Obstacles to Modernization

Research Report: Tech Leaders on Why So Many App Modernization Projects Fail

About this Research

In partnership with Wakefield Research, we surveyed 250 tech leaders who are responsible for maintaining a monolithic app at a large company (5,000+ employees). As code bases grow, complexity increases and engineering velocity slows down. With many digital transformation initiatives and the continued drive towards the cloud increasing the obvious need for app modernization, 92% of respondents say they are planning to start or have already started such projects. Here’s what we learned about what makes them successful or not.

State of Modernization Across the Industry

<table>
<thead>
<tr>
<th>State of Modernization</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Modernization</td>
<td>8%</td>
</tr>
<tr>
<td>Modernization</td>
<td>92%</td>
</tr>
</tbody>
</table>

- 8% don’t plan on modernizing at all
- 28% plan to modernize, but haven’t started yet
- 16% have just started to modernize
- 34% have made moderate progress
- 14% have made significant progress
- 16% have made major progress

89% predict someone in their organization would push back on a proposed project

Modernization Majority—High Cost of Failure

<table>
<thead>
<tr>
<th>App Modernization Efforts Fail</th>
<th>Average Cost of Modernization Project</th>
<th>Average Time of Modernization Project</th>
</tr>
</thead>
<tbody>
<tr>
<td>16%</td>
<td>$1.5m</td>
<td>16 mo</td>
</tr>
</tbody>
</table>

Organizational pushback can often hamstring projects before they start. When asked what has stopped modernization efforts, both executives and architects say a lack of “prioritization from management.” 97% predict someone in their organization would push back on a proposed project.

Top Obstacles to Modernization Projects

- Cost
- Risk
- Complexity

Divergence in Thought for Executives and Architects

<table>
<thead>
<tr>
<th>GOALS</th>
<th>CHALLENGES</th>
<th>REASONS FOR FAILURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowering Technical Debt</td>
<td>Time Developing with Business Requirements</td>
<td>Expectation</td>
</tr>
<tr>
<td>Keeping Up with Business Requirements</td>
<td>Difficulty in Refactoring and Training Developers</td>
<td>Lack of Intelligent Tools</td>
</tr>
<tr>
<td>For New Developers</td>
<td>Difficulty in Refactoring</td>
<td>Lack of Intelligent Tools</td>
</tr>
<tr>
<td>Difficulty in Refactoring</td>
<td>Difficulty in Training Developers</td>
<td>Lack of Intelligent Tools</td>
</tr>
</tbody>
</table>

- Executive Aligns for the Team
  - Teams need buy-in for what skills are in demand and what skills are in short supply. To align the team, there needs to be a clear vision for what skills are in demand and what skills are in short supply.
  - Teams need buy-in for what technologies are important and what technologies are not.
  - Teams need buy-in for what tools are needed and what tools are not.
  - Teams need buy-in for what processes are in place and what processes are not.

- Create/Combine Team
  - Teams need buy-in for what teams are in place and what teams are not.
  - Teams need buy-in for what stakeholders are in place and what stakeholders are not.
  - Teams need buy-in for what resources are in place and what resources are not.

Elements of a Successful Plan

1. Make the Case
   - First analyze the portfolio for architectural complexity, technical debt and identify aging frameworks. From here, prioritizations can be made, and then scoping the time, budget and team members needed can be aligned.

2. Secure Budget & Resources
   - Armed with the right business case, securing budget and resources may be easier, especially when tied to strategic business outcomes.

3. Give It Consistent Support
   - Executive management supporting the project through the changing tides of business cycles is key to its success. When commitment from leadership falters, projects can stall, stop or end.

4. Define, Align & Train the Team
   - Look across your team to determine what skills are in house already, and what skills are needed to add to the team. Next, gather the stakeholders and put the right people in the right places. Finally, organize your team around the microservice—a critical element of successful projects.

5. Provide Your Architects with Intelligent Tools
   - Architects noted that not having the right tools was the top reason for failure. Giving them intelligent tools will help reduce time and risk.

About vFunction

vFunction is the first and only AI-driven platform for architects and developers and architects that intelligently and automatically transforms complex monolithic Java applications into microservices, restoring engineering velocity and optimizing the benefits of the cloud. Designed to eliminate the time, risk and cost constraints of manually modernizing business applications, vFunction delivers a scalable, repeatable factory model purpose-built for cloud native modernization. With vFunction, leading companies around the world are accelerating the journey to cloud-native architecture and gaining a competitive edge. vFunction is headquartered in Palo Alto, CA, with offices in Israel. To learn more, visit vFunction.com.

Request a Demo