



# WordPress Bottlenecks

Typical issues and how to easily solve them



# The Advantages of **WordPress**



## Fast Development

Premade Layouts

Niche Themes



## No Code

Plug & Play

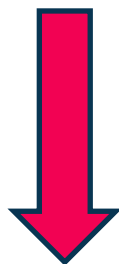
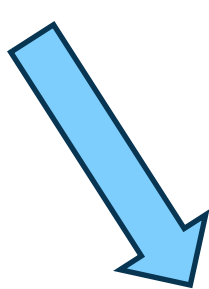
Easy to Use



## Flexibility

25 000+ Themes

60 000+ Plugins



# WORDPRESS



# Most frequent **WordPress** issues



# Security

Approximately 5,000 vulnerabilities were reported in 2023.



# Speed

Average load time: 3.79s



# Functionality

The real showtopper

# Security

## Typical WordPress attacks

- **Low risk attacks** (spam, traffic hijack, etc)
- Automated attacks
- Exploiting 0-day vulnerabilities
- Exploiting recently fixed known vulnerabilities

# Security

## Deal with WordPress security

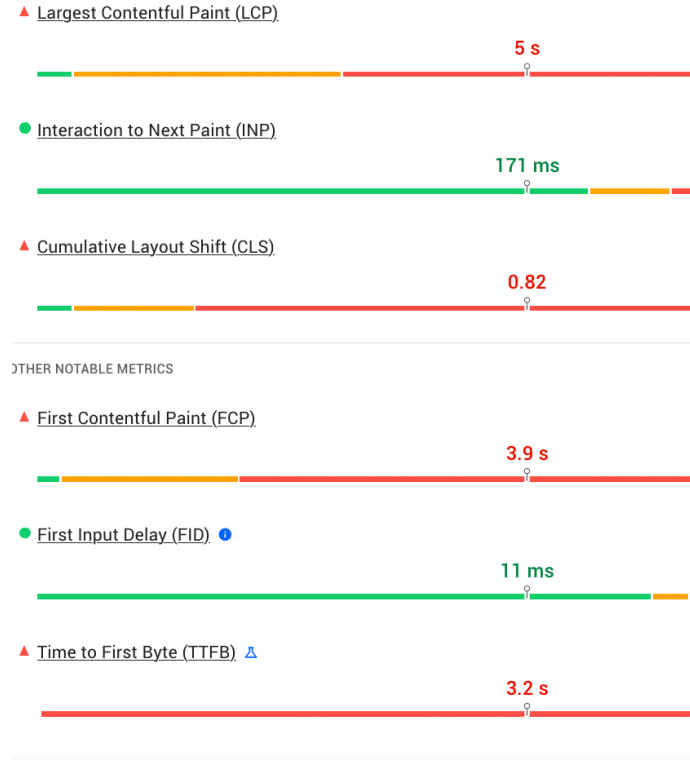
- Backup (at least daily!)
- Keep WordPress up to date
- Use Web Application Firewall
- Use Security plugins



# Speed

## WordPress Speed Issues

- Page generation time
- Bloated resources
- Wrong media types
- JavaScript DOM manipulations
- Wrong loading priorities
- 3rd party resources
- Network





# Speed

## Choose the right hosting

- Know your visitors (location, number)
- Proper planning (technologies, plugins, theme)
- Scalability (Cloud, VPS)
- CDN, caching

**HETZNER**



**KINSTA**

SiteGround

# Speed

## Caching and optimization

- Use a cache plugin (better cache management)
- Use edge caching (Cloudflare)
- Critical CSS
- Plugin & asset organizers
- Prefer all-in-one optimization plugins



# Speed

## Use the proper theme & plugins

- Use a niche or custom theme
- Avoid heavy page builders
- Avoid multipurpose themes and plugins
- Ensure code quality
- Limit the number of plugins

QXYGEN

 **bricks**



# Speed

## Optimize media

- Use modern image formats
- Use proper image sizes
- Avoid self-hosted videos

webp

 YouTube



*vimeo*

# Speed

## Tweaks

- Custom CSS to reduce CLS
- Avoid large hero images/sliders
- Avoid embedded content above the fold
- Tweak the loaded resources (Perfmatters, WP Shifty)
- Tweak the loaded plugins (Plugin organizer, WP Shifty)



# Functionality

## What are functional errors?

All errors which **break site functionality**, like:

- Broken forms/buttons
- Unresponsive elements
- Layout issues on specific devices
- JavaScript errors

# Functionality

## What can cause functional errors?

- Bugs in the code
- Plugin conflicts
- Device/screen size related issues
- Server issues
- Misconfiguration

# Functionality

How can we fix functional errors

- Manual testing (ideally in a staging environment)
- Automated testing tools
- **Realtime User Session Monitoring (RUSM)**





# Functionality

## How does RUSM Work?

### Preparing

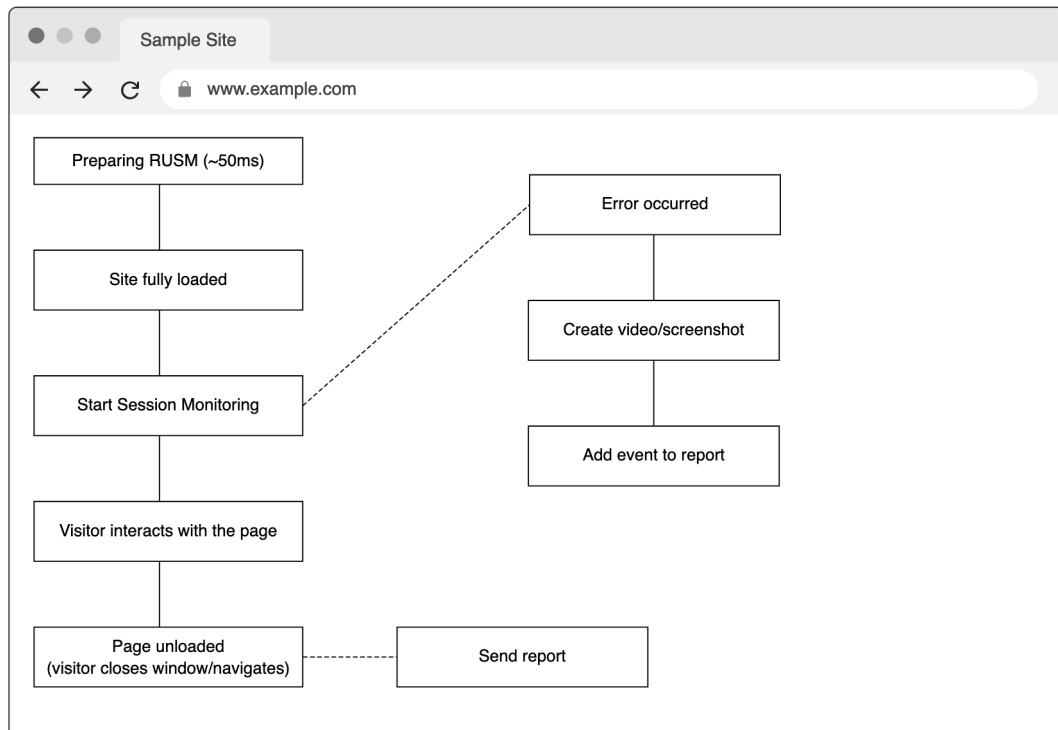
- Set up variables
- Bind events (AJAX/Fetch, CSP, etc)

### Monitoring

- Monitor user interactions
- Create video/screenshots

### Page unloaded

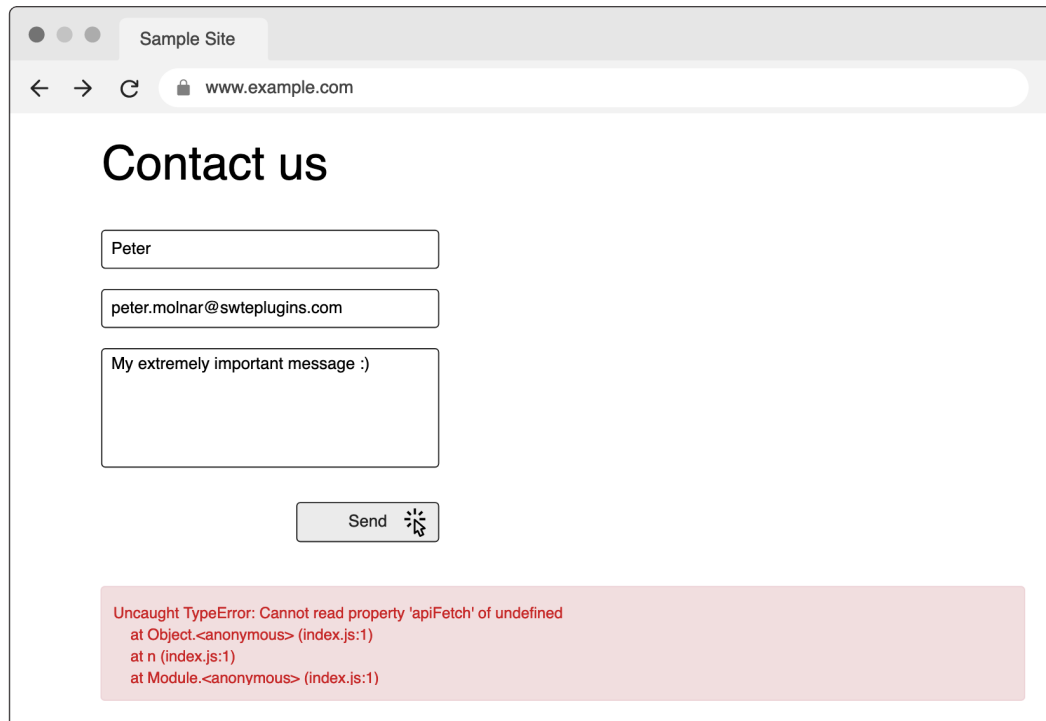
- Send report with beacon request



# Functionality

## Example: Form submission error

1. The visitor fills out the form
2. Clicks to send
3. Nothing happens (form broken due to a JS error)
4. RUSM detects that the form wasn't submitted
5. The visitor leaves the page
6. Errors can now be reported



# Thanks!

## Any questions?

E-mail:

[peter.molnar@swteplugins.com](mailto:peter.molnar@swteplugins.com)

