

What are the problems Parts Unlimited faces that are indicative of a lack of DevOps culture?

Parts Unlimited faces many problems that clearly point to a lack of DevOps culture. Working across the organization is hindered by bottlenecks and prolonged waiting, where developers and QA frequently spend days or weeks without the necessary credentials, environments, or approvals to be productive. The processes are highly bureaucratic, requiring multiple tickets, managerial approvals, and handoffs between teams, many of which result in tickets being closed without real progress. Development, QA, and Operations are stuck in silos, often blaming each other instead of collaborating to solve shared problems. Developers have no fast feedback loops, and instead of building and testing code daily, integration happens late, creating massive risks. Leadership also contributes to the dysfunction by focusing on features rather than fixing the broken build and environment processes, leaving the teams frustrated and unproductive. Finally, executives remain disconnected from the day-to-day reality of engineers, pushing deadlines without regard to technical feasibility. These issues together highlight a culture without collaboration, shared responsibility, or the fast feedback that defines DevOps.

What DevOps process and behaviors that Parts Unlimited should implement?

To address these challenges, Parts Unlimited should adopt DevOps practices focused on automation, collaboration, and fast feedback. Automating environment setup with infrastructure-as-code lets developers start immediately rather than waiting weeks. CI/CD pipelines ensure code is built, integrated, and tested frequently, reducing failure risks. Cross-functional teams from Dev, QA, Ops, and Security promote collaboration by breaking silos. Short feedback loops through automated testing and metrics help teams identify and fix problems early. Reducing bureaucracy in ticketing and empowering teams encourage a culture of action and shared responsibility.

List the major pain points that Maxine experienced while onboarding to the Phoenix project. What are the contributing factors to their existence?

Maxine's onboarding experience with the Phoenix Project exposed several issues that highlight the lack of DevOps practices. She faced challenges with missing documentation, inconsistent build scripts, and struggled to get the entire system operational. Access to credentials, licence keys, and environments was slow and reliant on a ticketing system that bounced requests between people and often closed them too soon. Despite days of effort, she still lacked a working build environment, forcing her to rely on multiple teams across storage, networking, and security to make progress. The organization's resource shortages, such as being out of storage for months, further underscored a lack of planning and prioritization for essential development needs. Culturally, she experienced indifference, with some colleagues dismissing her concerns and others telling her to "stay in her lane" instead of tackling systemic issues.

Describe a possible DevOps behavior or practice that would improve this process.

DevOps practices can significantly improve onboarding for new developers like Maxine. Automated, well-documented build processes allow quick setup without searching for dependencies and configs. Self-service environments enable developers to create their own systems on demand, avoiding tickets and approvals. Using modern tools and containerization ensures system consistency and reliability. Fostering shared

responsibility prevents developers from feeling powerless. Integrating fast feedback loops through automated testing and continuous integration offers immediate insights into code performance. These practices make onboarding smoother and promote a DevOps culture of collaboration, automation, and continuous improvement over bureaucracy.