



## Checklist

# Quick Wins to Optimize Cloud Infrastructure

### 1. Enable Content Delivery Networks (CDNs)

- Static Assets:**  
Ensure all static assets (e.g., images, CSS, JavaScript files) are served via a CDN.
- Dynamic Content:**  
Can you easily increase resources (compute, storage, bandwidth) as needed during peak traffic?
- Performance Monitoring:**  
Monitor CDN performance to ensure optimal response times across different regions.
- Resource Flexibility:**  
Can you easily increase resources (compute, storage, bandwidth) as needed during peak traffic?

### 2. Optimize Database Queries and Indexing

- Analyze Queries:**  
Use database performance monitoring tools to identify slow queries.
- Add Indexes:**  
Add appropriate indexes to reduce query execution time.
- Optimize Queries:**  
Rewrite inefficient queries to minimize resource usage during high-traffic periods.



### 3. Leverage Auto-Scaling and Right-Sizing

- Review Auto-Scaling Policies:**  
Adjust auto-scaling policies to meet anticipated demand during peak periods.
- Right-Size Resources:**  
Use cloud provider tools (e.g., AWS Trusted Advisor, Azure Advisor) to identify and correct under- or over-provisioned resources.
- Align with Demand Forecasts:**  
Ensure your auto-scaling setup is aligned with real-time demand forecasts to optimize resource usage.

### 4. Implement Caching Mechanisms

- Server-Side Caching:**  
Implement server-side caching solutions such as Redis or Memcached to store frequently accessed data.
- Cache Static Resources:**  
Use HTTP caching headers (Cache-Control, ETag) to instruct browsers to cache static resources.
- Monitor Cache Performance:**  
Track cache performance and hit/miss ratios to ensure the effectiveness of your caching strategy.



## 5. Compress and Optimize Media Files

- Use Compression Tools:**  
Compress image and media files using tools like ImageOptim or TinyPNG ect.
- Lazy Loading:**  
Implement lazy loading to defer loading of images and videos until the user scrolls to them.
- Next-Gen Formats:**  
Consider using modern image formats such as WebP to achieve better compression rates without losing quality.

## 6. Minimize and Optimize Code

- Minify CSS and JavaScript:**  
Minify your code using tools like UglifyJS or CSSNano to reduce file size.
- Code-Splitting:**  
Implement code-splitting strategies to load only the necessary code for each page.
- Reduce Third-Party Scripts:**  
Review and reduce third-party scripts or plugins that may slow down page performance.



## 7. Monitor and Optimize API Calls

- Review API Calls:**  
Regularly review and optimize frequently used API calls to minimize data transfer.
- Rate Limiting & Batching:**  
Implement rate limiting and request batching to prevent API overload during traffic spikes.
- Cache API Responses:**  
Cache API responses wherever possible to reduce the number of repeat requests.

## 8. Implement Real-Time Performance Monitoring

- Set Up Dashboards:**  
Configure performance dashboards to monitor key metrics like CPU usage, memory, and network throughput.
- Use Monitoring Tools:**  
Use cloud-specific monitoring tools such as AWS CloudWatch, Azure Monitor, or Google Cloud Monitoring to track infrastructure health in real time.
- Set Up Alerts:**  
Implement proactive alerting and anomaly detection to catch early signs of infrastructure strain.

**Have questions or need help?**

**Find us at [AknoStic.com](https://aknoStic.com)**