YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

AUDITING A DRUPAL 7 WEB SITE - OUR METHODOLOGY

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

THE COMPANY

THE COMPANY

YOGARIK is a web engineering firm dedicated to DRUPAL technology.
We build mostly B2B web sites exclusively with DRUPAL CMS.

Our case studies : Intranet, extranet, dedicated portals and any web site that needs to be connected to our customer's IT system

Our business partners : SAGE, CREDIT AGRICOLE, KEA Partners, Swissquote…

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

KEY INDICATORS FOR THE AUDIT

KEY INDICATORS FOR THE AUDIT

Some key indicators highlight the requirement of a technical audit.

Poor performances	Strong regressions	Site in disarray
Difficulties and cost of evolutions	Multiple service providers	
Data loss	Loss of trust	
Site administration problems	Security issues	

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

CASE STUDIES

AUDITS CARRIED OUT FOR OUR CUSTOMERS

SWISSQUOTE THE SWISS LEADER IN ONLINE BANKING

sage

Kea & Partners
Transformation consulting

h5 audits

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

OUR METHODOLOGY IN 9 STEPS

OUR METHODOLOGY

This is our **OWN** methodology

It does not cover servers and production stage audit.

It is based on tools developed by the Drupal community and our own **field expertise**

OUR METHODOLOGY IN 9 STEPS

We propose to :

- Understand both customer and project contexts
- Set-up your local Drupal environment and the audit tools
- Go through the back-office & front-office
- Understand the data model
- Control the Drupal development standards
- Analyse the community modules installed
- Study custom code developed (modules & themes)
- Uncover the deployment, the versioning of source code
- Write the audit report

OUR METHODOLOGY - STEP 1

To understand both customer and project contexts

- Relationships with the former service provider
- Project history
- Perspectives, strategic vision for the project
- Analyse what works and what doesn't
- Present our methodology, our assets, our differences
- Get project documentation (Specifications, CMD, DATD)

OUR METHODOLOGY - STEP 2

To Set-up your local Drupal environment and the audit tools

- Recover the latest production source code
- Recover the production database
- Recover static content (files, images, documents)
- Set-up major tools : drush, hacked, devel, link checker, xhprof, blackfire, code sniffer, coder, firebug, syslog, watchdog, log…
- Ideally get an audit environment identical to the live one

OUR METHODOLOGY - STEP 3

To go through the back-office & front-office

- Read the customer documentation
- Anonymous / authentified pages, user experience
- Administration screens according to roles and access rights
- Functionalities
- Configuration errors

OUR METHODOLOGY - STEP 4

To understand the data model

- Content types and their application
- Taxonomies
- Additional entities
- Menus, tree
- Blocks
- Users, roles, access rights

OUR METHODOLOGY - STEP 5

To control the Drupal development standards

- With **Drupal guidelines**, **code sniffer & hacked** tools and our **CMS expertise**
- The alteration of Drupal Core and community modules
- Missing and applied patches
- In case of specific code, respect of best practices, tree and structure
- Respect of functional packaging via features & co
- Errors, warnings in site logs

OUR METHODOLOGY - STEP 6

To analyse the contributing / community modules installed

- With **drush**
- Activated modules but irrelevant or unused
- Available updates
- Alternatives to existing modules
- Missing modules that could improve the site
- Installation of external Javascript libraries
- Dashboard of Drupal errors reports, performances

OUR METHODOLOGY - STEP 7

To study custom code developed (modules & themes)

- Respect of development standards
- Functional perimeter covered by each module
- The modules centralization, packaging with features
- Performances
- Analyse logs

OUR METHODOLOGY - STEP 8

To uncover the deployment, the versioning of source code

- Deliverables / Deployment between stages
- History and evolutions of source code
- Proposed improvements

OUR METHODOLOGY - STEP 9

To write the audit report

- Introduction
- Context : customer / project
- Synthesis of identified anomalies
- Details of anomalies
- Detailed analyses of the data model
- In annex : development standards and our methodology

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

CONCLUSION

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

QUESTIONS ?

YOGARIK

CONSEIL | INGÉNIERIE | SERVICES

SITES WEB OPEN SOURCE

THANKS