Poster (album.me) (whatsup.me)

Testing & Performance Considerations

- Static Files storage and retrieval

- Database transaction speed

- Load balancing and Data Transfer

- Local Improvements & Web Improvements

Tests

Response testing
Measure how long the page takes to load and render.

- Django Debug Toolbar (database queries)
- Google PageSpeed (site optimization)
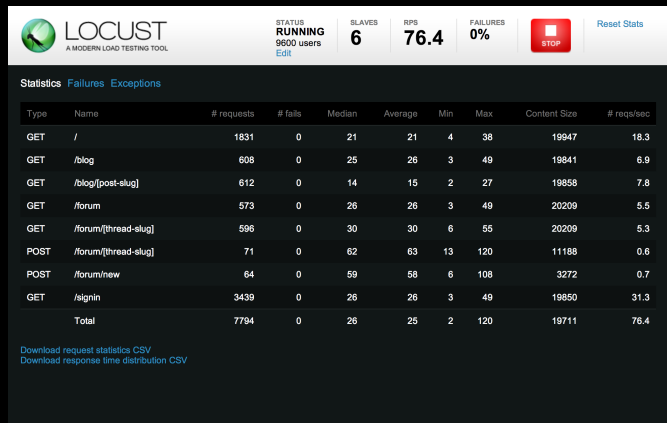- Browser check (loading time)

Performance Tests

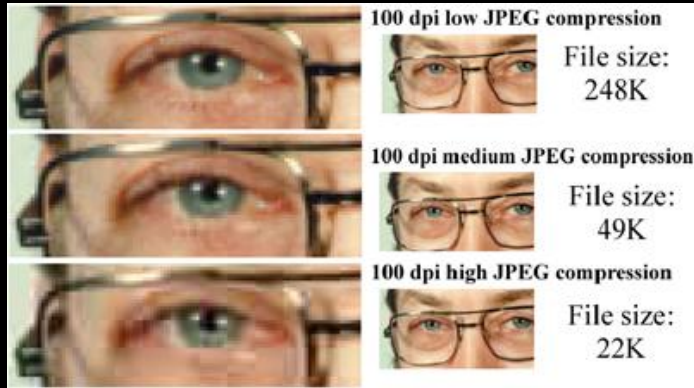Munin: Measure and track performance of hardware and software. Tracks difference

Apache Bench: Stress testing a web server by sending tons of requests and measuring performance

Locust.io: Stress testing a web server through this python API. Allows for user behavior definitions, and up to millions of concurrent user requests.

Improvement

Images:
- Compression (django imagekit)
- Serve off CDN
- Optimize for web (resize and use jpg/png)
- Use Sprites (reduce network requests)

Files:
- Minimize CSS/JS files
- Reduce file size (GZip)
- Load JS near end of body

Django:
- Reduce abstraction (operate on db level)

Off Load

Off load resources to other companies.
- S3 to service static files (fast and cheap
- Amazon Web Service to run the server (scalable and reliable)
- Cloudflare for DNS managing and caching, and website optimization

Caching

Johnny-Cache Package
- Django Caching

Framework Package

- Uses Memcached API

- Increase response time

Template Caching

- Cached.Loader to reduce recompiling of templates