

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1. Tinjauan Pustaka**

Penelitian yang berkaitan dengan pembuatan *Institutional Repositories* dengan berbagai macam *platform* sudah beberapa kali dilakukan. Hendryanto Djohan *et al.* (2010) pada penelitiannya mengatakan bahwa DSpace telah digunakan oleh sembilan ratus perpustakaan perguruan tinggi terkemuka di dunia seperti: Harvard (US), MIT (US), National University of Singapore (Singapore), dan ratusan universitas besar lainnya di seluruh dunia. Selain itu, hasil penelitian DSpace memiliki relevansi yang tinggi untuk dapat diimplementasikan di perpustakaan Johannes Oentoro – UPH yang juga merupakan sebuah perpustakaan perguruan tinggi. Selain itu, menurut Tansley *et al.* (2003) DSpace dapat dengan mudah dimodifikasi dan disesuaikan dengan kebutuhan penggunanya. Disamping itu, DSpace juga memiliki tingkat portabilitas yang tinggi, sehingga dapat diimplementasikan pada Linux, Mac OSX ataupun Windows. Bahkan, DSpace juga memiliki kapasitas preservasi yang baik, sehingga sekali suatu objek digital diunggah, maka hampir tidak mungkin objek tersebut dihapuskan lagi.

Berdasarkan kumpulan kelebihan dan relevansinya untuk digunakan pada repositori universitas, maka DSpace dipilih sebagai sistem yang akan dieksplorasi lebih jauh pada penelitian ini.

Selain itu, penelitian lain yang berkaitan dengan pembuatan repositori (Sigit Wahyudi, 2010) membangun sebuah repositori yang bertujuan untuk kebutuhan fungsional Perpustakaan Sains dan Teknologi Universitas Negeri Syarif

Hidayatullah Jakarta dan kesalahan-kesalahan yang sering terjadi pada sistem manual dapat teratasi serta tersimpan dengan baik sehingga mengurangi hilangnya koleksi perpustakaan.

Namun, pembangunan repositori tidak hanya sebatas kebutuhan fungsional saja, akan tetapi bagaimana repositori dapat bermanfaat tidak hanya bagi pengguna tetapi juga bagi universitas. Untuk pengguna, repositori bermanfaat sebagai sumber pengetahuan sementara bagi universitas, selain sebagai tempat pengelolaan artikel ilmiah, juga dapat bermanfaat agar universitas lebih dikenal oleh masyarakat luar.

Selanjutnya penelitian (Miftakhul Yazid Fuadi, 2014) yang membangun repositori dengan aplikasi *open source* lain yaitu aplikasi Eprints. Alasan memilih aplikasi ini karena metadata antara *Ganesha Digital Library* (GDL) dan Eprints sama (dengan harapan dapat mempermudah proses migrasi data).

Akan tetapi, pemilihan penggunaan aplikasi tidak hanya dilihat dari seberapa mudah proses migrasi data yang dilakukan. Aspek-aspek lain seperti dukungan data dan hubungan data juga perlu dipertimbangkan. Berbeda dengan aplikasi DSpace, aplikasi Eprints hanya mengelompokan data objek dalam bidang tertentu misalnya tahun, subjek, divisi dan judul tetapi tidak bisa mengaitkan antara data yang satu dengan data yang lain kecuali menggunakan URL dalam metadata yang spesifik.

Dari bebrapa penelitian yang telah dilakukan, disimpulkan bahwa pembangunan repositori selain memerhatikan kebutuhan secara fungsional juga harus memerhatikan aspek-aspek *usability* dan *user experience* (UX) yang dirasakan oleh pengguna (Garrett, 2011). Selain itu, repositori juga memberikan

manfaat lebih tidak hanya pada pengelolaan artikel tetapi juga pada pengindeksan *website*.

Adapun yang membedakan penelitian ini dengan penelitian sebelumnya (Djohan *et al.*, 2010) terletak pada tahapan penelitian. Pada penelitian sebelumnya, tahapan penelitian menggunakan tiga sudut pandang dan enam pertanyaan interogatif dari *Enterprise Architecture Zachman Framework*, sementara pada penelitian ini digunakan tahapan penelitian dengan metode *Waterfall (Classic Life Cycle)*. Solusi yang digunakan (Fuadi, 2014) juga berbeda. Penelitian Fuadi menggunakan aplikasi Eprints, sementara pada penelitian ini menggunakan aplikasi DSpace.

Selain pada tahapan penelitian dan solusi yang digunakan, perbedaan penelitian ini juga terdapat pada solusi yang ditawarkan dan tujuan pembangunan aplikasi, pada penelitian lain (Sigit Wahyudi, 2010) pembangunan dilakukan dengan pengembangan aplikasi sendiri menggunakan bahasa *php*, sementara pada penelitian ini digunakan aplikasi DSpace. Tujuan penelitian sebelumnya juga terfokus pada kebutuhan fungsional, sementara pada penelitian ini terfokus bagaimana universitas memiliki *website* yang dapat terindeks oleh *search engine*.

## **2.2. Landasan Teori**

### **2.2.1. Repositori**

Saat ini repositori merupakan sumber potensial yang kaya akan informasi, data, gambar dan hasil penelitian yang bermanfaat. Tetapi pergerakannya yang baru dan waktu yang dibutuhkan untuk merencanakan, membuat aturan, dan membawa komunitas institusional kepada konsensus dapat memperlambat prosesnya.

Repositori adalah sistem yang memungkinkan lembaga-lembaga untuk menyimpan dan mengelola dokumen digital mereka serta melakukan interaksi dan kolaborasi antar pengguna dalam satu lembaga. Terdapat beberapa perangkat lunak perpustakaan digital yang tersedia sebagai "open source" maupun sebagai "proprietary format". Perangkat lunak *open source* membantu perpustakaan terutama dalam menurunkan biaya awal dan biaya berkelanjutan, menghilangkan vendor *lock-in* dan memungkinkan fleksibilitas aplikasi yang lebih besar. Keuntungan utama dari perangkat lunak *open source* adalah penggunaannya yang umumnya gratis seperti Dspace.

### **2.2.2. Institutional Repository (IR)**

*Institutional Repository* (IR) adalah ruang penyimpanan berbasis online untuk mengumpulkan, memelihara, dan menyebarkan informasi dalam bentuk digital untuk keluaran intelektual sebuah institusi.

"Sebuah *Institutional Repository* berbasis universitas adalah seperangkat layanan yang ditawarkan oleh universitas kepada masyarakat untuk pengelolaan dan penyebaran materi digital yang dibuat oleh anggota lembaga masyarakat. Hal ini adalah komitmen dasar organisasi untuk pengelolaan materi digital, termasuk pelestarian jangka panjang yang tepat, serta organisasi dan akses atau distribusi" (Clifford A.Lynch, 2003: 1-7).

Sebuah IR dapat berisi karya seorang penulis atau lembaga yang memiliki hak cipta, atau karya yang telah mendapatkan izin untuk disimpan kedalam repositori. Seperti, repositori dapat berisi jurnal dengan persetujuan penulis atau kondisi kontrak kerja. Atau dapat berisi salinan publikasi jurnal yang telah diatur

dengan persetujuan penerbit, dimana penulis dianjurkan oleh lembaga mereka untuk memastikan bahwa perjanjian hak cipta penerbit memungkinkan untuk kemungkinan ini. Oleh karena itu, sebuah IR tidak boleh berisi karya yang belum mendapatkan izin hak cipta ataupun aturan lisensi yang belum dibuat.

Meningkatnya permintaan untuk artikel ilmiah, terutama dalam bidang ilmu pengetahuan akan meningkatkan tekanan pada peneliti dan universitas. Penerbitan digital, jaringan global, penelitian lain, dan peningkatan komunikasi diantara masyarakat mendorong permintaan untuk akses yang lebih luas. Tetapi, dengan terjadinya hal ini, kepopuleran sebuah universitas akan terus meningkat jika IR mereka berbasis web dan terindeks oleh *search engine*.

### **2.2.3. Perangkingan Website**

Perangkingan *website* adalah proses pemeringkatan *website* berdasarkan dengan ketentuan setiap organisasi perangkingan. Perangkingan ini merangking *website* yang telah terindeks. Cara yang paling sering digunakan adalah *Search Engine Optimization* (SEO). SEO adalah serangkaian upaya yang dilakukan oleh seorang *webmaster* pada sebuah *website* yang bertujuan untuk meningkatkan visibilitas/rangking sebuah halaman *website* menjadi lebih baik di *search engine*, terutama Google. Untuk mendapatkan ranking yang tinggi, ditemukan oleh Crawler (suatu program yang dipergunakan untuk mengindeks isi suatu situs) *search engine* saja tidaklah cukup. Ada sejumlah kriteria tertentu yang harus dipenuhi agar *website* kita terdaftar dengan nilai lebih tinggi dari *website-website* lainnya. Beberapa kriteria yang diidentifikasi crawler *search engine* diantaranya:

- Anchor text (teks yang dipergunakan sebagai link menuju halaman / *website* lain.
- Popularitas *website*
- Relevansi link
- Kata kunci
- Bahasa
- Isi secara keseluruhan
- Umur *website*

Sementara itu, teknik peringkatan repositori maupun *website* di dunia sudah banyak diketahui. Tingkat pemeringkatan repositori yang telah diakui didunia yaitu menurut *Webometrics Ranking of World Universities* sementara teknik pemeringkatan *website* yang diakui didunia yaitu menurut *4 International Colleges & Universities (4ICU)*.

#### ***Webometrics Ranking of World Universities (Webometrics)***

*Webometrics* merupakan ilmu yang mempelajari aspek kuantitatif dari konstruksi dan penggunaan sumberdaya informasi, struktur dan teknologi di Web menggunakan pendekatan *bibliometrics* dan *informetrics* (Björneborn & Ingwersen 2004). *Bibliometrics* adalah studi kuantitatif dari unit fisik buku yang dipublikasikan atau unit dari bibliografi. *Informetrics* adalah studi tentang aspek kuantitatif informasi dalam bentuk apapun atau dengan kata lain tidak harus dalam bentuk buku atau jurnal.

*World Universities ranking on the Web* merupakan inisiatif dari *Cybermetrics Lab*, grup riset dari *Centro de Ciencias Humanas y Sociales (CCHS)*

bagian dari *National Research Council (CSIC)* yang merupakan badan riset publik terbesar di Spanyol. *Webometrics Ranking of World Universities* pertama kali dipublikasikan pada tahun 2004, dan diperbarui setiap enam bulan. Data dikumpulkan pada Januari dan Juli tiap tahun dan dipublikasikan peringkat terbaru setelah satu bulan kemudian.

Aktivitas universitas sangat kompleks dan multidimensi. Aktivitas universitas tersebut dapat dicerminkan melalui keberadaan situs web universitas tersebut. Pemingkatan yang dibuat oleh *Cybermetrics Lab* dibangun dengan menggabungkan indikator-indikator yang mengukur aspek-aspek yang berbeda (Webometrics 2010).

Indikator-indikator Webometrics yang benar-benar dipertimbangkan adalah total publikasi elektronik di situs utama universitas (*size*) dan *visibility* halaman web menurut banyaknya *external inlink* yang mereka terima dari situs web lain (*impact*). Kedua variabel tersebut memiliki rasio 1:1. Aspek lain seperti desain, aksesibilitas, banyaknya pengunjung tidak termasuk aspek yang diperhitungkan dalam Webometrics. Komponen *size* dipecah menjadi tiga variabel yaitu banyaknya halaman web (20%), banyaknya dokumen dengan format pdf, doc, ps, dan ppt (15%), dan banyaknya *paper* atau jurnal yang ada di Google Scholar (15%). Sedangkan 50% sisanya untuk banyaknya sitasi (*site link citations*) (Aguillo 2009). Ada empat indikator yang menjadi ukuran suatu situs antara lain sebagai berikut:

- *Size (S)*. Indikator ini merupakan banyaknya halaman yang terdapat di empat *search engine*, yaitu Google.

- *Visibility (V)*. Indikator ini menunjukkan total *unique external link* yang masuk (*inlinks*) yang didapatkan dari *Google Search*,
- *Rich File (R)*. Total publikasi akademik dalam *file* dengan format Adobe Acrobat (.pdf), Adobe PostScript (.ps), Microsoft Word (.doc) dan Microsoft Powerpoint (.ppt),
- *Scholar (Sc)*. Banyaknya *paper* dan sitasi untuk masing-masing domain yang terdapat di *Google Scholar*.

Setelah diperoleh *score* untuk masing-masing situs, situs diurutkan secara menurun berdasarkan *score* tersebut sehingga diperoleh peringkat untuk masing-masing situs.

#### ***International Colleges & Universities (4ICU)***

4ICU adalah suatu *search engine* dan direktori yang melakukan penilaian berdasarkan kepopuleran situs yang dimiliki oleh 11.000 perguruan tinggi di seluruh dunia yang telah terakreditasi dan tersebar dalam 200 negara.

Dalam hal ini, yang dimaksud sebagai perguruan tinggi yang terakreditasi adalah sebagai berikut:

- Telah resmi diakui, memiliki ijin atau telah diakreditasi oleh badan-badan nasional atau regional seperti kementerian pendidikan atau organisasi pendidikan tinggi yang berwenang.
- Berhak memberikan gelar sarjana atau pascasarjana dalam empat tahun.
- Melakukan kegiatan pembelajaran secara umum dengan bertatap muka, program dan kursus.



## Metodologi Penilaian 4ICU

Dalam melakukan penilaian, ada kebijakan khusus yang dikemukakan oleh 4ICU. Mengingat yang dinilai oleh 4ICU adalah kepopuleran alamat situs resmi milik perguruan tinggi, maka 4ICU tidak memberitahukan cara penilaian yang dilakukannya guna menghindari kecurangan yang dilakukan *webmaster* perguruan tinggi dalam meningkatkan peringkat perguruan tinggi masing-masing.

Namun meskipun begitu, 4ICU memberitahukan parameter acuan yang mereka gunakan dalam menilai, yaitu sebagai berikut :

- *Google Page Rank*
- *Alexa Traffic Rank*
- *Majestic SEO (Referring Domain, Citation Flow, Citation Trust)*

## Langkah Penilaian 4ICU

- 1) Pengumpulan data *metric* dilakukan dalam hari yang sama untuk menghindari perubahan data yang tidak tentu serta memaksimalkan perbandingan data
- 2) Penggunaan filter untuk mengidentifikasi keberadaan data yang nilainya sangat jauh dari data lain yang ada (*data outliers*)
- 3) *Review* data dari *Alexa Traffic Rank*, untuk menyeleksi perguruan tinggi yang masih menggunakan subdomain sebagai halaman utama yang resmi
- 4) Data *webmetric* kemudian dinormalisasikan pada skala 0-100 dengan mempertimbangkan perhitungan logaritma alami dari *Google Page Rank* dan *Alexa Traffic Rank*, serta hasil normalisasi dari data *Majestic SEO*

- 5) Nilai normalisasi dari ketiga data tersebut digabungkan dan kemudian dirata-rata untuk menghasilkan nilai akhir dan peringkat perguruan tinggi

#### **2.2.4. DSpace**

DSpace ([www.DSpace.org](http://www.DSpace.org)) adalah aplikasi repositori digital, dibuat sebagai proyek bersama oleh Perpustakaan MIT dan Perusahaan Hewlett-Packard, dan dirilis ke publik pada bulan November 2002 sebagai Aplikasi *Open Source*.

Aplikasi repositori digital tersedia secara bebas sebagai perangkat lunak *open source* dari SourceForge ([www.sourceforge.net/projects/DSpace](http://www.sourceforge.net/projects/DSpace)) dengan lisensi distribusi BSD. Aplikasi *open source* DSpace tersedia untuk siapa saja yang ingin mengunduh dan menggunakan aplikasi pada semua jenis institusi, organisasi, maupun perusahaan (atau bahkan hanya individu). Pengguna juga mengizinkan modifikasi DSpace untuk memenuhi kebutuhan spesifik organisasi. Aturan khusus penggunaan dijelaskan dalam lisensi distribusi *Berkeley Software Distribution* (BSD).

DSpace adalah salah satu *platform* perangkat lunak *open source* untuk menyimpan, mengelola dan mendistribusikan koleksi ke dalam format digital. Seperti halnya konten dunia yang sekarang dikembangkan dan disebarluaskan dalam format digital, aplikasi DSpace mendukung generasi pengarsipan digital yang lebih permanen dan dapat di bagikan dibanding dengan pengarsipan analog saat ini. DSpace dapat mendukung berbagai macam artefak, termasuk buku, tesis, dan scan 3D digital seperti object, film fotografi, video, kumpulan data penelitian dan segala bentuk konten lainnya.

DSpace dikembangkan dalam menanggapi kebutuhan universitas akan aplikasi yang mudah digunakan, memiliki layanan yang handal dan dapat dikelola, sebagai *host*, serta memelihara dan mendistribusikan material universitas dalam format digital. Menawarkan keuntungan dan kemudahan dalam penyimpanan dan penyebaran berbasis web. DSpace dapat menampung berbagai *file* seperti; dokumen, dataset, gambar dalam format seperti; txt, pdf, doc, dan jpg. DSpace mengatur dan mendistribusikan materi digital, terdiri dari *file* digital (atau “*bitstream*”) dan memungkinkan untuk membuat, mengindeks, dan mencari metadata terkait untuk mencari dan mengambil materi. Hal ini juga dirancang untuk mendukung pelestarian jangka panjang dari materi digital yang disimpan dalam repositori.

DSpace menyediakan cara untuk mengelola bahan penelitian dan publikasi dengan cara pengelolaan profesional untuk memberikan pengguna aksesibilitas yang lebih besar dari waktu ke waktu.

### **Alasan Menggunakan DSpace**

Berikut alasan menggunakan aplikasi DSpace:

#### **a. Komunitas Pengguna dan Pengembang Terbesar di Seluruh Dunia**

DSpace memiliki lebih dari 1000 lembaga yang saat ini menggunakan aplikasi DSpace dalam organisasi mereka dalam lingkungan produksi ataupun proyek. Penggunaan yang paling umum adalah dengan perpustakaan ilmiah sebagai IR, namun terdapat banyak organisasi yang menggunakan aplikasi DSpace untuk mengelola data digital untuk proyek, subjek repositori, arsip webm dan dataset repositori.

### **b. Dapat Mengindeks Website**

DSpace dilengkapi *tools* yang dapat memastikan *search engine* (Google, Bing, Yahoo, Google Scholar) dapat dengan mudah dan efektif mengindeks semua konten *website*. Namun, *tools* ini juga menyediakan konfigurasi dasar yang dapat dilakukan. Berikut cara agar *website* terindeks secara optimal:

- Menjaga DSpace tetap *up to date* (DuraSpace akan tetap menambahkan perbaikan pengindeksan pada DSpace versi terbaru)
- Pastikan DSpace terlihat oleh *search engine*
- Mengaktifkan fitur sitemap (dapat dilakukan dengan mendaftar pada *Google Webmaster Tools*)
- Pastikan *robots.txt* memungkinkan akses ke *item* halaman "*splash*" dan *full text*
- Pastikan item metadata muncul di header HTML dengan benar
- Hindari mengarahkan download *file* ke halaman *landing item*
- Nonaktifkan setiap generasi halaman sampul PDF

### **c. Dapat Diubah Sesuai Kebutuhan**

Beberapa kunci yang dapat di ubah pada aplikasi DSpace sesuai kebutuhan seseorang adalah sebagai berikut:

**Pengaturan tema *website*** - Pengguna dapat menyesuaikan tampilan DSpace mereka sesuai dengan kebutuhan sehingga mudah untuk digunakan. Hal ini dapat dilakukan dengan menggunakan ekstensi Manakin, yang sekarang ini telah tersedia dalam DSpace versi 5.x.

**Kemampuan mengatur metadata** – *dublin core* adalah format standar metadata dalam aplikasi DSpace, namun salah satunya dapat menambahkan atau mengubah bagian apapun untuk disesuaikan pada aplikasi.

**Kemampuan mengatur browser dan pencarian** - Dapat memutuskan bidang apa yang ingin ditampilkan pada situs DSpace, seperti penulis, judul, tanggal, dll. Selain itu juga dapat memilih bidang metadata apa yang ingin disertakan dalam antarmuka pencarian. Jika diinginkan, semua teks yang terdapat didalam artikel maupun metadata terkait, dapat diindeks untuk pencarian teks.

**Database yang mudah diatur** - Dapat memilih database yang akan digunakan, baik itu Postgres maupun Oracle, untuk mengelola item maupun metadata.

**Kemampuan untuk memilih standar bahasa** - Aplikasi situs DSpace tersedia dalam duapuluh lebih bahasa.

#### **d. Dapat Mengelola dan Memelihara Semua Jenis Konten Digital**

Aplikasi DSpace dapat mengenali dan mengelola *file* dengan segala bentuk format *file* dan tipe MIME. Beberapa format *file* yang paling umum saat ini dikelola dalam lingkungan DSpace adalah PDF, Word, JPG, MPG dan TIFF. Penerimaan *file* dimasa depan akan dilakukan dan di validasi oleh *Global Digital Format Registry* (GDFR) dan aplikasi JHOME.

Sistem ini menyediakan pemeriksaan integritas *bit* melalui laporan *checksum* MD5. Sejarah sistem memberikan jejak audit dari semua perubahan ke item dan metadata terkait.

**e. Digunakan oleh Lembaga Pendidikan, Pemerintahan, Swasta dan Komersial**

*Platform DSpace* tidak hanya digunakan oleh lembaga pendidikan tinggi, dimana platform tersebut hanya dikembangkan untuk lembaga terkait, tetapi juga dikembangkan menjadi lebih luas sehingga memiliki daya tarik sendiri. Seperti museum, arsip negara, repositori jurnal, konsorsium, dan perusahaan komersial juga telah menggunakan perangkat lunak ini untuk mengelola aset digital mereka.

**f. Dapat Dibangun Dimana Saja**

*DSpace* dilengkapi dengan antarmuka berbasis web yang mudah dikonfigurasi, dimana setiap administrator dapat membangun *DSpace* pada Linux maupun Windows.

### **2.2.5. Teknologi Pengembangan Aplikasi**

#### ***United Markup Language (UML)***

*United Markup Language (UML)* merupakan bahasa pemodelan secara grafis yang digunakan untuk menspesifikasikan, memvisualisasikan, membangun, dan mendokumentasikan seluruh rancangan aplikasi perangkat lunak. Penggunaan model UML bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem di dalam aplikasi. Model UML yang digunakan dalam pengembangan Repositori UMY adalah *Use Case Diagram* dan *Activity Diagram*.

**a. *Use Case Diagram***

*Use Case Diagram* mendeskripsikan sebuah interaksi antara satu atau lebih *actor* dengan sistem yang dibuat. Dapat dikatakan *Use Case* digunakan untuk

mengetahui fungsi yang ada di dalam sistem dan siapa saja yang berhak menggunakan fungsi-fungsi yang dibutuhkan.

b. *Activity Diagram*

*Activity Diagram* merupakan diagram yang digunakan untuk menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis.

### **Basis Data PostgreSQL**

Basis data merupakan salah satu teknologi yang harus dimiliki sebuah perusahaan, institusi, ataupun organisasi adalah teknologi yang dapat memproses data. Dalam Repositori UMY, digunakan basis data PostgreSQL.

PostgreSQL merupakan sebuah *Object-Relational Database Management System* (ORDBMS) berdasarkan pada PostgreSQL Versi 4.2 yang dikembangkan di Universitas California pada Berkeley Computer Science Department.

PostgreSQL merupakan DBMS yang *open source* yang mendukung bahasa SQL secara luas dan menawarkan beberapa fitur-fitur modern seperti:

- Complex Queries
- Foreign Keys
- Triggers
- Views
- Transactional Integrity
- Multiversion Concurrency Control

Selain itu, PostgreSQL telah mendukung teknologi lama dengan menambahkan fitur-fitur baru pada:

- Data Types
- Functions
- Operators
- Aggregate Functions
- Index Methods
- Procedural Languages

SQL di Postgres tidaklah seperti yang kita temui pada RDBMS umumnya. Perbedaan penting antara Postgres dengan sistem relasional standar adalah arsitektur Postgres yang memungkinkan user untuk mendefinisikan sendiri SQL-nya, terutama pada pembuatan *function* atau biasa disebut sebagai *stored procedure*. Hal ini dimungkinkan karena informasi yang disimpan oleh Postgres bukan hanya table dan kolom, melainkan tipe, fungsi, metode akses, dan banyak lagi yang terkait dengan table dan kolom tersebut. Semuanya terhimpun dalam bentuk *class* yang bisa diubah user. Arsitektur yang menggunakan class ini lazim disebut sebagai *object oriented*. Karena Postgres bekerja dengan class, berarti Postgres lebih mudah dikembangkan di tingkat user, dan bisa mendefinisikan sebuah table sebagai turunan dari table lain. Sebagai perbandingan bahwa system basis data konvensional hanya dapat diperluas dengan mengubah *source codenya*, atau menggunakan modul tambahan yang ditulis khusus oleh vendor, maka dengan Postgres memungkinkan user untuk membuat sendiri *object file* atau *shared library* yang dapat diterapkan untuk mendefinisikan tipe data, fungsi, bahkan bahasa yang baru.



Dengan demikian Postgres memiliki dua kekuatan besar: source code dan arsitektur yang luwes, tentunya di samping fitur penting lainnya seperti dokumentasi yang lengkap. Disamping itu Postgres juga didukung oleh banyak antarmuka ke berbagai bahasa pemrograman seperti C++, Java, Perl, PHP, Python, dan Tcl. ODBC dan JDBC juga tersedia yang membuat Postgres lebih terbuka dan dapat diterapkan secara meluas.

#### **2.2.6. Perencanaan dan Penerapan *Institutional Repositories***

Setiap implementasi DSpace itu unik. Ketika teknologi ini cukup mudah untuk di-*install* dan dikonfigurasi, perancangan dan pembangunan IR dengan DSpace membutuhkan perencanaan terlebih dahulu, sebelum kita membangun *platform* teknologi dan memperkenalkan layanan kepada pengguna. Untuk membantu penerapan rancangan dan pembangunan DSpace, disarankan untuk merencanakan terlebih dahulu alat dan konten pada setiap tahap dari pembangunan DSpace, seperti:

- 1) Mendefinisikan Penawaran Layanan DSpace

DSpace adalah sebuah sistem repositori yang fleksibel dan kuat. Sebelum membangun infrastruktur teknis dari sistem, sangat penting untuk mendefinisikan bagaimana pastinya merencanakan untuk menggunakan sistem dan tipe layanan seperti apa yang akan ditawarkan.

- 2) Membuat Infrastruktur Pendukung Layanan

Sama seperti halnya seorang staf teknis menyusun infrastruktur teknis DSpace, diperlukan juga untuk menyusun infrastruktur layanan dari DSpace. Membangun sebuah layanan DSpace membutuhkan masukan dan perencanaan dari

berbagai sektor dari institusi *research* seperti staf dan *administrator* perpustakaan, fakultas, dan pemimpin institusi.

### 3) Model Objek DSpace (Membangun Komunitas dan Koleksi)

DSpace didesain untuk membuat suatu partisipasi oleh penyimpan menjadi mudah. Informasi dari sistem dibangun disekitar ide dari “*Communities*” organisasi, sebuah Sub-unit alami dari institusi yang mempunyai informasi yang dibutuhkan manajemen. Setiap komunitas dapat beradaptasi dengan sistem untuk memenuhi kebutuhan tertentu dan mengatur proses pengumpulannya sendiri.

Item-item disusun ke dalam sebuah hirarki dimana menyerupai item yang dikelompokkan dan dikumpulkan ke dalam koleksi dari konten yang serupa. Komunitas merupakan level tertinggi dari konten organisasi. Sama seperti suatu koleksi yang dapat masuk kedalam lebih dari satu komunitas. Setiap item yang tersimpan di dalam repositori DSpace terbuat dari sebuah ikatan dari *stream*, sehingga *file-file* dapat disimpan di dalam objek digital tunggal sebanyak yang dibutuhkan. *Bitstream-bitstream* mengikuti format *bitsream* yang sudah dikenali oleh sistem sebelumnya, dan DSpace mempunyai sifat yang bertolak belakang dengan tipe tipe objek yang berbeda, sebagai contoh gambar dapat ditampilkan *thumbnail*-nya ketika berada di sistem *browsing*.

### 4) *Metadata*

DSpace menggunakan standar metadata *Dublin Core* yang sudah berkualifikasi untuk mendeskripsikan item secara intelektual. Hanya 3 *field* yang dibutuhkan yaitu *title*, *language*, dan *submission date*, *field-field* yang lain hanyalah optional. Terdapat beberapa *field* untuk abstrak dokumen, *keyword*, dan metadata

teknis dan metadata hak, dan lain lain. Metadata ini ditampilkan di dalam catatan item di dalam DSpace, dan diindekskan untuk sistem *browsing* dan *searching* (di dalam *collection*, antar *collection*, atau antar *communities*). Untuk *Dissemination Information Package* (DIP) dari *framework* OAIS, sistem saat ini melakukan *export* metadata dan bahan digital di dalam sebuah skema *custom XML* sedangkan metadata bekerja dengan komunitas METS untuk mengembangkan skema-skema ekstensi yang diperlukan untuk metadata teknis dan hak akses tentang format digital yang sembarangan.

#### 5) *User Interface*

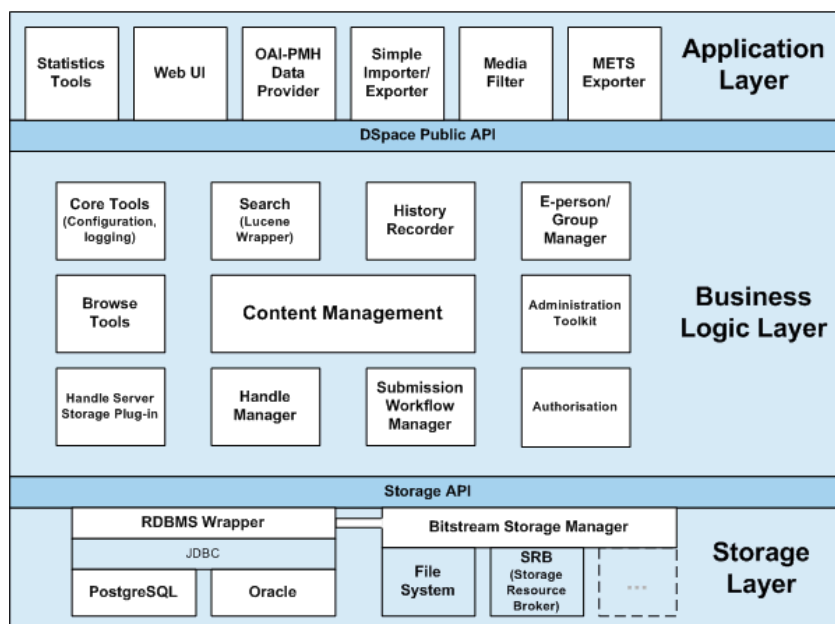
*User interface* yang dimiliki DSpace saat ini adalah *web-based*. Terdapat beberapa *interface*, satu untuk pengumpul dan yang lainnya melibatkan di dalam proses pengumpulan, satu bagi *end-user* yang ingin mencari informasi, dan satu untuk *administrator* sistem.

*Interface end-user* maupun *public* mendukung pencarian dan penerimaan dari item dengan cara *browsing* atau *searching* dari metadata (semua *field* sekarang, dan beberapa *field* tertentu di masa depan). Setelah item di tempatkan di dalam sistem, penerimaan dapat diselesaikan dengan melakukan *click* pada *link* yang dapat menyebabkan bahan yang tersimpan dapat diunduh ke *web browser* milik *user*. Format “*Web-native*” (yang akan menampilkan secara langsung pada *web browser* atau dengan *plug-in*) dapat dilihat secara langsung, format lain harus disimpan ke lokal komputer milik user dan dilihat dengan program yang terpisah dan dapat menginterpretasikan filenya (sebagai contoh *spreadsheet* Microsoft Excel, *dataset* SAS, atau *file* CAD/CAM).

## 6) Platform Teknologi

DSpace dikembangkan menjadi *open source*, dan dalam hal ini institusi dan organisasi dengan resource yang minim dapat menjalankannya. Sistem ini dirancang untuk berjalan pada *platform* UNIX, dan meliputi *middleware* dan *tool open source* lain, dan program yang ditulis oleh team DSpace. Semua kode asli ditulis dengan bahasa pemrograman Java. Teknologi lain yang dapat tersambung seperti *relational database management system* (PostgreSQL), *web server* dan Java *servlet engine* (Apache dan Tomcat, keduanya dari Apache Foundation), Jena (toolkit RDF dari lab HP), OAICat dari OCLC, dan beberapa *library* yang berguna lainnya. Semua komponen *leveraged* dan *library* juga merupakan *software open source*.

## 7) Arsitektur Sistem



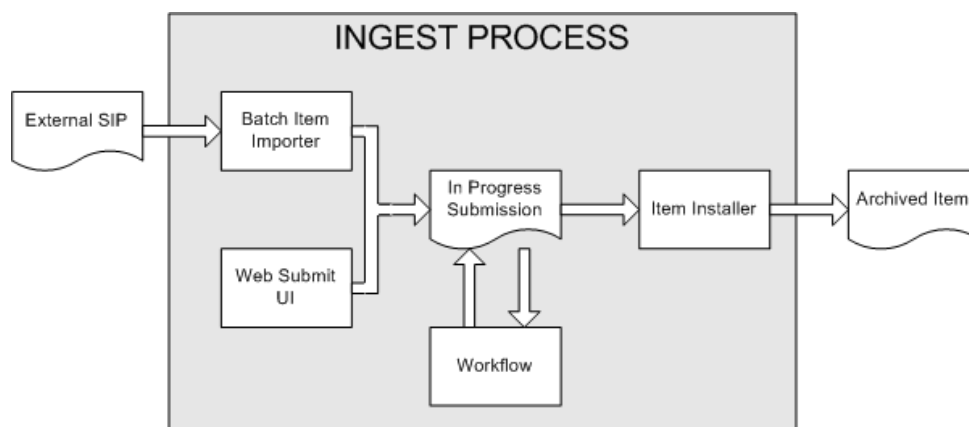
**Gambar 2.1** DSpace Technical Architecture

Seperti yang sudah dijelaskan pada Gambar 2.1, arsitektur DSpace merupakan arsitektur 3 lapis yang lurus, yang didalamnya terdapat *storage*, *business*, dan *application layer*. Masing-masing menggunakan dokumentasi API untuk mengizinkan customisasi dan penambahan di waktu yang akan datang. *Layer storage* diterapkan menggunakan sistem *file*, sebagaimana tabel-tabel database yang diatur oleh PostgreSQL. *Layer* bisnis adalah dimana fungsionalitas spesifik DSpace berada, termasuk *workflow*, manajemen konten, administrasi, serta pencarian modul. Setiap modul mempunyai API yang mengizinkan pengadopsi DSpace untuk menggantikan dan menambah fungsi tersebut sesuai keinginan. Selanjutnya, *application layer* meng-over *interface* dari sistem: UI *web* dan *batch loader*, di bagian tertentu, namun juga pada OAI *support* dan *handle server* untuk mengatasi *persistent identifier* pada item-item DSpace. Layer ini akan mendapatkan banyak perhatian terkait *release* di masa depan, seperti penambahan web service untuk fitur-fitur baru (contoh: untuk mendukung interoperasi dengan sistem lain) dan mendefinisikan layanan-layanan pemerintah lintas jangkauan institusi dengan mengadopsi DSpace.

#### 8) Ingesti DSpace

Di dalam sistem yang memiliki tujuan, bisa jadi aspek paling kritical dari sistem adalah bagaimana data masuk ke dalam sistem. Hal ini muncul di dalam DSpace dalam 2 cara. *Web base UI* untuk *software* mengizinkan pengguna untuk mengumpulkan *item* ke *collection* selama mereka masuk sebagai pengguna terdaftar. Ketika pengguna *log in*, mereka akan melewati *workflow* yang dapat

dikonfigurasi dimana mereka dapat mengunggah dan mendeskripsikan *item* yang mereka kumpulkan seperti pada Gambar 2.2.



**Gambar 2.2** *DSpace Ingest Process*

Alternatifnya, Administrator DSpace yang memiliki konten yang akan diimpor batch berskala besar dapat mengambil keuntungan dari fungsionalitas impor/ekspor dari sistem. Item pengimpor berupa command line tool yang datang satu paket dengan sistem dan memungkinkan pengguna untuk mengimpor collection dari konten ke dalam sistem yang dapat dilihat pada Gambar 2.3.

```

archive_directory/
  item_000/
    dublin_core.xml — qualified Dublin Core metadata
    contents — text file containing one line per filename
    file_1.doc — files to be added as bitstreams to the item
    file_2.pdf
  item_001/
    dublin_core.xml
    contents
    file_1.prg
    .....
  
```

**Gambar 2.3** *DSpace's Simple Archive Format for Importing and Exporting*

Pengimpor *item* menggunakan format arsip DSpace, yang mana merupakan direktori sederhana yang menyimpan item-item yang akan diimpor ke dalam sistem (Contoh arsip sederhana ditunjukkan seperti gambar 4 diatas). Sebuah direktori arsip *top-level* mengandung nama direktori yang unik, masing-masing mengandung semua yang diperlukan untuk mengimpor sebuah item tunggal. Setiap sub-folder dibutuhkan untuk menyimpan 2 *file*, sebagai tambahan untuk konten nyata yang akan diimpor. *File* yang dibutuhkan “*dublin\_core.xml*” mengandung representasi XML dari nama-nama elemen Dublin Core yang berkualifikasi dan konten tekstual yang mengandung catatan metadata, termasuk *author*, *title*, dan sebagainya. Sebuah teks datar *file* “*content*” mempunyai satu baris yang mengandung nama *file* dari setiap *file* yang akan diikut sertakan di dalam objek digital. Ketika struktur ini diletakkan pada tempatnya, *command impor* dapat langsung dijalankan dan semua konten akan diimpor menuju repositori. Tool menyediakan “*map file*” setelah dijalankan, yang mana menjelaskan semua item yang diimpor dan lokasi barunya di dalam sistem tersebut, *file* ini dapat berguna di masa yang akan datang untuk mengekspor atau menghilangkan konten yang telah diimpor.

#### 9) Alur Kerja DSpace

Alur kerja sistem pengumpulan DSpace adalah bagian kritical dari arsitektur DSpace yang mengijinkan pengumpulan, pemrosesan, dan tambahan akhir dari konten ke repository yang ada. Model yang dimiliki DSpace termasuk *E-People*, adalah pengguna yang terdaftar dengan sistem dan mempunyai *authorisasi*, *role*, hak, dan *privilege* tertentu yang menerjemahkan kemampuan untuk menyelesaikan tugas tertentu dalam sistem DSpace. Pengumpulan biasanya dimulai

dengan sistem yang menanyakan beberapa pertanyaan kepada pengguna tentang dokumen digital yang akan ditambahkan ke dalam repositori dan beberapa *file* yang terkait di dalam pengumpulan. Lalu sistem mengarahkan pengguna melalui beberapa langkah, yang dapat dilihat dalam pada Tabel 2.1.

**Tabel 2.1** Alur Kerja DSpace

Langkah Alur Kerja	Deskripsi
1. <i>Describe</i>	Pengguna memasukkan metadata tentang dokumen yang mereka kumpulkan, termasuk tentang pengarang, judul, kata kunci, dan deskripsi
2. <i>Upload</i>	Pengguna memilih dan mengupload <i>file</i> yang ada di dalam mesin lokal yang akan mereka upload sebagai bagian dari pengumpulan. Setiap tipe <i>file</i> diidentifikasi oleh sistem dan pengguna memverifikasinya.
3. <i>Verify</i>	Disini sebuah gambaran semua rincian dari pengumpulan diberikan, termasuk kesimpulan dari metadata yang telah dimasukkan dan <i>file-file</i> yang terkait pada pengumpulan.
4. <i>License</i>	Pengguna diperlihatkan dan harus setuju dengan lisensi dari administrator sistem yang telah ditetapkan untuk mengumpulkan konten untuk koleksi ini.
5. <i>Complete</i>	Aksi pengguna di dalam proses pengumpulan telah selesai. Berdasarkan langkah-langkah alur kerja yang ditetapkan untuk pengkoleksian, item boleh ditambahkan segera kepada koleksi atau harus direview oleh administrator sistem terlebih dahulu sebelum penambahan ke koleksi.

#### 10) Penyebaran

Item-item yang telah dikumpulkan dan diarsipkan menjadi DSpace repositori *digital library* dapat disebar dan diakses oleh pengguna melalui internet dan *browser*. DSpace menyediakan penggunaanya kemampuan untuk



mencari item DSpace dengan sederhana, mudah, dan canggih. Dari halaman *home* DSpace, pengguna dapat melihat semua item yang ada di dalam DSpace dengan kategori pengarang, judul, atau tanggal terbit.

#### 11) Observasi

DSpace menyediakan cara untuk mengatur material penelitian dan publikasi dalam repositori yang tertata secara profesional untuk memberikan *visibility* dan *accessibility* yang besar dari waktu ke waktu. Hal ini dapat membantu untuk:

- Mendapatkan hasil penelitian secara cepat, kepada audiensi dalam *range* dunia
- Meraih audiensi dalam skala dunia melalui keterbukaannya dengan *course management systems*
- Mengarsipkan dan mendistribusikan material yang dapat diletakkan di *website personal*
- Menyimpan contoh dari proyek-proyek pelajar/mahasiswa (dengan persetujuan)
- Menampilkan skripsi mahasiswa (dengan persetujuan)
- Menyimpan *track* dari publikasi/bibliografi pribadi
- Memiliki *network identifier* yang kuat untuk bekerja, yang tidak akan pernah berubah atau rusak

#### **2.2.7. Black Box Testing**

Pengujian perangkat lunak ada dua yaitu *white box testing* dan *black box testing*. Pengujian pada penelitian ini menggunakan *black box testing*. Perangkat lunak memerlukan seperangkat tes untuk pencarian kesalahan fungsi-fungsi dalam

aplikasi sehingga dalam hal ini *black box testing* lebih sesuai. Pengujian ini digunakan untuk mengetahui apakah fungsi-fungsi dalam perangkat lunak sudah sesuai dengan yang diharapkan.

Menurut Roger S. Pressman (2010), *black box testing* berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan *engineer* untuk memperoleh *input* yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program. *Black box testing* berusaha untuk menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau fungsi yang hilang.
2. Kesalahan antarmuka.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja.
5. Kesalahan inisialisasi dan pemutusan kesalahan.