efficient, and automated deployment solutions for modern applications

1. Lightning-fast deployments in under 7 minutes

Experience fully operational deployments in less than five minutes, drastically reducing time to market and enhancing productivity for your team.

2. Automated monitoring for seamless oversight

Benefit from our auto-provisioned monitoring system that ensures your applications run smoothly and performance is constantly evaluated without manual intervention.

3. Rollback safety for real-world applications

Our system incorporates robust rollback safety features, allowing you to revert changes quickly and safely in case of unexpected issues during deployment.

4. Integrated Terraform, Docker, and ECS automation

Utilizing a Python CLI, we seamlessly automate Terraform, Docker, ECR, and ECS, simplifying the whole deployment process for enhanced scalability and speed.

Engaging Q&A Session: Let's Connect

Your questions and feedback are invaluable to us, let's discuss

1 Thank you for your time and attention

We truly appreciate your participation and interest in our presentation today.

2 We welcome your feedback

Your insights are crucial for us to improve and tailor our offerings to meet your needs.

3 Ready for a live demo if needed

If you're interested, we can showcase a live demonstration of our tool's capabilities.

4 Questions? Let's have an open discussion

We're here to clarify any doubts and engage in meaningful conversations with you.

5 Looking forward to your thoughts

We value your opinions and are eager to hear your thoughts on our presentation.

6 Connect with us on social media

Stay updated and connect with our community through various social media platforms.