magnolia® SERVICE BRIEF

Project review and code audit

March 2021

Service description

One of our Senior Solution Architects performs a project review and code audit before or after the project lifecycle.

The purpose of a project review and code audit is to validate the use of Magnolia best practices and other guidelines. This is to guarantee the security, stability and sustainability of the platform and forthcoming migrations.

We analyse your project, architecture, or specification and will present the results in a technical or business report. Recommendations on quick wins are also included in this technical or business report.

When this is recommended

The goal of the review is to give the customer feedback on their Magnolia project. The review covers a range of topics from project setup and organization, implementation, and operating a Magnolia deployment.

The review also identifies areas in the project that diverge from the Magnolia roadmap, which may cause problems in keeping pace with Magnolia features and releases. The review provides feedback on the importance or severity of all issues, and recommends fixes or alternative approaches if possible.

We recommend a project review and code audit for Magnolia projects older than two years. It is also recommended for projects with complex setup and customisations.

A project review and code audit is ideal for projects set to be migrated to the most recent Magnolia version.

We also advise a project review and code audit when changing to a new partner. This provides the new partner an overview of the current project and code status.

Project review and code audits do not include performance review or diagnostics service.

Knowledge requirements

Our Senior Solution Architect will be coordinating with one of your technical point of contact to get an overview of the project, gain access to the code and have a continuous exchange throughout the service delivery.

The technical point of contact should ideally have:

- In-depth knowledge of the Magnolia project to be audited or reviewed
- Sufficient Magnolia knowledge and project experience
- Access to the project code, external modules and database, snapshot of sources and content repository

Technical requirements

As a standard practice, we ask for a run-able or build-able copy of the project. If a run-able copy of the project is not possible, we require access to a live instance instead.

This includes:

- A copy of all Java code and light modules as last deployed to Production
- Copies of all dependencies used when building the project which we cannot otherwise access via public repositories (eg. Maven, NPM, etc)
- Details of all configurations necessary to match the current production environment if this is not automatically boot-strapped or contained within properties files
- All necessary documentation to setup, edit, build and run the project locally
- Any other pertinent documentation on set up, build processes and, in particular, deployments

Additionally:

- Where the system interacts with external services (eg. databases apart from the JackRabbit source, APIs, online storage) we will require connection details, credentials, etc.
- Where private services cannot be reached via the public internet, we will require details of how to work around this so the system can still be run

Delivery

It takes five man-days to complete a project review and code audit for a standard setup. More days may be required for large and complex projects with customisations.

Focus of service	Number of days	Tasks breakdown
Pre-go live review	5	**Day 1:** Kick-off call, code access and tools setup **Day 2–4:** Project review and code audit **Day 5:** Results discussion and/or follow-up
Project review and code audit	5	**Day 1:** Kick-off call, code access and tools setup **Day 2–4:** Project review and code audit **Day 5:** Results discussion and/or follow-up

Scope

Tasks

- Review and analysis of the actual project structure
- Identification of the actual configuration and implementation (modularization, organization and documentation) that do not follow the best practice and propose the best implementation
- Identification of what should be adhered to in the future as a guideline / specification for all implementations
- Provide a solution pack with the suggestions
- Review and follow-up

Deliverables

- In-depth analysis
- Recommendations on quick wins, mid-term and long-term actions
- Technical or business report

The result will be a comprehensive technical analysis with all findings and recommendations. The standard format of the report is DIN A4, with a maximum length of 30 pages.

It is possible to request a special format if there is interest in a particular report or a more specific one. Extra fees will apply.

Scope

Technical or business report

Each review item is classified:

- **Best practice: Magnolia recommended** - organization and approaches corresponding to the Magnolia roadmap and avoidance of common Magnolia problems.
- **Best practice: clean code** - general guidelines for writing clean code (usually Java), not necessarily Magnolia specific but helpful to maintaining your Magnolia project.

Each review item is evaluated by severity or significance:

- **Problem: immediate correction recommended** - a problem that should be corrected as soon as possible, may cause significant headaches if left as is.
- **Problem: correct soon** - a problem that should be addressed in the near future, may cause difficulties in using a new Magnolia release or not align with the Magnolia roadmap.
- **Problem: minor concern** - a stylistic or cosmetic problem that can affect the readability of code.

The review covers the following areas:

- Magnolia project setup
- Magnolia module organization
- Magnolia light development
- Magnolia themes
- Magnolia FreeMarker templates
- Magnolia Java code
- Magnolia legacy
- Magnolia operation (optional)

Method and software

Our Senior Solution Architects follow the Magnolia operational audit guidelines to perform a project review and code audit. PMD, a source code analyser, is also used. Perl scripts are used to automate part of the review steps. These scripts mainly check light modules to verify naming patterns, hard-coded names and descriptions and more.

Request and schedule

Reach out

Please contact the Magnolia account manager in your area to learn more about our service packages or request a specific service.

Book in advance

We always advise you to book in advance. If you decide to book early, we'll have ample time to evaluate and devise solutions.

Continuous exchange

Through guidance and assistance, we ensure project success and in-depth exchange with the Magnolia Services team.

Complete tasks

Your success is our success. Together we'll solve your project-related problems or improve the overall performance of your Magnolia bundle.

Follow-up call

We don't leave you right after providing solutions and code packs. We are interested in knowing if our solutions work in your project. A follow-up session is typically arranged just before a service package culminates.

Contact

Cass Weber
Professional Services Manager

Oslo-Strasse 2, 4142 Münchenstein (Basel), Switzerland

Office: +41 61 228 90 00

www.magnolia-cms.com