

Document Management System Menggunakan Open-Source CMS, Drupal

Alexander Waworuntu, Evawaty Tanuar

Institut Teknologi dan Bisnis kalbe, Jakarta

Abstract:

Organization and individual have digital and non-digital assets in the form of documents. While working sometimes people need information that stored in some documents that owned by other person inside organization. This is why document sharing process is an important part to support people inside organization to complete their tasks. This paper propose an initial process of developing document management system to support document sharing using open-source Drupal content management system (CMS) as a development base. The development including integration various modules to enhance and add features to upload and displaying list of documents. Result of the research is a simple document management system application with classification feature based on keywords and categories. The application also classifying users whose can read and download documents.

Keywords: document management system, drupal, CMS, document sharing

I. Pendahuluan

Pada beberapa bidang pekerjaan seperti pendidik dan peneliti, saling berbagi dokumen adalah hal yang sudah biasa terjadi. Hal ini seringkali dilakukan dengan saling berkirim email berisi *attachment file* atau melalui transfer media penyimpanan seperti *flash drive*. Cara seperti ini dapat dilakukan jika seseorang membutuhkan dokumen tertentu dan dia tahu kepada siapa dia bisa mendapatkan dokumen tersebut. Permasalahan timbul ketika seseorang membutuhkan sebuah dokumen tertentu, namun tidak tahu kepada siapa harus meminta, sehingga ketika bertanya ke beberapa orang dan tidak mendapatkan hasil yang positif, akhirnya membuat ulang dokumen tersebut melakukan pekerjaan ulang sehingga menghabiskan waktu kerja. (*reinventing the wheel*).

Dengan melihat latar belakang tersebut, salah satu solusi sederhana yang dapat dilakukan adalah dengan mempublikasikan dokumen apa saja yang kita miliki yang sekiranya akan dibutuhkan oleh rekan kerja kita di kemudian hari. Dalam tulisan ini akan di paparkan tahapan pengembangan sebuah *Document Management System* menggunakan *Open Source CMS* berbasis web, yaitu Drupal.

Penelitian ini bertujuan untuk mengembangkan sebuah document management system dengan memanfaatkan content management system drupal yang berbasis open-source.

II. Tinjauan Pustaka

Manajemen dokumen merupakan suatu pendekatan untuk menangani dokumen yang kompleks. Manajemen

dokumen membutuhkan pengaturan semua tahapan dalam mempersiapkan dan menghasilkan dokumen, mengubah, membuat format, penyusunan, pencetakan, distribusi dan penyimpanan [1]. *Document management* merupakan suatu sistem yang bertujuan untuk mengelola dokumen-dokumen dalam bentuk digital dengan berbagai format yang beragam serta tersentralisasi.

Document Management System merupakan penggunaan sistem komputer dan aplikasi untuk menyimpan, mengelola dan melacak dokumen elektronik.

Tujuan utama dari manajemen dokumen adalah untuk menyampaikan nilai yang terkait dengan suatu informasi. Manajemen dokumen adalah tentang bagaimana menghantarkan konten atau informasi dari kontributor kepada konsumernya. Server manajemen dokumen atau repositori menyediakan lingkungan yang terkelola untuk melakukan penyimpanan, kontrol versi, pencarian dan konversi dari dokumen.

Manfaat dari DMS adalah pengelolaan konten dalam bentuk dokumen digital untuk disampaikan secara tepat dari pembuat ke pengguna yang tepat melalui proses yang memenuhi berbagai aturan dan kebijakan yang berlaku, antara lain proses bisnis melalui *workflow*, persetujuan, keamanan, dan pengawasan terhadap perubahan di dalam sistem, secara tersentralisasi melalui arsitektur sistem yang terintegrasi. Hasil dari proses-

proses tersebut adalah terbentuknya sistem yang efektif dan efisien dalam pengelolaan informasi dalam bentuk konten digital dalam organisasi untuk mendukung pengerjaan tugas-tugas dan proses kerja yang dapat meningkatkan kinerja organisasi secara keseluruhan[2].

Fungsi-fungsi DMS yang berkaitan dengan dokumen [3], yaitu:

- *Storage*: DMS memiliki fasilitas penyimpanan yang terpusat, dengan teknologi *database* atau teknologi penyimpanan lainnya.
- *Check in/check out*: dokumen dikelola pada suatu repositori terpusat, untuk memfasilitasi pembuatan dan perubahan dokumen dan untuk menjaga konsistensi versi dokumen dimana mungkin ada banyak pengguna yang dapat mengubah suatu dokumen. *Check out* merupakan proses dimana pengguna mengakses atau mengunduh suatu dokumen dari repositori untuk diubah, sedangkan *check in* digunakan untuk menyimpan atau mengunggah dokumen ini kembali ke repositori setelah dilakukan perubahan. Proses *check in* dan *check out* ini akan menjaga dokumen dari konflik yang mungkin timbul dari isi suatu dokumen dikarenakan adanya perubahan isi dokumen yang bersamaan dari beberapa pengguna.

- *Version Control*: merupakan fasilitas untuk menyimpan tiap versi dari dokumen setiap kali terjadi modifikasi (*check in*). Tiap versi dokumen memiliki identifikasi berupa nomor versi. Fungsi ini penting ketika ada kebutuhan untuk mengakses dokumen versi lama yang telah mengalami beberapa perubahan. Pengguna dapat memilih *check out* dokumen versi terbaru atau versi lama dengan nomor tertentu.
- *Locking*: merupakan proses dimana pengguna dapat mengunci suatu dokumen agar tidak terjadi perubahan atau penghapusan dari pengguna lain. Fitur *locking* sangat bermanfaat untuk menghilangkan intervensi perubahan dari pengguna lain ketika ada keperluan perubahan yang memiliki prioritas tinggi.
- *Rollback/revert*: merupakan fungsi dimana seorang pengguna dapat membatalkan perubahan yang telah dibuat pada dokumen.
- *Audit trail*: merupakan proses pelacakan (*tracking*) informasi sejarah pembuatan dan perubahan dokumen, dengan cara mencatat informasi nama pengguna, tanggal, deskripsi tipe perubahan (*create, edit, delete*).
- *Metadata*: merupakan data tambahan yang bertujuan untuk memberikan deskripsi dokumen, contoh *metadata* antara lain: id pembuat, id pengguna yang menyimpan, tanggal penyimpanan, serta dapat juga ditambahkan kata kunci yang menggambarkan isi dan fungsi dari dokumen. *Metadata* digunakan dalam proses pencarian dan menemukan dokumen.
- *Security*: penerapan kontrol akses terhadap dokumen yang dapat disesuaikan dengan hak dari grup pengguna tertentu, untuk mencegah akses dokumen oleh pengguna yang tidak berhak.
- *Indexing*: merupakan proses membuat suatu daftar *lookup (index)* terhadap kunci-kunci tertentu yang berasal dari *metadata* dokumen ataupun isi dokumen dengan tujuan untuk mempercepat proses pencarian.
- *Searching*: merupakan proses pencarian suatu dokumen.
- *Categorization*: merupakan proses pengelompokan dokumen sesuai dengan struktur dan hirarki kategori yang sesuai dengan proses bisnis organisasi yang bersangkutan.
- *Retrieval*: merupakan metode akses dokumen.
- *Workflow*: merupakan alur proses kerja yang berlaku dalam suatu

organisasi, seperti proses pembuatan dokumen, revisi, persetujuan dan publikasi melalui jalur tertentu seperti *review* dari individu tertentu untuk persetujuan ataupun proses otomatis terhadap suatu aplikasi.

Siklus hidup dari sebuah dokumen terdiri dari:

- *Create*: pembuatan dokumen dengan menggunakan menggunakan aplikasi tertentu dengan format aslinya.
- *Capture*: fungsi yang menyediakan jembatan bagi pengguna untuk memasukkan dokumen ke dalam DMS
- *Store*: setelah dokumen dimasukkan, DMS menjalankan fungsi penyimpanan ke dalam repositori.
- *Version*: untuk menjaga keakuratan informasi yang terkandung dalam dokumen, kontrol versi dilakukan sehingga riwayat dan pembaruan data dapat terjamin.
- *Index*: fungsi *indexing* untuk membentuk proses pencarian yang efektif terhadap dokumen.
- *Manage*: tata kelola konten yang baik dan terstruktur untuk menjamin keandalan pengelolaan dokumen.
- *Cleanse*: fungsi pembersihan terhadap dokumen-dokumen yang berifat sampah dan tidak bermanfaat

lagi bagi organisasi.

- *Distribute*: dengan adanya kontrol keamanan yang baik, penyebaran dokumen dapat dilakukan kepada pihak yang membutuhkan sesuai dengan hak aksesnya masing-masing.
- *Publish*: memberikan kemudahan bagi kontributor konten untuk menampilkan dokumen pada halaman web secara mudah dan cepat.
- *Search*: fungsi standar pencarian untuk memudahkan dalam menemukan suatu informasi yang dibutuhkan secara cepat dan tepat.
- *Retain/Destroy*: retensi memberikan fungsi untuk mempertahankan suatu dokumen berdasarkan jadwal yang dapat diatur sesuai kebutuhan, yang selanjutnya akan diproses untuk penghancuran item konten apabila masa retensi telah berakhir.

Aktivitas yang terjadi dalam suatu DMS antara lain[4]:

- **Managing Document Images**
Proses manajemen dokumen tidak hanya mampu untuk menciptakan, merevisi dan mendistribusikan dokumen elektronik, tetapi juga mampu untuk mengelola semua dokumen secara komprehensif, termasuk manajemen dokumen gambar. Data gambar sangat

berbeda dari data komputer lainnya. Dokumen gambar merupakan koleksi sederhana yang dapat dengan mudah dilihat mata manusia tetapi tidak bagi pengguna sistem komputer. Pada dokumen gambar, kata atau frase tidak dapat ditelusuri, dieja, diubah, dihapus atau disusun kembali elemennya. Selain itu, ukuran file yang lebih besar daripada ukuran file data komputer.

- Document Capture

Capture merupakan solusi yang penting bagi manajemen dokumen dimana proses *capture* merespon aplikasi manajemen pada data gambar dengan bentuk proses OCR/ICR dan *raster-to-vector* untuk mengkonversi gambar.

- Document Storage

Fasilitas untuk mengelola penyimpanan menggunakan beragam alat termasuk WORM (*write once read many*) *optical disk*, *optical jukebox*, RAID (*redundant array of independent disk*) dan/atau penyimpanan tape otomatis.

- Communications

DMS membutuhkan infrastruktur komunikasi untuk aktivitas kerja sama dalam *workgroup* dan juga untuk kepentingan publikasi elektronik.

- Utilizing Workflow Technology

Aplikasi *workflow* dapat dalam bentuk email atau *server-based*.

- Document and Content Management on the Web

Kegiatan manajemen dokumen secara mendasar dirasakan pada internet, web dan intranet. Hal ini berdampak pada platform DMS, kemampuan *capture*, akses dan penyimpanan informasi melalui web.

CMS merupakan piranti lunak yang mengakomodasi pembuatan dan manajemen informasi dalam bentuk teks, gambar, dokumen untuk ditampilkan dalam bentuk website. CMS merupakan perpaduan antara basis data, sistem file, dan modul piranti lunak yang dapat digunakan untuk menyimpan dan menampilkan data dan informasi. Spesifikasi dasar dari sebuah CMS adalah sebagai berikut:

- *Content management*: adalah bagian terpenting dari sistem yang berfungsi sebagai perantara antara pengguna dan sistem informasi dan basis data untuk melakukan manajemen data dan informasi, baik itu untuk menambah, mengubah atau menghapus data dan informasi dari *databases* serta menampilkannya.
- *User authentication*: merupakan bagian dari CMS yang membatasi akses pengguna, sehingga hanya pengguna yang terdaftar saja

yang dapat melakukan aktivitas manajemen konten pada CMS sesuai dengan yang telah ditentukan oleh administrator.

- *Theme*: Merupakan bagian terdepan yang nantinya akan dilihat oleh pengguna akhir, pada bagian ini layout penempatan konten, warna dan rancangan grafis dilakukan.

Secara sederhana CMS dapat diartikan sebagai sebuah sistem yang memberikan kemudahan bagi para penggunanya dalam membuat dan mengelola konten dari website secara dinamis tanpa memerlukan pengetahuan yang bersifat teknis. CMS juga memisahkan antara bagian konten dan tampilan desain (*theme*) sehingga konsistensi tampilan dapat terjaga dengan baik.

Website yang menggunakan CMS lebih berorientasi kepada konten. Pada umumnya sebuah CMS memiliki dua bagian, yaitu *front-end* dan *back-end*. *Front-end* merupakan bagian yang dapat diakses oleh semua orang tanpa perlu melakukan *login* atau registrasi terlebih dahulu, pada bagian ini terdapat semua konten yang bersifat publik dan dapat diakses oleh siapa saja. Sedangkan *back-end* merupakan bagian administrasi website yang mengharuskan pengguna untuk *login* terlebih dahulu. Setelah melakukan *login* pengguna dapat melakukan manajemen konten.

CMS memiliki beberapa manfaat, antara lain:

- Memudahkan proses pengelolaan, unggah, dan penggabungan file data pada suatu organisasi.
- Menjadi media komunikasi dengan *messaging system* yang terintegrasi.
- Menampilkan dan mengatur isi konten untuk dipublikasikan melalui internet.

Pada perusahaan atau organisasi CMS dapat dimanfaatkan untuk menyimpan dan mengatur dokumen elektronik sehingga para karyawan dapat menggunakan informasi tersebut berulang kali jika diperlukan. Dengan teknologi web, informasi juga dapat disebarluaskan tanpa batasan ruang dan waktu.

Tujuan utama dari CMS adalah untuk memudahkan para pembuat konten, terutama mereka yang tidak memiliki kemampuan teknis mengenai pembuatan web agar tetap dapat produktif dalam mempublikasikan konten dengan media web. Tingkatan pengguna CMS dibagi menjadi tiga bagian, yaitu:

- *Anonymous user*: merupakan tingkatan pengguna yang paling dasar. Pengguna pada tingkatan ini dapat mengakses konten-konten yang memiliki hak baca publik. Pengguna pada tingkatan ini pada umumnya hanya dapat melihat konten tanpa dapat melakukan

perubahan apapun terhadap konten tersebut.

- *Authenticated user*: merupakan pengguna yang memiliki *login* dan dapat masuk ke dalam CMS untuk melakukan manajemen konten miliknya sendiri. Pengguna pada tingkatan ini biasanya dapat mengakses lebih banyak konten, termasuk konten-konten yang hanya diperuntukkan bagi pengguna yang sudah *login*.
- *Administrator*: Merupakan pengguna yang memiliki hak akses penuh terhadap semua fungsional CMS. Administrator dapat melakukan pengelolaan pengguna, menambah, mengubah atau menghapus pengguna, memberikan hak akses, mengelompokkan pengguna. Selain itu semua konten juga dapat di akses dan dikelola oleh pengguna pada tingkatan ini. Administrator juga dapat melakukan pengelolaan CMS secara teknis seperti mengaktifkan fitur-fitur tertentu, mengubah tampilan, melakukan instalasi modul tambahan.

Drupal merupakan *Web Content Management System* berbasis *Open Source* yang dirancang untuk dapat menghasilkan website dari skala website sederhana seperti *blog* pribadi hingga website terdistribusi yang kompleks. Dengan melihat tujuan ini maka Drupal

juga dirancang untuk dapat berjalan pada teknologi web yang populer digunakan. Drupal ditulis menggunakan Bahasa pemrograman PHP dan menggunakan databaseMySQL, PostgreSQL atau SQLite[5]. Dari perpaduan ini maka dapat dilihat bahwa Drupal dapat berjalan pada berbagai jenis webserver yang mendukung PHP seperti Apache, Nginx, lighttpd, LiteSpeed atau Microsoft IIS. Sedangkan untuk sistem operasi juga tidak terbatas pada teknologi tertentu, Drupal dapat berjalan pada semua jenis sistem operasi yang mendukung PHP, seperti Linux, Windows, Mac OS, BSD, Solaris.

Drupal sendiri memberikan panduan syarat minimal sebuah sistem untuk menjalankan Drupal, yaitu [2]: ruang penyimpanan sebesar 60 MB, webserverApache, Nginx atau Microsoft IIS, databaseMySQL 5.0.15, PostgreSQL 8.3 atau SQLite 3.3.7 (Drupal 7), dan PHP versi 5.3 (Drupal 7)[6].

Secara umum Drupal dapat dilihat dalam beberapa bagian, yaitu: *Node*, *Comment*, *Taxonomy*, *User*, *Module*, *Regions & Blocks*, *Menus*, *Theme*, *Views*[7][8]:

A. Node

Node merupakan terminologi generik yang digunakan untuk semua konten yang ada dalam website drupal. *Node* dapat memiliki *field* yang berbeda-beda, tergantung jenis *node* yang disebut juga dengan *content type*.

B. Comment

Comment atau komentar merupakan jenis konten yang memiliki keterkaitan dengan salah satu *node* yang ada.

C. Taxonomy

Drupal memiliki sistem untuk mengklasifikasikan konten yang disebut dengan *taxonomy*. Disini pengembang website dapat menambahkan *terms* dan kemudian mengelompokkan *terms* tersebut pada *vocabulary* tertentu. Setiap *vocabulary* kemudian dapat dikaitkan pada satu atau lebih *contenttype*.

D. User

User adalah representasi dari pengguna website. Secara default user memiliki sekumpulan *property* meliputi *username*, *password*, *role* dan alamat email. Pengembang website juga dapat menambahkan *property* lain terhadap *user* melalui modul.

E. Module

Sebuah *module* adalah kode yang menambahkan fungsionalitas Drupal. Terdapat tiga jenis *module*, yaitu *Coremodules* yang merupakan modul utama yang terdapat dalam paket instalasi Drupal, *Contributedmodules* adalah modul-modul yang terdapat pada halaman pengunduhan *module* pada website drupal dan *Custom modules* adalah *module* yang dibuat sendiri oleh pengembang website.

F. Regions & Blocks

Halaman pada website Drupal

terbagi-bagi menjadi beberapa *regions*, antara lain *header*, *footer*, *sidebars* dan *content*. *Regions* ditentukan dari *theme* yang ada. Sedangkan *blocks* merupakan potongan konten yang ditampilkan pada *regions* dari halaman web. *Blocks* dapat berupa konten HTML statis, menu atau navigasi website, hasil *output* dari *module* atau konten dinamis yang dibuat oleh pengembang website.

G. Menu

Pada Drupal 7 terdapat empat menu standar, yaitu Main Menu untuk menampilkan kontenweb, Management berisi sekumpulan *link* untuk administrasi web, *Navigation* berisi *link* yang dihasilkan oleh *module* dan *UserMenu* berisi *link* untuk melihat akun *user* dan juga untuk *log out*.

H. Theme

Theme merupakan bagian yang terpisah dari sistem utama Drupal. *Theme* mengontrol tampilan dari website, berisi sekumpulan *filetemplate* PHP dan file CSS yang menentukan layout, *font*, warna dan *style* lainnya dari halaman web.

I. Views

Tidak semua website menggunakan *moduleviews*, namun sebagian besar website menggunakan *views* untuk melakukan modifikasi terhadap tampilan web. Dengan menggunakan *views*, pengembang web dapat memilih konten mana saja yang mau ditampilkan, bagaimana urutan konten yang di

tampilkan, bagaimana menampilkan konten dan lain sebagainya.

III. Metodologi Penelitian

Penelitian ini dilakukan dengan melakukan studi literatur mengenai document management system dan fitur-fitur yang diperlukan, serta CMS drupal dan modul-modul yang dapat digunakan untuk menghasilkan fungsionalitas *document management system*. Kemudian dilakukan instalasi pada lingkungan pengembangan lokal dan dilakukan modifikasi untuk menghasilkan sebuah *document management system*.

IV. Perancangan dan Implementasi

Dalam pengembangan web menggunakan Drupal, direktori file yang akan banyak digunakan adalah direktori “*sites*” yang didalamnya terdapat dua sub-direktori, yaitu “*all*” dan “*default*”. Direktori “*all*” merupakan tempat untuk meletakkan semua *modules*, *themes* dan *libraries* yang akan digunakan untuk menambah dan mengembangkan fungsionalitas dan tampilan dari website. Secara sederhana, masing-masing direktori dan file yang ada pada direktori “*sites*” dijelaskan sebagai berikut:

Direktori *Modules* (/sites/all/modules), merupakan tempat untuk meletakkan semua *modules*, yang kemudian dapat di aktivasi melalui halaman administrasi web setelah *login*.

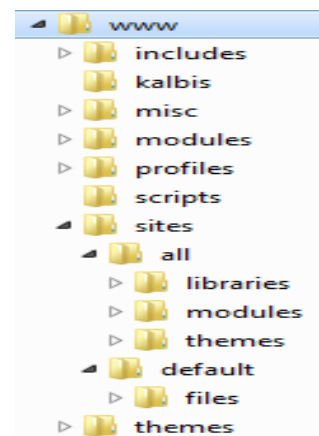
Direktori *Themes* (/sites/all/themes), Merupakan tempat untuk meletakkan *themes*, baik yang diunduh dari website Drupal, maupun yang dikembangkan sendiri.

Direktori *Libraries* (/sites/all/libraries), merupakan tempat untuk meletakkan *library* tambahan yang dibuat diluar drupal, untuk dikombinasikan dengan *modules*.

Direktori *Files* (/sites/default/files), merupakan tempat yang akan digunakan Drupal untuk menyimpan semua file konten yang di unggah oleh *user*.

Filesettings.php (/sites/default/settings.php) merupakan file yang berisi informasi konfigurasi website, seperti koneksi database.

Diluar direktori “*sites*”, semua direktori lainnya merupakan file utama dari instalasi Drupal yang diperlukan agar website dapat bekerja dengan baik, sehingga sangat jarang diperlukan untuk melakukan modifikasi, karena semua modifikasi dapat dilakukan melalui *modules*.



Gambar 1. Struktur Direktori Drupal

Dalam proses pengembangan *document management system*, perlu dilakukan analisis komponen tambahan apa saja yang diperlukan. Berikut ini merupakan tahapan yang dilakukan dalam pembuatan *document management system*.

A. Instalasi Drupal pada lingkungan pengembangan

Dalam pembuatan *document management system* ini, lingkungan pengembangan yang digunakan adalah:

Sistem operasi: Windows 7 Pro

Server: Apache 2.4.9

Database: MySQL 5.5.36

PHP: php-5.4.27

Instalasi Drupal menggunakan Drupal versi 7.27 yang di-unduh dari website Drupal. Untuk melakukan instalasi Drupal, database harus di-*setup* terlebih dahulu.

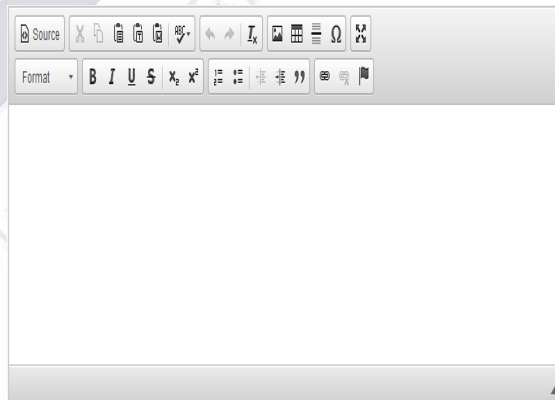
B. Instalasi dan konfigurasi *module* tambahan

Instalasi standar Drupal sudah meliputi beberapa modul dasar yang dapat digunakan untuk membangun sebuah website, namun untuk dapat meningkatkan fungsionalitas dari Drupal, perlu dilakukan instalasi beberapa modul tambahan. Dalam pengembangan *document management system*, beberapa *module* yang digunakan, yaitu: *ckeditor*, *libraries*, *views*.

Modul *libraries* merupakan modul yang menghubungkan modul tambahan jika memerlukan *library* dari

luar. Salah satu modul yang menggunakan fungsionalitas dari *libraries* adalah *ckeditor*, dimana modul *ckeditor* membutuhkan *libraryJavaScriptckeditor*.

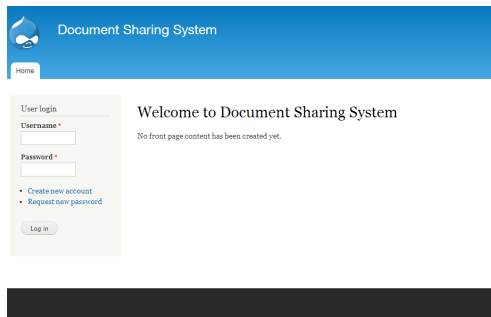
Modul *ckeditor* merupakan modul yang menambahkan panel *rich text editor* pada *textarea* seperti terlihat pada Gambar 2. Dengan *ckeditor*, *user* dapat melakukan modifikasi teks dengan mudah tanpa harus mengetikkan kode HTML. Untuk meningkatkan fungsionalitas *ckeditor*, dapat ditambahkan modul *imce* yang berfungsi sebagai *filebrowser* yang dapat digunakan untuk melakukan *browsing* file atau gambar di server.



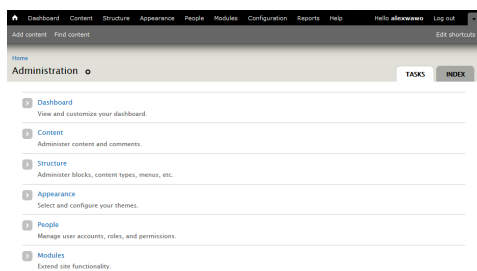
Gambar 2. *Textarea* dengan *ckeditor*

C. Instalasi dan konfigurasi *theme*

Instalasi Drupal standar menyertakan *theme* “Bartik” pada Gambar 3 sebagai *theme* untuk *front-end* dan *theme* “Seven” pada Gambar 4 untuk halaman admin. Instalasi *theme* baru dapat dilakukan jika ingin merubah tampilan dari tampilan standar Drupal dengan tampilan *theme* lain yang dapat di-unduh dari website Drupal, maupun *theme* yang dikembangkan sendiri.

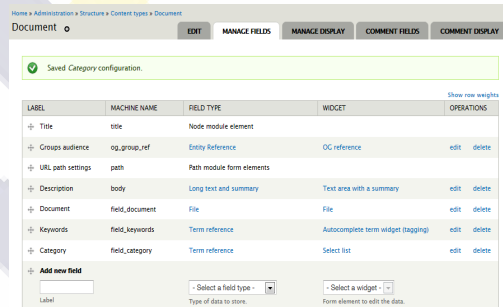


Gambar 3. Tampilan ThemeBartik



Gambar 4. Tampilan Theme Seven

system adalah bagaimana menyajikan kumpulan dokumen yang sudah di-unggah oleh *user* sehingga dapat dengan mudah diakses oleh *user* lain. Untuk kebutuhan ini, *module views* dapat melakukan filterisasi, konten mana yang mau ditampilkan sesuai dengan *content type* atau sesuai dengan filter *taxonomy* tertentu.



Gambar 5. Konfigurasi ContentType

D. Konfigurasi Content Type

Fungsionalitas utama dari *document management system* adalah kemampuan untuk melakukan unggah dan unduhfile atau dokumen. Untuk mendapatkan fungsionalitas ini, maka dibuat sebuah *content type* baru pada Drupal, dengan nama “*Document*”.

Ketika membuat *Content Type* baru, Drupal sudah menyediakan *field* “*Title*” untuk judul dari konten yang dibuat, dalam hal ini dokumen yang di-unggah, dan “*Description*” untuk keterangan atau deskripsi dari konten atau dokumen yang di-unggah. Fitur unggahfile dapat ditambahkan dengan menambahkan *field* dengan tipe “*File*”.

E. Konfigurasi Module Views

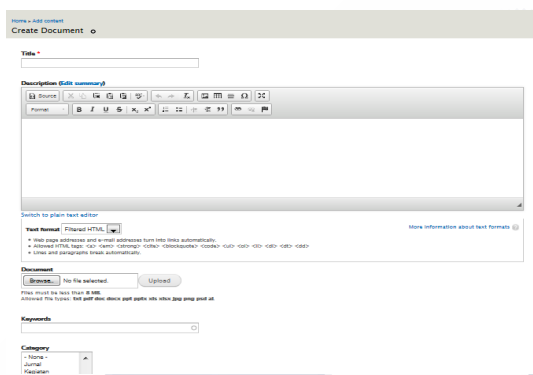
Tahapan terakhir dalam mengembangkan *document management*

V. Hasil dan Pembahasan

Dengan fitur manajemen *user* yang sudah tersedia pada CMS Drupal, digabungkan dengan konfigurasi *Content Type* “*Document*”, *user* yang sudah terdaftar dapat melakukan unggah dokumen dengan disertai dengan keterangan berupa deskripsi, kata kunci dan kategori. Tampilan ketika membuat konten dokumen baru dapat terlihat pada Gambar 6.

Setelah dokumen dibuat, maka akan ditampilkan pada halaman daftar dokumen yang dihasilkan melalui konfigurasi *views* yang telah dilakukan sebelumnya. Tampilan halaman daftar dokumen berupa tabel, dengan kolom Judul, Deskripsi, *Keyword* dan *Download*. Pada bagian samping dari halaman daftar dokumen

terdapat daftar *link* kategori dokumen, sehingga memudahkan *user* ketika melakukan pencarian dokumen tertentu sesuai dengan kategorinya. Dibagian samping halaman daftar dokumen juga disediakan formulir pencarian, yang akan melakukan pencarian sesuai dengan kata kunci yang diketikkan. Halaman daftar dokumen dapat dilihat pada Gambar 7.



Gambar 6. Membuat Dokumen Baru



Gambar 7. Halaman Daftar Dokumen

Dengan fitur manajemen *user* juga dapat dilakukan pengelompokan *user*, sehingga dapat ditentukan *user* mana saja yang dapat membuat atau mengunggah dokumen baru, *user* mana yang hanya dapat melihat dokumen.

VI. Kesimpulan

Pembuatan *document management system* dengan menggunakan CMS Drupal dapat dilakukan dengan baik. Adapun hasil yang didapatkan adalah sebuah *document management system* yang terintegrasi dengan manajemen *user* dari Drupal, sehingga dapat dilakukan pengelompokan *user* yang dapat melakukan unggah dokumen dan *user* yang hanya dapat melihat dan unduh dokumen.

Secara umum, modul tambahan yang digunakan berfungsi untuk meningkatkan usability dari sistem yang dibangun. Instalasi Drupal dengan konfigurasi *Content Type* sudah dapat membangun sebuah *Document Management System* sangat sederhana. Untuk kedepannya diharapkan *document management system* ini dapat dikembangkan lagi sehingga dapat melakukan pengelompokan dokumen berdasarkan departemen dalam suatu organisasi.

V. Daftar Pustaka

- [1] E. A. Regan dan B. N. O'Connor, *End-User Information Systems: Implementing Individual and Work Group Technologies*, New Jersey: Prentice Hall, 2001.
- [2] M. J. D. Sutton, *Document Management for the Enterprise: Principles, Techniques, and Applications*, New York: Wiley, 1996.

- [3] A. Adam, Implementing Electronic Document and Record Management Systems, NW: Auerbach Publications, 2008.
- [4] M. F. Robek, G. F. Brown dan D. O. Stephens, Information and records management: document-based information systems, Michigan: McGraw-Hill, 1995.
- [5] T. Tomlinson dan J. K. VanDyk, Pro Drupal 7 Development, New York: Apress, 2010.
- [6] "Drupal," [Online]. Available: <http://drupal.org/requirements>. [Diakses 27 April 2014].
- [7] "Drupal General Concepts," [Online]. Available: <https://drupal.org/node/19828>. [Diakses 27 April 2014].
- [8] X. Cheng dan Z. Wang, "The Web Development Based on the Drupal System," Second International Conference on Business Computing and Global Informatization, 2012.