

## Part 4: Bonus answers

### 1. System Design and Scalability:

- **Optimize Database:** Use CodeIgniter's profiler to identify and optimize slow queries.
- **Implement Caching:** Use database, page, and opcode caching mechanisms.
- **Horizontal Scaling:** Add more servers and use a load balancer.
- **Use a CDN:** Serve static assets via a Content Delivery Network.
- **Monitor Performance:** Use tools like New Relic
- **Centralize Logs:** Use solutions like the ELK stack.
- **Backup Regularly:** Ensure robust backup and recovery processes.

### 2. Security:

- **Validation:** Use CodeIgniter's Upload Library to strictly validate file types based on MIME types, and set a maximum file size.
- **Storage & Access:** Rename uploaded files, store them outside the web root or in a protected directory, and ensure they're not executable. Use .htaccess or server configurations to prevent direct execution.
- **Monitoring & Protection:** Implement rate limiting, use CSRF tokens for upload forms, and maintain logs of all file uploads to monitor for suspicious activity.