

# Requisiti per la scelta di un Content Management System

## Aspetti comunicativi e gestionali

Nicola Zotti - Angon

### 1. Criteri generali

### 2. Tecnologie e processi di collazione dei contenuti

### 3. Gestione

### 4. Publishing

#### **Premessa**

Questo documento esamina i requisiti per la scelta di un CMS che faccia parte, supportandolo, del progetto costitutivo di un Enterprise Portal.

I requisiti indicati, quindi, prevedono gli sviluppi di un progetto di ampia portata e sono emendabili o integrabili da esigenze particolari.

## 1. Criteri generali

### **1.1. Processi e staff del progetto**

- a) Il fornitore deve descrivere sinteticamente i passaggi principali per l'implementazione del sistema.
- b) Il fornitore deve segnalare i fattori necessari ad assicurare il successo del progetto.
- c) Devono essere fornite indicazioni dei tempi prevedibili e delle misure che possono essere intraprese per minimizzarli.
- d) Può essere molto utile prendere in esame progetti simili a quello dell'azienda, già trattati dal fornitore.
- e) Deve essere richiesta una descrizione dei vari ruoli di progetto previsti (sia del fornitore che interni all'azienda) e il numero di persone coinvolte per ciascun ruolo.
- f) Il fornitore deve indicare i servizi professionali necessari per la completa realizzazione del progetto, indicando quelli che può fornire direttamente, quelli che raccomanda e una descrizione delle posizioni di staff e dei costi.

### **1.2. Sviluppo**

- a) Le capacità e i tool necessari a creare e modificare i template basici per l'autoring, il submitting e il publishing dei contenuti non devono richiedere particolari skill di programmazione: in linea di massima una persona che utilizza un editor HTML grafico deve potere imparare facilmente e rapidamente l'utilizzo del tool.
- b) Sono preferibili linguaggi di sviluppo non proprietari e tool di layout indirizzati alla tipologia di skill di un designer grafico.
- c) Sugeriamo che le funzioni amministrative del prodotto – incluso workflow, amministrazione utente, database management e reporting – possano essere richiamate in modo indipendente utilizzando tecnologie Web come ASP, JSP, CORBA, COM, J2EE e simili.

- d) I linguaggi e i tools che automatizzano l'authoring e l'acquisizione dei contenuti devono essere estendibili e modificabili e offrire una integrazione diretta con i comuni sistemi di authoring come la suite Office, editor HTML come Dreamweaver, applicazioni come Flash, ecc.

### **1.3. Localizzazione**

- a) Il fornitore deve presentare una panoramica delle opzioni di localizzazione del prodotto.
- b) In linea di massima, il sistema deve mantenere collegamenti tra componenti tradotte, per un'agile gestione degli aggiornamenti, deve essere capace di gestire contenuti in lingue diverse, date in formati internazionali, e varie valute. Inoltre, è importante che le videate utilizzate dai gestori dei contenuti siano facilmente traducibili in lingue diverse.
- c) Sarebbero desiderabili meccanismi comparativi tra testo originale e versione tradotta per un confronto diretto.
- d) Devono essere possibili il versioning e il tracking delle modifiche a livello di singolo elemento, in modo da poter destinare alla localizzazione alcuni elementi, separandoli da altri che non necessitano di localizzazione.
- e) Sarebbe utile fornire User Interface multilingue per autori non italiani, localizzando la loro interfaccia.

### **1.4. Training**

- a) Il fornitore deve presentare una panoramica dei servizi di addestramento che può erogare.
- b) Deve indicare disponibilità, tempi, costi e sedi per permettere di valutare l'incidenza dell'addestramento sui tempi di avvio del progetto.
- c) Devono essere inoltre indicate le posizioni di staff per le quali viene fornito l'addestramento: oltre l'utente finale e l'amministratore è utile sapere se c'è addestramento per sviluppatori, managers, informations architect, staff IT, designers, ecc.

### **1.5. Supporto tecnico**

- a) Dal punto di vista comunicativo è utile conoscere quali strumenti di autoaiuto sono previsti: FAQ, tutorials, help, ecc.
- b) In termini generali in fase di realizzazione e di avvio del progetto può essere utile avere un punto di riferimento per il supporto tecnico da indicare agli utenti.

### **1.6. Impiego**

- a) Tutte le funzionalità di base devono essere disponibili tramite un'interfaccia Web in modo che agli utenti sia necessario solo un browser per accedere al sistema.
- b) Deve essere disponibile un software client per consentire un'interfaccia più ricca di funzionalità specifiche.
- c) I browser utilizzabili devono essere quelli attualmente in uso dall'azienda.
- d) Il prodotto deve essere accessibile anche all'esterno del firewall per consentire a utenti remoti di contribuire al CMS.
- e) Il prodotto deve consentire la contribuzione in un ambiente distribuito in numerose sedi e località, mantenendolo organizzato e minimizzando le ridondanze.

## **2. Tecnologie e processi di collazione dei contenuti**

### **2.1. Authoring**

- a) Può essere utile la presenza nel CMS di strumenti di authoring basici: word processor, editor HTML, creazione immagini.
- b) Più utile ancora è l'integrazione diretta, ovvero la presenza del CMS nei menù dei principali tool di authoring utilizzati dall'azienda.

- c) Il fornitore deve provvedere un elenco dei tipi di file che il CMS può gestire – come sistema di document management – ed un elenco di quelli che può invece interpretare.
- d) Il fornitore deve indicare i formati di file che il CMS può riconoscere e convertire in XML o in record del database
- e) Il fornitore deve descrivere come un autore inesperto può contribuire contenuti di testo in una pagina Web, includendo grafici e altri tipi di media all'interno del testo: il meccanismo deve essere sufficientemente semplice da consentire la creazione di pagine semplici con scarso o nessun aiuto.
- f) Analogamente è bene conoscere il supporto che viene fornito nei form al linking e all'uploading di immagini e di altri media: aggiungere media di vario tipo associandoli al testo deve essere il più possibile semplice.
- g) Il fornitore deve illustrare il metodo di inserimento di contenuti basato su interfaccia Web.
- h) Il CMS deve consentire l'editing di rich HTML o di XML nei campi dei propri form Web based, in modo che gli autori possano inserire anche attributi di formato quando realizzano i contenuti.
- i) I form di input devono poter essere personalizzabili a seconda delle autorizzazioni dei vari autori: visualizzando nei form campi diversi a seconda dei contenuti che devono inserire.
- j) Il CMS deve poter creare automaticamente i form da uno schema DTD (document type definition) e/o database: l'autogenerazione di form Web based su regole predefinite consente notevole risparmio di tempo.
- k) I form Web based devono poter essere estesi o modificati da un linguaggio non proprietario come l'HTML mediante un editor.
- l) I form del CMS devono poter rappresentare strutture relazionali gerarchiche: questo facilita l'inserimento di titoli e di testo nelle sezioni e sottosezioni.
- m) Il CMS deve poter consentire il lavoro off-line, rimandando l'inserimento dei contenuti a quando l'autore è connesso.
- n) Sarebbe oltremodo utile una funzione di preview del contenuto composto mediante i form.
- o) L'autore deve poter consultare la revisione della History e del Workflow del proprio lavoro.
- p) E' utile una funzione di controllo ortografico per chi edita direttamente i testi nei form.
- q) Il fornitore deve indicare una previsione dei bisogni di addestramento dei nuovi autori di contributi.

## **2.2. Acquisizione**

- a) Il fornitore deve illustrare le capacità dei tool di acquisizione: se possono, ad esempio, scansionare automaticamente fonti di contenuto catturando le informazioni necessarie, oppure se il CMS può essere abilitato ad accettare fonti di syndication come news, valori di borsa, ecc.
- b) E' bene conoscere l'eventuale integrazione con tool di acquisizione di terze parti, per una ulteriore espansione delle capacità di acquisizione di contenuti.
- c) Il CMS deve poter migrare siti Web esistenti all'interno del proprio sistema: possibilmente in modo automatico e senza l'intervento di analisi dei contenuti, redesign e strutturazione dei contenuti predefinita.
- d) Il CMS deve poter gestire, sulla base del Workflow aziendale, lo scheduling, i diritti d'uso, l'attribuzione e il pagamento dei contenuti acquisiti.
- e) Il CMS deve poter ritrovare automaticamente – con download periodici e senza intervento umano – record da database disponibili in sede locale, trasformandoli in componenti di contenuto nel proprio repository.

## **2.3. Conversione**

- a) Se i contenuti acquisiti non sono nel formato e nella struttura richiesta dal sistema (ad esempio hanno una navigazione o header e footer non necessari), il CMS deve poterli convertire automaticamente utilizzando tool interni o di terze parti.
- b) Può essere utile una conversione automatica attivata dalla creazione dinamica di pagine Web, in modo che testi e media siano convertiti solo a richiesta.

## **2.4. Aggregazione**

- a) Di particolare importanza è l'inclusione o l'integrazione con strumenti di gestione dei metadata: con autodiscovery e tool di creazione di tassonomie, che aiutino il design di liste di metadata e ritrovino automaticamente i metadata nei file processati.
- b) Il CMS deve applicare in modo automatico valori di metadata a contenuti basandosi sulla tipologia sulla location, ad esempio con una applicazione batch di metadata sulla base di una directory o l'assegnazione di metadata di default a seconda del tipo di contenuto.
- c) Il CMS deve avere un tool che consenta all'amministratore di combinare o dividere i campi di metadata esistenti nel sistema: ad esempio se si desidera fare un merge di due campi, oppure se è necessario suddividere un campo.
- d) Deve essere possibile estrarre liste di metadata da altre fonti, come database o semplici file, ad esempio per estrarre una lista di prodotti da un database, presentandola automaticamente in una tabella Web.
- e) Analogamente, deve essere possibile impostare una lista di metadata e quindi esportarla in file esterni e database: ad esempio modificando nel CMS una lista di prodotti e quindi aggiornando il database da cui originava.
- f) Nel processo di gestione dei metadata è necessario che sia possibile rivedere periodicamente la loro validità sulla base di una lista, per aggiornarli o cancellarli.
- g) Il sistema deve poter dialogare con le iniziative internazionali di uniformazione dei metadata ed eventualmente aiutare la modifica e l'adeguamento dei metadata aziendali.
- h) Infine è necessario che il fornitore indichi se il prodotto supporta tutte le varie tipologie di metadata: strutturali (dai singoli caratteri ai gruppi di pubblicazioni), di formato (applicabili ad ogni livello strutturale), di accesso (gerarchie, indici, associazioni e sequenze), gestionali (per il tracking e l'amministrazione), di inclusione (per referenziare i contenuti non inclusi nella struttura dei contenuti).

## **3. Gestione**

### **3.1. Storage**

- a) E' necessario conoscere le metodologie di storage dei contenuti del CMS per evidenziare eventuali conflitti con i requisiti infrastrutturali e per sapere se sarà in grado di gestire tutte le tipologie di contenuti che si intende gestire.
- b) Il CMS deve poter gestire gerarchie multiple, indicizzazioni multiple, cross reference tra componenti e tra pagine pubblicate, e sequenze: questo per poter costruire pubblicazioni multiple, ciascuna con la propria tavola dei contenuti.
- c) L'archivio dei contenuti deve poter supportare i set di caratteri estesi.
- d) Il sistema di archiviazione deve supportare gli "schema" o i DTD dell'XML, creando un DTD per l'intero repository e DTD multipli assegnabili a parte della struttura complessiva del repository.
- e) Il CMS deve avere la capacità di controllare la validità dei link e delle relazioni con eventuali media che compaiono entro i campi di testo del repository.
- f) Il search del CMS deve consentire il rich indexing e il rich quering, la ricerca mediante metadata (parole chiave), ricerca full-text, ricerca di prossimità, e risultati presentati secondo grado di rilevanza. Inoltre il sistema di ricerca deve avere la capacità di escludere quei risultati che non si è autorizzati a vedere.

### **3.2. Amministrazione**

- a) Il sistema deve consentire a utilizzatori non tecnici di gestire autonomamente i propri contenuti.
- b) Tutti gli amministratori devono utilizzare tool Web-based. Se ci sono alcune funzioni amministrative che richiedono l'uso di software particolari, questo deve essere specificato in anticipo.

- c) Le persone autorizzate devono poter effettuare il search & replace globale in tutto il repository: sia nei contenuti che nei metadata.
- d) Sarebbe utile verificare e gestire automaticamente link interni ed esterni interrotti.
- e) La configurazione dei ruoli e dei diritti di utente deve essere elastica e completa per poter aderire sia ai servizi di directory del network aziendale, sia alle gerarchie delle tipologie e dei ruoli di utenza.
- f) Il CMS deve garantire la sincronizzazione e la replica tra database multipli, gestendola con apposite autorizzazioni e comunicazioni.

### **3.3. Workflow**

- a) Il fornitore deve presentare una panoramica degli strumenti di workflow del CMS.
- b) Anche se il prodotto ha propri tool di workflow – e ovviamente se non li ha – è importante conoscere se è abilitato ad usare applicazioni di workflow di terze parti: si deve avere la sicurezza che eventi e cambiamenti di status dei contenuti nel CMS innescano gli opportuni nell'applicazione di workflow esterna e viceversa.
- c) Allo stesso modo, anche semplici input interni od esterni (ad esempio una e-mail o un file aggiunto ad una directory) devono poter innescare processi di workflow del CMS.
- d) Il CMS deve possedere un'interfaccia utente che fornisce all'amministratore un accesso completo e semplice a tutte le funzioni offerte dal prodotto.
- e) Il CMS deve supportare sia workflow sequenziali che paralleli.
- f) Il workflow deve trattare un ampio range di attività di sistema, ad esempio: richieste di contenuto, sottomissione di contenuto, aggiunte di contenuto, cancellazione e modifica di contenuti esistenti, raccolte di contenuti, revisioni, approvazioni, migrazioni, approvazione di migrazioni, ecc.
- g) Gli attributi di contenuto o di sistema utilizzati per le regole dell'automatizzazione dell'instradamento del workflow devono poter utilizzare ogni metadata utilizzato.
- h) La creazione e la modifica dei workflow non deve richiedere skill di programmazione e l'interfaccia deve essere molto completa ed estendibile.
- i) Il CMS deve gestire tutte le richieste di authoring e di publishing, incluse sottomissione, aggiunta, cancellazione ed editazione di contenuti, revisione/correzione di contenuti, approvazione, migrazione, pubblicazione e test.
- j) I tool di workflow devono essere aperti ad eventuali partecipanti esterni (ad esempio fornitori o consulenti) che entrino nel CMS eventualmente tramite una connessione l'internet lenta, senza accedere alla LAN aziendale attraverso il firewall, oppure anche con una e-mail.
- k) Deve essere possibile aggiungere commenti a ciascuno stadio del workflow.
- l) Deve essere conservata una storia del percorso di controllo, in modo da poter seguire a ritroso passo dopo passo il percorso di un segmento di contenuto.
- m) Il processo di Workflow deve poter essere gestito interamente anche solo con e-mail con link interni che lanciano verso i task richiesti nel CMS.
- n) I tool di workflow devono consentire di fermare o reindirizzare un processo di workflow, superando gli automatismi.
- o) Deve poter essere assegnato un workflow anche ad elementi particolari di un contenuto, in modo da consentire di eliminare ridondanze e ripetizioni nei processi.

### **3.4. Gestione delle integrazioni**

- a) Il CMS deve potersi integrare con:
  - sistemi di Document management
  - Sistemi di ERP
  - Sistemi di e-procurement
  - Database customizzati dell'azienda
  - Altri ed eventuali da determinare con le esigenze dell'azienda.

- b) Per connettersi e condividere dati con altri database utilizzabili dagli amministratori, il CMS deve utilizzare l'interfaccia non tecnica. Capacità particolari possono essere gestite con estensioni realizzate da programmatori in linguaggi standard.
- c) Si devono poter lanciare altre applicazioni dall'interno del CMS: ad esempio la suite Office e Acrobat. Il CMS deve inoltre riconoscere dall'estensione del file l'applicazione cui appartiene e lanciarla.
- d) L'accesso e la condivisione dei dati deve avvenire sulla base di standard aperti, come XML e ODBC.

### **3.5. Versioning**

- a) Il CMS deve conservare le varie versioni dei contenuti, in modo che si possano recuperare le versioni precedenti.
- b) Il livello di granularità supportato per il controllo in/out dei contenuti deve essere ampio, in modo da consentire l'authoring collaborativi.
- c) Per lo stesso motivo i processi a ritroso devono avere lo stesso ampio grado di granularità.
- d) Il blocco dei contenuti in uscita dal controllo deve poter essere esteso a tutte le componenti, e alle varie parentele della gerarchia del repository.
- e) E' utile un versioning comparativo per facilitare i controlli.

### **3.6. Reporting**

- a) Si devono poter effettuare track e log degli utenti, il tempo di accesso, le azioni eseguite (come il download di una particolare versione di un file oppure l'approvazione o la correzione di un file). Si deve poter inoltre monitorare l'uso del sistema per utente, gruppo, area geografica, ecc.
- b) Sono necessari anche particolareggiati strumenti di reporting dei workflow.
- c) Il CMS deve controllare l'uso di media coperti da copyright, acquistati da fornitori esterni.
- d) Il CMS deve compilare la mappa del sito, includendo i nomi delle pagine e le statistiche.

### **3.7. Disaster recovery, sicurezza, performance**

- a) Le feature di disaster recovering devono comprendere il versioning, il processo a ritroso e back-up automatici, che devono essere parte del software di database usato dal CMS, ma devono essere controllabili dall'interfaccia amministrativa del CMS.
- b) Qualsiasi autorizzazione già disponibile nel sistema operativo deve essere trasferibile al CMS.
- c) La capacità di pubblicare contenuti e media deve essere sottoposta a una procedura di sicurezza.
- d) I requisiti hardware e software del CMS devono coprire le varie esigenze dell'azienda.
- e) Il CMS deve possedere limiti massimi di regole, di componenti uniche archiviabili, di contributori contemporanei supportabili, di utenti adeguati alle esigenze aziendali.

## **4. Publishing**

### **4.1. Template**

- a) I linguaggi di produzione dei template devono essere quelli in uso nell'azienda, o comunque linguaggi aperti con un'ampia base di competenze diffuse nel mondo della programmazione.
- b) Il CMS deve supportare template per pubblicazioni non HTML.
- c) Il CMS deve processare template sia per pubblicazioni di HTML statiche che dinamiche, consentendo l'utilizzo dell'una o dell'altra possibilità a seconda delle convenienze.
- d) La creazione dinamica di pagine deve essere supportata a livello di creazione di elementi di navigazione: search, link, indici.

- e) Il CMS deve supportare i template di componente (che nominano una certa area di tipologia di contenuti e indicano quali aree dinamiche e statiche contiene) e di navigazione (che nominano una certa area di navigazione e indicano quali aree dinamiche e statiche contiene)
- f) I template devono supportare l'integrazione con sistemi esterni.
- g) I template devono essere personalizzabili.

#### **4.2. Content Deployment**

- a) Il CMS deve dispiegare i file di contenuto in web server remoti e anche record di database in database remoti.
- b) Il CMS deve possedere un sito di staging.
- c) Il sistema deve poter dispiegare i contenuti appropriati verso varie destinazioni, basandosi sulle proprietà dei contenuti.
- d) Il sistema deve poter assegnare dei valori de default a seconda delle proprietà dei contenuti.
- e) Il sistema deve automaticamente dispiegare dei file referenziati in un campo.
- f) Deve essere possibile stabilire le date di pubblicazione e di scadenza di un file.
- g) Pubblicazioni frequenti e regolari (e-mail periodiche, materiale di syndication, ecc) devono poter essere pubblicate a determinate scadenze senza intervento manuale.

#### **4.3. Piattaforma**

- a) Il sistema deve potersi integrare con il sistema di e-mail aziendale e pubblicarvi contenuti, ,possibilmente creando e-mail personalizzate.
- b) Il sistema deve poter essere utilizzato per realizzare pubblicazioni a stampa.
- c) Il sistema deve poter essere realizzato per creare pubblicazioni Wap o per altri formati basati su XML.
- d) Il sistema deve poter creare contenuti a formato neutrale (generalmente XML), per dispiegarli su siti FTP, server remoti, dischi rigidi, CD-ROM, DVD, ecc.
- e) Il sistema deve creare contenuti per formati elettronici proprietari come Microsoft Help, Macromedia Director, o altre piattaforme non HTML.
- f) Il sistema deve creare documenti scaricabili in PDF o latri formati necessari all'azienda.

#### **4.4. Personalizzazione**

- a) L'azienda deve fornire una panoramica completa degli strumenti di personalizzazione, specificando le possibili integrazioni con sistemi di personalizzazione di terze parti.
- b) Le regole di personalizzazione devono essere semplici e ben sviluppate, utilizzabili da personale non tecnico.
- c) Il sistema deve consentire l'archiviazione e la consultazione dei dati del profilo utente.
- d) I dati accessibili dal motore di personalizzazione devono comprendere: i dati della sessione utente, le informazioni degli header HTTP, la page history, tutti i metadata dei contenuti, i profili utente interni ed esterni al CMS, i gruppi di utenze, e le regole di personalizzazione.
- e) Il CMS deve produrre pubblicazioni a stampa individualizzate, o altre pubblicazioni non Web personalizzate.
- f) Ricerche, link, indici e altri ausili di navigazione devono riflettere la personalizzazione utente disponibile per uno specifico utente.
- g) Il processo di costruzione delle regole di personalizzazione deve utilizzare un'interfaccia semplice ed intuitiva.
- h) Il sistema deve consentire il collaborative filtering, ovvero il sistema (utilizzato ad esempio da Amazon) mediante il quale un utente riceve consigli sui contenuti da consultare, basandosi sulle consultazioni di utenti simili.
- i) Il sistema deve consentire interazioni tra personalizzazione e profili utente, ad esempio per indirizzare contenuti verso gli utenti. e personalizzazioni basate sui comportamenti.
- j) Il sistema deve la creazione e la conduzione di campagne push.

#### **4.5. Integrazione di publishing**

- a) Deve essere compilato un elenco di sistemi ai quali si vuole collegare le proprie pubblicazioni Web, ad esempio:
- Strumenti di brand management
  - Gestione di pubblicità
  - Gestione di campagne promozionali
  - Gestione di sistemi di merchandising
  - Sistemi di CRM
  - Survey e questionari
  - Sistemi di accesso e di sicurezza Web
  - Prodotti di Data Mining
  - Alberi decisionali
  - Sistemi di download
  - Software per chat
  - Sistemi di bacheca on-line
  - Sistemi di accettazione di e-mail e di risposta automatica
  - Sistemi per transazioni di e-commerce e finanziarie
  - Sistemi di transazione con carta di credito
  - Sistemi per Enterprise portal
  - Sistemi per la conversione di contenuti (ad esempio da XML a HTML)
  - Web service, Web Server e Web application server
  - Applicazioni database
  - Altri necessari all'azienda

#### **4.6. Web User Interface**

- a) Deve essere possibile integrare search engine di terze parti con quelle del CMS integrandolo tra loro.
- b) Il CMS dovrebbe individuare il browser dell'utente per offrirgli la versione delle pagine HTML adatta.
- c) Il sistema deve supportare la gestione delle sessioni utente, includendo cookies e database-caching per una migliore personalizzazione e customizzazione.