

# Is Your Retail App Ready for Black Friday Traffic?

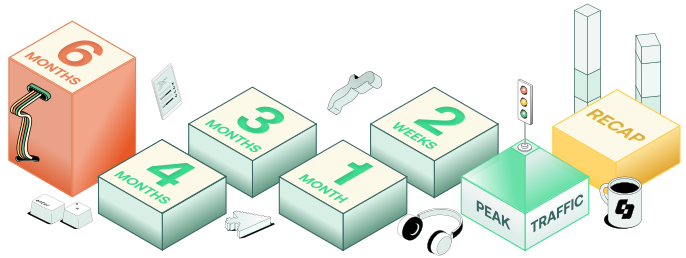
Your Retail App Holiday Readiness Timeline

The holiday shopping season, particularly **Black Friday** and **Cyber Monday**, brings both **immense opportunity** and **significant risk** for retail businesses.

There is zero margin for error. Preparing for these traffic spikes is nearly a year-round task for most commercial retail apps and websites. This workbook plan provides a timeline for your team to follow as you prepare for your retail apps' peak traffic periods.



## Milestone 1: 6 Months Out



### Strategic Planning

At the six-month mark, your focus should be on developing a comprehensive plan that addresses all aspects of holiday readiness. There are many tradeoffs to consider, though. How much infrastructure should the team stand up and at what cost? The more you spend on additional infrastructure, the more your margins are squeezed. **What is the right balance of margin and risk tolerance?** These conversations go beyond an engineering meeting; they are company-wide conversations.

#### 1 Establish traffic targets based on historical data

- Review and analyze last year's Black Friday/Cyber Monday traffic patterns
- Consider year-over-year growth projections
- Aim to test for 5x your anticipated peak traffic

#### 2 Document regional traffic patterns

- Identify which geographic regions your traffic comes from
- Map your browser and device mix (mobile vs. desktop)
- Consider regional holiday variations if you serve international markets

#### 3 Executive alignment

- Create a concise (2-3 page) plan document that describes the team's strategy for peak traffic preparation
- Gather all stakeholders for review and sign-off
- Ensure system architects and platform engineers are involved
- Secure executive sponsorship for your strategy

#### 4 Infrastructure scaling strategy

- Design your approach to scaling up and down infrastructure
- Balance cost management with capacity needs
- Plan for regional failover (e.g., between AWS regions)

### Risk Management

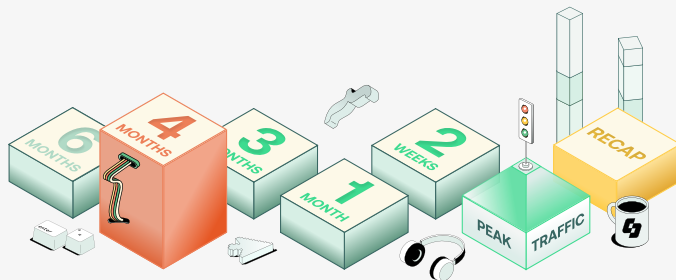
#### 1 Identify any system changes since last year

- New components (analytics platforms, payment systems, etc.)
- Removed components
- Modified infrastructure

#### 2 Third-party dependency review

- Map all external service dependencies
- Determine which can be made asynchronous
- Identify critical vs. non-critical third-party services

## Milestone 2: 4 Months Out



### Disaster Recovery Testing

This is the time to imagine the worst-case scenario and what your team would do in this situation. While it's unlikely for an entire app to go down, degraded services and experience can have severe effects on revenue. As you prepare, plan a chaos testing blitz to simulate various failure scenarios:

#### 1 Tiered recovery scenarios

- Site degradation response
- Complete outage recovery
- Data loss recovery

#### 2 Regional failover testing

- Simulate failure of a primary data center
- Ensure traffic can shift to backup regions
- Measure performance impact during regional failover

#### 3 Backup validation

- Verify data backup procedures
- Practice restoration processes
- Document and time the recovery process

### Prepare a Static Site

While a static site won't be able to provide inventory updates and other key details, it will continue to allow your users to purchase items from your website or app.

#### 1 Create a bare-minimum static version of your site

- Focus on revenue-critical paths only
- Minimize dynamic content
- Include monetization even on error pages (404, 500)

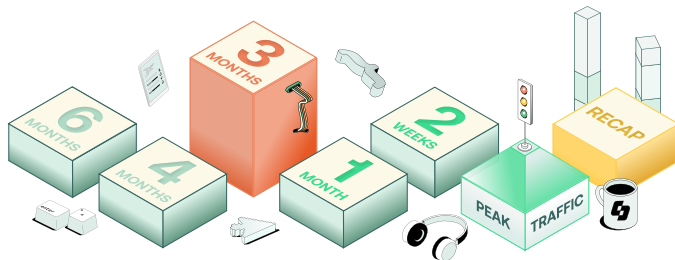
#### 2 Confirm vital elements with marketing teams

- Determine which content must be included in the static version
- Create processes for updating static content when needed

#### 3 Test static site performance

- Ensure the static site can handle extreme load
- Validate that revenue paths function correctly
- Test the process for switching to the static site

## Milestone 3: 3 Months Out



### Code Management and Feature Planning

Restricting the deployment of new features will help you minimize risk when introducing a new bug into production. This doesn't mean that development teams stop innovating and building during this time—just that new features don't get deployed until after the peak holiday season.

#### 1 Begin restricting new feature development

Categorize features by risk level:

**LOW RISK:**

New content on  
existing pages

**MEDIUM RISK:**

New analytics  
events

**HIGH RISK:**

New payment  
methods and  
authentication  
systems

#### 2 Implement a feature flag for deployments

Ensure all new features can be  
toggled off independently

Test the disabling/enabling process

Document decision-making  
protocols for feature disablement

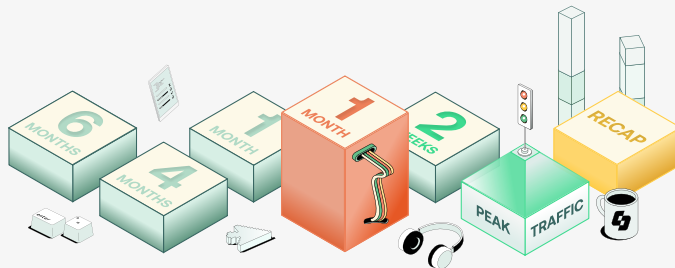
#### 3 Performance optimization

Implement and validate code  
minification

Review and optimize image delivery

Reduce payload sizes across the  
application

## Milestone 3: 1 Month Out



The following schedules represent a lead-up to Black Friday, but you can alter them to accommodate any peak traffic spike throughout the year.

### Code Freeze Schedules

#### 1 Implement tiered deployment restrictions

- October 1:**  
Director-level approval for feature flag activation
- November 1:**  
VP-level approval for feature flag activation
- November 15:**  
VP-level approval for any code deployments

#### 2 Testing environment management

- Continue deploying new features to testing environments
- Maintain separation between production and testing
- Plan for post-holiday code consolidation

### Final Performance Testing

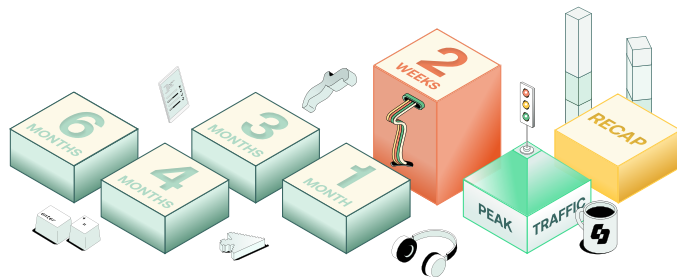
#### 1 Full-system testing under load

- Test with 5x last year's peak traffic
- Simulate realistic user patterns and regional distribution
- Monitor all system components under load

#### 2 Browser and device testing during load

- Run real browser tests on platforms like Sauce Labs during load tests
- Verify page performance and user experience under stress
- Validate all revenue paths work correctly

## Milestone 4: 2 Weeks Out



### Final Preparations

#### 1 Update and validate the static site

- Refresh content for current promotions
- Test the failover mechanism
- Ensure all team members know how to trigger the switchover

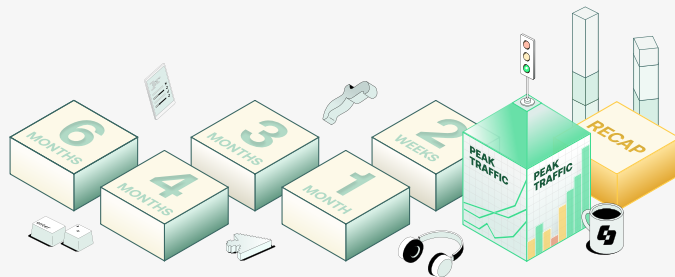
#### 2 Monitoring setup

- Deploy dashboards in visible locations
- Configure alerting systems
- Ensure proper instrumentation with tools like Sauce Labs error reporting

#### 3 Team scheduling

- Plan for all-hands coverage during peak periods
- Establish clear escalation paths
- Ensure everyone knows their responsibilities

## Milestone 5: During Peak Holiday Traffic



### Operational Management

#### 1 Active monitoring

- Watch dashboards for anomalies
- Monitor performance across regions
- Track user behavior patterns

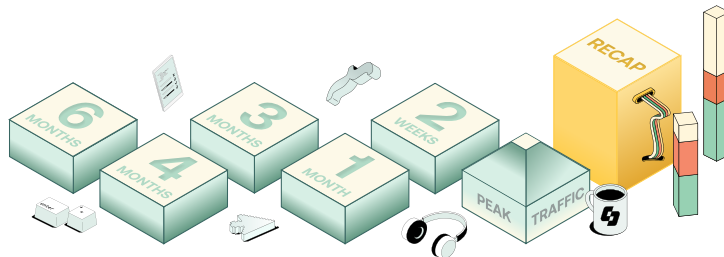
#### 2 Rapid response protocols

- Establish clear thresholds for intervention
- Use feature flags to disable problematic features
- Implement failover procedures when necessary

#### 3 Partner monitoring

- For affiliate businesses, monitor partner site performance
- Prepare contingencies for partner outages

## Milestone 6: Post-Holiday Analysis



### Operational Management

#### 1 Celebrate successes

- | Acknowledge what went well
- | Recognize team contributions

#### 3 Traffic pattern review

- | Analyze actual vs. projected traffic
- | Identify unexpected patterns or user behaviors
- | Note changes in device or browser mix

#### 2 Incident analysis

- | Perform root cause analysis for any issues
- | Document findings and recommendations

#### 4 Planning for next year

- | Begin documenting lessons learned
- | Identify areas for improvement
- | Update forecasting models





# Key Considerations Throughout the Process

## Revenue Path Focus

- Prioritize testing and stability for direct revenue-generating paths
- Ask for each feature: "If this breaks during Black Friday, would we take down the site to fix it?" (The answer may often be no here)
- Be selective about what gets through testing during high-risk periods

## Infrastructure Optimization

- Balance between sufficient capacity and cost management
- Implement infrastructure scaling mechanisms
- Consider cross-cloud provider failover for maximum redundancy (if budget allows)

## Team Collaboration

- Include all relevant stakeholders (engineering, product, marketing, content)
- Ensure clear communication channels
- Document and share responsibilities

## Regional and Global Considerations

- Account for region-specific holidays and shopping patterns
- Test from the actual regions your traffic comes from
- Consider cultural differences in shopping behaviors

## Takeaways

By following this comprehensive timeline and focusing on the most critical elements of your system, you can significantly reduce the risk of outages or performance issues during the high-stakes holiday shopping season. Remember that preparation is an ongoing process that should evolve based on your specific business needs and past experiences.

**Learn more about how Sauce Labs helps retailers provide frictionless shopping experiences**

[Retail testing solutions](#)

