Why Do Software Projects Need

QA Testing?

OnPath

A Stakeholder's Guide

"It costs about six times more to fix a bug found during implementation than during design. Moreover, the cost to fix bugs found during the final testing phase could be 15 times more than the cost of fixing those identified during design."

Source: The Systems
Sciences Institute at IBM

OnPath

Every project has unique challenges, but there are common pitfalls that can bring down even the most well-planned initiatives.

What follows are the significant contributions of independent QA testing on project success.

How Software Projects Typically Go…

1. Spend $$ on hiring a project/product manager and building a GREAT developer team.

2. Get underway, let developers fix bugs as they become apparent.

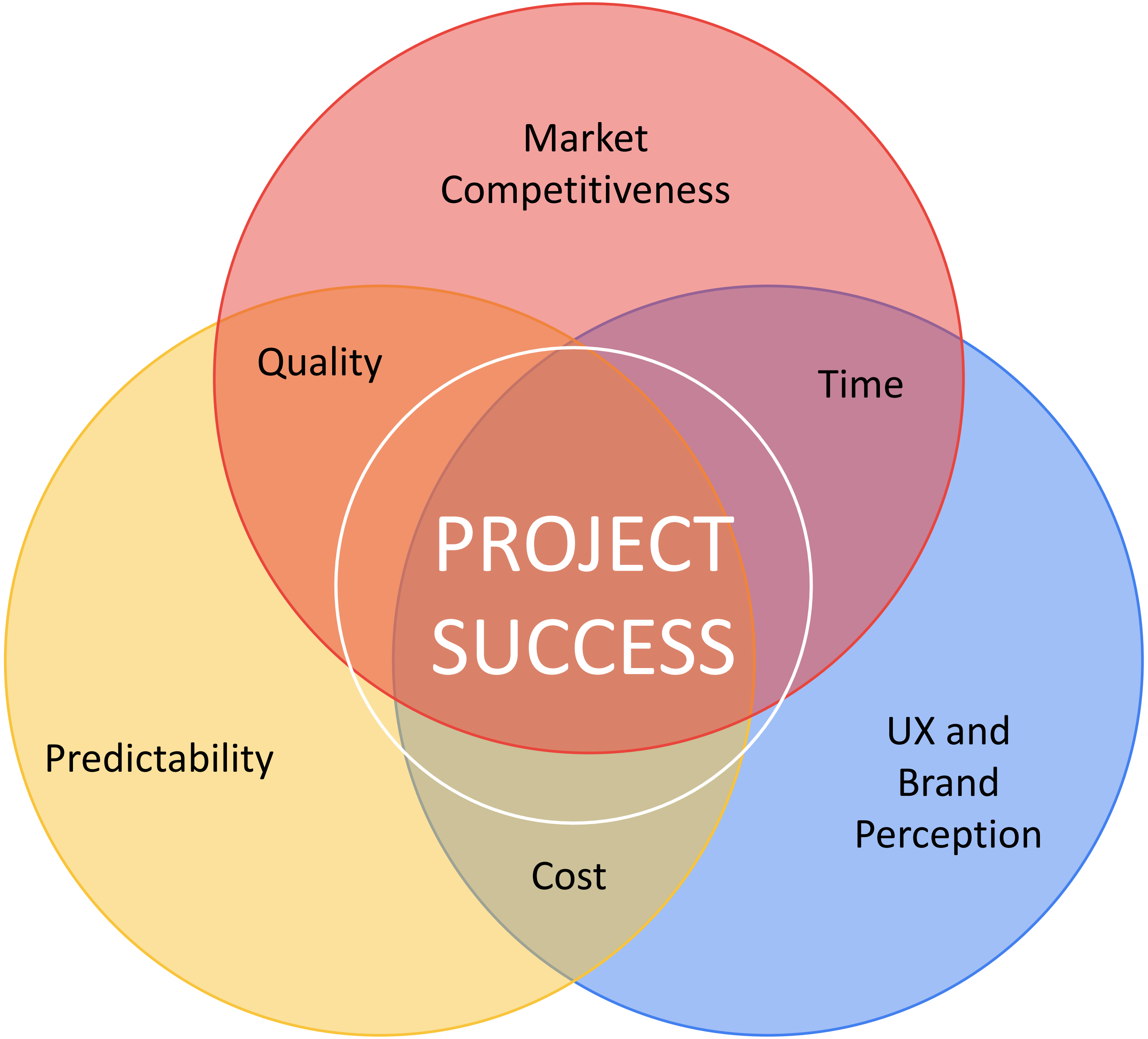
3. Spend as little time and money as possible cleaning up for a tight release date.

Consequences…

- Unplanned costs
- Project delays, stalls, bottlenecks
- Poor code
- Team communication breakdown, burnout
- Dissatisfied end users, high churn
- Damaged brand perception
- Poor marketplace performance

QA Testing:

Six Critical
Contributions
to Project Success



OnPath

Effectiveness

Quality

- QA testing brings a product into alignment with specified requirements and standards.
- Quality-focused cultures foster a collaborative environment, contributing to the overall improvement of the end product.
- This extrapolates to higher product adoption rates.

Predictability

- QA testing contributes to a predictable, consistent development environment.
- This means better project planning and resource allocation, and on-time, on-budget delivery.
- Predictability contributes to team productivity and happiness, resulting in healthy SDLC cultures with minimal stress and burnout.

Impact

UX / Brand Perception

- Poor user experience directly impacts consumer perception of a brand. News of dissatisfaction spreads fast via online reviews and social media.

- Glitchy product launches create negative brand perception. This leads to costly brand repair.

- Poor UX means high churn rates.

Market Competitiveness

- Early and ongoing QA testing investments produces products that rise above competition in terms of:

  1. Quality
  2. Reliability
  3. User Experience

- Falling behind competitors leads to loss of market share.

Resources

Time

- Failure to identify critical bugs can derail a release timeline, impacting budgets.
- Introducing testing early in the SDLC prevents compounding errors, which escalate complexity, cost and time.
- Integrating QA testing into development helps maintain schedules and budgets within scope, fostering better outcomes.

Money

It costs six times more to fix a bug found during implementation than during design. Moreover, the cost to fix bugs found during the final testing phase could be 15 times higher (or more) than the cost of fixing those identified during design.

Source: The Systems Sciences Institute at IBM

Onboarding QA to a project as early as possible helps reduce expensive later-phase fixes.

OnPath

Problem

1. Unexpected cost increases
2. Schedule delays
3. Unpredictability: surprises and unforeseen obstacles
4. Poor code quality
5. Dissatisfied end users, high churn
6. Damaged brand perception
7. Poor marketplace performance

Solution

1. By onboarding QA testing in design and development stages, bugs can be addressed at a fraction of the cost of post-release intervention.
2. QA testing helps minimize delays caused by developer stalls and bottlenecks.
3. QA testing contributes to a predictable, consistent development environment, reducing stress and burnout and improving productivity.
4. QA testing brings a product into alignment with specified requirements and standards, resulting in a higher quality product.
5. Quality-centric standards deliver products consumers adapt and fosters loyalty.
6. Good user outcomes lead to positive brand perception.
7. A brand and product with positive perception have a competitive advantage in the marketplace.

Why Can’t Developers Test Their Own Work?

Experience

- Front end, black box QA test engineering and software engineering are different disciplines with different practical experience.
- QA engineers begin with an understanding of bugs and good usability. A developer does not think in those terms.
- Typical functional testing is not an ideal use of a developer’s time.
- Developers are thinking about how to create an application, but testers are challenging the design to find flaws so that end users don’t.

Why Hire **OnPath?**

By partnering with OnPath Testing, clients gain access to expert knowledge, cutting-edge tools, and a team of dedicated professionals working tirelessly to ensure software projects meet the highest standards of quality.

- Our flexible testing schedules are designed to accommodate project timelines.
- **OnPath's** testing solutions are designed to scale up or down as needed, ensuring that resources are allocated effectively.
- We seamlessly integrate with teams with minimal disruption or interference.

Want To Learn More?

- The Hidden Costs of Skipping Software Testing in Development
- The Incredible Rate of Diminishing Returns of Fixing Software Bugs
- The Latest Insights into the Costs of Bad Code
- The Cost of Poor Software Quality in the US: A 2020 Report
- Lower Total Cost in Independent Software QA Testing
- How to Get Accurate Software Testing Cost Estimation?
- Real Money: Poor Software Testing Practices Cost US Companies $59 Billion
- The exponential cost of fixing bugs