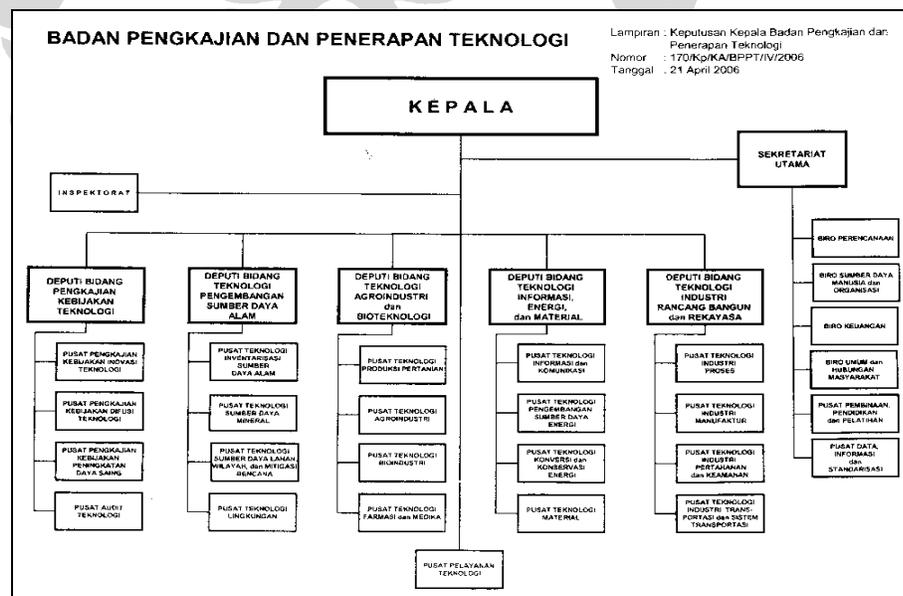


4. ANALISIS DAN DISAIN

Bab ini menjelaskan *project flow* di BPPT, metodologi pengembangan, disain ontologi, serta *use cases* dengan *use cases specification* sebagai landasan untuk pengembangan prototipe sistem.

4.1 Project Flow BPPT

Proyek di lembaga riset BPPT diklasifikasikan menjadi tiga jenis, yaitu *BPPT Project*, *Cooperation Project*, dan *Participation Project*. Proyek yang dikembangkan oleh peneliti BPPT tanpa ada campur tangan pihak luar termasuk ke dalam *BPPT Project*. Sedangkan pada *Cooperation Project*, proyek yang dikembangkan kerjasama yang setara antara BPPT dan pihak luar dimana kedua pihak sama-sama saling menentukan pengembangan proyek. *Participation Project* adalah proyek yang dikelola penuh pihak luar dimana pihak BPPT hanya sebagai narasumber ahli. Pihak BPPT yang terlibat dalam proyek dapat berasal dari satu atau lebih satuan kerja organisasi. Sebagian struktur organisasi BPPT dapat dilihat pada bagan di bawah ini.



Gambar 4.1 Struktur Organisasi BPPT [18]

Seluruh proyek yang dihasilkan oleh lembaga ini akan dilaporkan ke dalam dokumen-dokumen dikelompokkan dalam Sistem Pelaporan Rekayasa. Dokumen-dokumen ini merupakan dokumen internal, yaitu *Decision Sheet*, *Instruction Sheet*, *Program Manual*, *Technical Report*, *Preliminary Report*, *Middle Term Report*, serta *Final Report*. Sedangkan ada pula dokumen yang ditulis untuk dipublikasikan kepada masyarakat, diantaranya karya ilmiah dan buku.

Alur pengembangan *BPPT Project* diawali dengan rencana strategis BPPT mengacu kepada suatu unit kerja (satuan kerja). Kemudian unit kerja ini mengusulkan proyek dan dievaluasi oleh *CEO (Chief Executive Officer)* BPPT. CEO memberikan *feedback* terhadap usulan proyek tersebut. *BPPT Project* akan berjalan ketika CEO sudah menyetujui usulan proyek yang dibuat oleh unit kerja.

Pada *Cooperation Project*, pengembangan dimulai dengan adanya campur tangan pihak luar yang berperan setara dengan pihak BPPT. Keduanya melakukan perjanjian kerjasama dengan membentuk tim kerja. Tim ini terdiri dari tim pihak luar dan tim pihak BPPT. Kemudian kedua tim tersebut bersama-sama dalam mengembangkan proyek.

Sedangkan *Participation Project*, pihak luar memanggil narasumber ahli dari BPPT. Akan tetapi BPPT tidak ikut dalam pengembangan proyek. Peran narasumber tersebut membantu pihak luar dalam mengembangkan proyek. Pihak luar inilah yang secara penuh memiliki, mengelola, dan mengembangkan proyek. Sehingga dapat dikatakan bahwa BPPT hanya mengikuti suatu *project event* dan tidak ikut dalam mengembangkannya.

4.2 Analisis Contingency Factor

Sebelum membahas lebih mendetail mengenai *contingency factor*, terlebih dahulu kita mengetahui definisi *contingency view* dan perbedaannya dengan *universalistic view* suatu *knowledge management*. *Universalistic view* fokus kepada mengidentifikasi hanya dengan satu cara untuk mendapatkan hasil yang baik, sedangkan *contingency view* fokus kepada indentifikasi melalui berbagai alternatif cara dan kesuksesan sesuatu diperoleh hanya dari satu cara yang sesuai. [6]

Berdasarkan pengamatan di BPPT dan pengetahuan mengenai *project flow* milik lembaga tersebut, penulis juga dapat menilai bahwa terdapat beberapa faktor yang mempengaruhi jenis *knowledge management system* yang akan dikembangkan pada lembaga ini. Penilaian dengan *contingency perspective* yang merupakan pendekatan berdasarkan kultur organisasi yang ada di suatu lembaga.

Adapun rekomendasi proses *knowledge management* terhadap efek karakteristik organisasi, lingkungan, dan strategi bisnis. Organisasi dilihat dari ukurannya, yaitu *large* atau *small*. Biasanya organisasi besar (*large*) memiliki birokrasi yang lebih besar pula dan terdapat suatu rutinitas (*routine*) karena organisasi tersebut menggunakan suatu standardisasi, sedangkan organisasi kecil (*small*) tidak terlalu birokratis sehingga dalam pelaksanaannya tidak bergantung kepada standardisasi dan aturan serta cenderung memerlukan mekanisme arahan (*direction*). Jika dilihat dari lingkungan tak tertentu (*environment uncertainty*) dibedakan menjadi tinggi (*high*) dan rendah (*low*).

Low environment uncertainty mengimplikasikan proses yang paling baik adalah *knowledge sharing* dan *knowledge capture*, sedangkan *high environment uncertainty* mengimplikasikan proses yang baik adalah *knowledge application* dan *discovery*. Terakhir adalah bisnis strategi dibedakan menjadi *low cost* dan *differentiation*. *Low cost strategy* fokus kepada menerapkan *knowledge* yang sudah ada daripada membuat *knowledge* baru. Lain halnya dengan *differentiation strategy* mengacu kepada inovasi dan mengembangkan *knowledge* baru. [6]. Pada tabel berikut merupakan ringkasan dari proses *knowledge management* yang direkomendasikan sesuatu dari karakteristik yang telah dibahas.

Tabel 4.1 Efek Karakteristik Lingkungan dan Organisasi Terhadap Proses KM [6]

Characteristic	Level/Type	Recommended KM Process
Organization size	Small	Knowledge sharing (socialization) Knowledge application (direction) Knowledge discovery (combination, socialization) Knowledge capture (externalization, internalization)
	Large	Knowledge sharing (exchange) Knowledge application (routines) Knowledge discovery (combination) Knowledge capture (externalization, internalization)
Business strategy	Low cost	Knowledge application (direction, routines) Knowledge capture (externalization, internalization) Knowledge sharing (socialization, exchange)
	Differentiation	Knowledge discovery (combination, socialization) Knowledge capture (externalization, internalization) Knowledge sharing (socialization, exchange)
Environmental uncertainty	Low	Knowledge sharing (socialization, exchange) Knowledge capture (externalization, internalization)
	High	Knowledge discovery (combination, socialization) Knowledge application (direction, routines)

Lalu dari proses *knowledge management* diatas, kita dapat melakukan penilaian berdasar *contingency factor*. Pengidentifikasiian proses tersebut dengan tiap-tiap *contingency factor* menggunakan suatu tabel penilaian sebagai berikut.

Tabel 4.2 Penilaian terhadap Keadaan yang Sesuai dengan Beberapa Proses KM [6]

KM Processes	Task Uncertainty	Task Interdependence	Explicit Or Tacit Know.	Procedural or Declarative Know.	Org. Size	Business Strategy	Environmental Uncertainty
Combinaton	Low	High	E	P/D	S/L	D	High
Socialization for knowledge discovery	High	High	T	P/D	S	D	High
Socialization for knowledge sharing	High	High	T	P/D	S	LC/D	Low
Exchange	Low	High	E	P/D	L	LC/D	Low
Externalization	Low	Low	T	P/D	S/L	LC/D	Low
Internalization	Low	Low	E	P/D	S/L	LC/D	Low
Direction	High	High/Low	T/E	P	S	LC	High
Routines	Low	High/Low	T/E	P	L	LC	High

Berdasarkan pengamatan, BPPT merupakan organisasi besar dengan strategi bisnis bertipe *differentiation* dan *environmental uncertainty* rendah sehingga dengan menggunakan tabel-tabel di atas sebagai alat untuk menilai, penulis dapat memprioritaskan proses *knowledge management* atau dengan kata lain dapat mengetahui model *knowledge management system* yang tepat untuk BPPT. Hasil penilaian antara proses *knowledge management* dengan *contingency factor* dapat dilihat dari tabel 4.3 berikut ini.

Tabel 4.3 Contingency Perspective pada BPPT

Contingency Factors KM Processes	Task Uncertainty=Low	Task Inter-dependence =Low	Explicit Know.	Procedural Know.	Org. Size =Large	Business Strategy= Differentiation	Environmental Uncertainty= Low
Combinaton	Yes	No	Yes	Ok	Ok	Yes	No
Socialization for knowledge discovery	No	No	No	Ok	No	Yes	No
Socialization for knowledge sharing	No	No	No	Ok	No	Yes	Yes
Exchange	Yes	No	Yes	Ok	Yes	Ok	Yes
Externalization	Yes	Yes	No	Ok	Ok	Ok	Yes
Internalization	Yes	Yes	Yes	Ok	Ok	Ok	Yes
Direction	No	Ok	Ok	Yes	No	No	No
Routines	Yes	Ok	Ok	Yes	Yes	No	No
Number of "yes"	Number of "Ok"	Number of "No"	Cummulative *				
3	2	2	4				
1	1	5	1.5				
2	1	4	2.5				
4	2	1	5				
3	3	1	4.5				
4	3	-	5.5				
1	2	4	2				
3	2	2	4				

* Yes = 1, Ok = 0.5, No = 0

. Suatu organisasi yang memiliki tingkat *uncertainty* yang rendah berarti *knowledge management* yang dibutuhkan untuk *task* tersebut adalah *low task interdependence* dan *low task uncertainty*. Berdasarkan tabel hasil di atas, BPPT

fokus kepada *internalization* dan *exchange* (pada keduanya ditekankan kepada *knowledge capture* dan *knowledge sharing*). Sehingga model *knowledge management system* yang sesuai adalah *knowledge capture* atau *knowledge sharing*. Akan tetapi, jika dilihat kembali kepada tujuan dari pengembangan *knowledge management system* di lembaga ini dan hasil pengamatan *project flow* serta penilaian faktor *contingency*, model sistem yang dikembangkan adalah *knowledge sharing system*.

4.3 Disain Ontologi

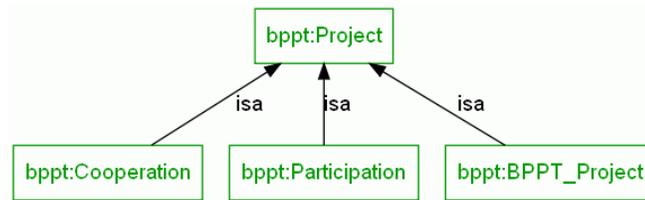
Disain ontologi pada BPPT Intranet tidak menggunakan *ontology reuse*, tetapi membuat sendiri dengan memperhatikan *project flow* yang terdapat di lembaga riset BPPT. Disain dibuat dengan menggunakan tool Protégé 3.4, terdiri dari lima *classes* utama ,yaitu *Project*, *Product*, *Competency*, *Person*, dan *Organization* dengan menggunakan bantuan OntoVizTab sebagai visualisasi grafik berikut ini.



Gambar 4.2 Classes Ontologi BPPT

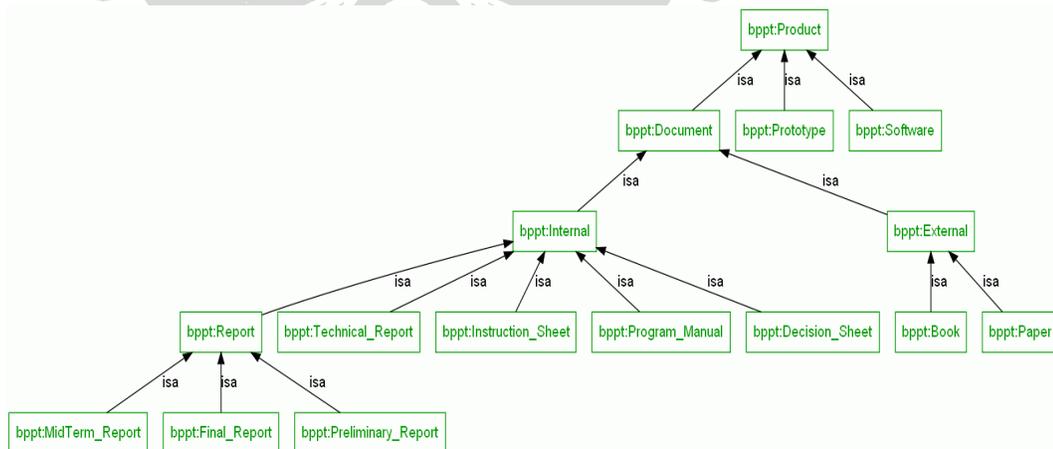
Seluruh *classes* memiliki *subclass*, kecuali *class Person*. Berikut merupakan penjelasan *subclasses* yang dimiliki oleh *class Project*, *Product*, *Competency*, dan *Organization*, yaitu:

1. *Class Project* merupakan *class* yang mengandung jenis-jenis proyek yang dimiliki oleh lembaga riset BPPT. Berdasarkan *project flow* yang telah dirinci pada subbab 3.1, *class Project* memiliki tiga buah *subclasses* yaitu, *BPPT_Project*, *Cooperation*, dan *Participation*.



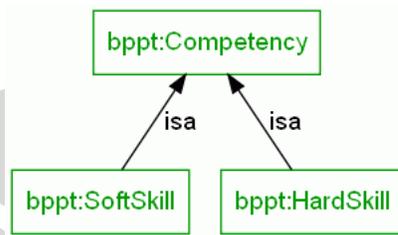
Gambar 4.3 Class Project

2. *Class Product* merupakan *class* yang berisikan semua jenis produk yang dihasilkan oleh suatu proyek dan ditulis atau dikembangkan oleh *knowledge worker* yang bekerja pada suatu unit/satuan kerja tertentu. Terdiri dari tiga *subclasses*, yaitu *Document*, *Prototype*, dan *Software*. Khusus *subclass Document* dibagi lagi menjadi dua *subclasses*, yaitu *Internal* dan *External*. *Subclass Internal* terdiri dari dokumen-dokumen yang dipublikasikan pada internal BPPT, sedangkan *subclass External* terdiri dari dokumen-dokumen yang dipublikasikan untuk pihak luar. *Subclass Internal* pun memiliki lima *subclasses* *Decision_Sheet*, *Instruction_Sheet*, *Program_Manual*, *Technical_Report*, dan *Report*. *Subclass Report* dibagi menjadi tiga *subclasses* lagi, yaitu *Preliminary_Report*, *MidTerm_Report*, dan *Final_Report*. Pada *subclasses External* dibagi menjadi dua, yaitu *Book* dan *Paper*. Penjelasan visualisasi dengan grafik dapat dilihat pada gambar berikut ini.



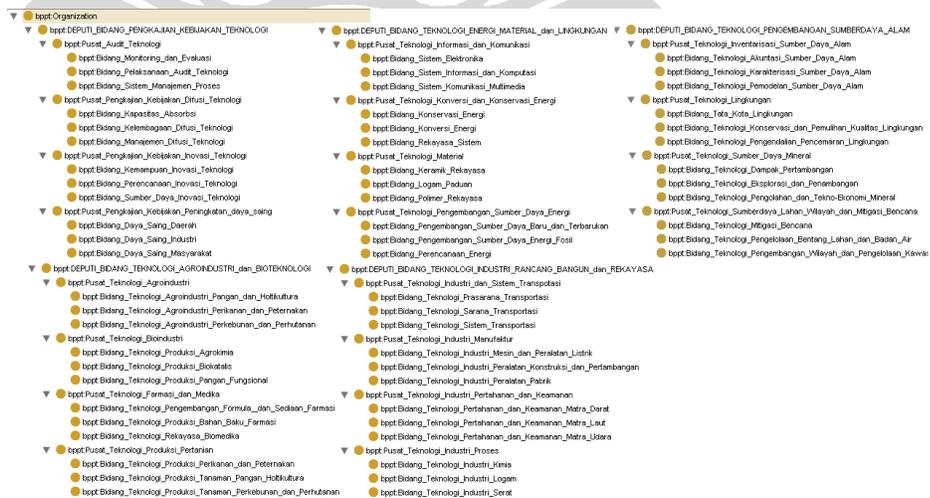
Gambar 4.4 Class Product

3. *Class Competency* merupakan sebuah *class* yang berisikan keahlian yang dimiliki oleh *knowledge worker* BPPT. *Class* ini memiliki dua buah *subclasses*, yaitu *HardSkill* dan *SoftSkill*. *HardSkill* mengarah kepada keahlian teknik perangkat keras sedangkan *SoftSkill* merupakan keahlian perangkat lunak. Berikut ini grafik yang menggambarkan hubungan *class* dan *subclass Competency*.



Gambar 4.5 Class Competency

4. *Class Organization*, merupakan struktur organisasi yang dimiliki BPPT. Struktur organisasi terdiri dari deputy-deputy yang membawahi beberapa satuan kerja. Satuan-satuan kerja tersebut membawahi beberapa bidang pula. Bidang inilah tempat dimana *knowledge worker* bekerja. Berikut merupakan penjelasan lebih detail tanpa visualisasi grafik. Hal ini dikarenakan *subclasses* struktur organisasi sangatlah banyak.



Gambar 4.6 Class Organization

Sehingga jika ditotal seluruhnya ontologi pada BPPT Intranet ini terdiri dari 110 *classes* (termasuk *classes* dan *subclasses*), 16 *object properties*, dan 17 *datatype properties*. Tabel berikut menjelaskan *object properties* dan *datatype properties*. Pada tabel 4.4 ini berisikan 16 *object properties* pada ontologi BPPT.

Tabel 4.4 *Object Properties* Ontologi BPPT

No.	<i>Object Properties</i>	Keterangan
1.	bppt:AuthorOf	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Person</i> dengan <i>class Product</i> . Hal ini menjelaskan satu atau beberapa orang menulis atau mengembangkan suatu produk (baik dokumen, perangkat lunak, atau prototipe).
2.	bppt:DevelopedBy	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:Developes, digunakan untuk membuat relasi antara <i>class Project</i> dengan <i>class Organization</i> . Hal ini menjelaskan suatu proyek dikembangkan oleh satu atau beberapa unit/satuan kerja.
3.	bppt:Developes	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Organization</i> dengan <i>class Project</i> , menjelaskan bahwa satu/beberapa unit kerja dalam struktur organisasi mengembangkan suatu proyek.
4.	bppt:DoneBy	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:IsDoing, digunakan untuk membuat relasi antara <i>class Project</i> dengan <i>class Person</i> . Hal ini menjelaskan suatu proyek dikerjakan oleh beberapa orang.
5.	bppt:Has	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Person</i> dengan <i>class Competency</i> . Hal ini menjelaskan bahwa seseorang memiliki keahlian atau <i>skill</i> .

Tabel 4.4 Object Properties Ontologi BPPT (lanjutan)

6.	bppt:HeadBy	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:HeadOf, digunakan untuk membuat relasi antara <i>class Project</i> dengan <i>class Person</i> . Hal ini menjelaskan suatu proyek dikepalai atau dipimpin oleh seseorang.
7.	bppt:HeadOf	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Person</i> dengan <i>class Project</i> . Hal ini menjelaskan bahwa seseorang memimpin suatu proyek.
8	bppt:IsAuthorBy	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:AuthorOf, digunakan untuk membuat relasi antara <i>class Product</i> dengan <i>class Person</i> . Hal ini menjelaskan suatu produk ditulis atau dikembangkan oleh satu atau beberapa orang.
9	bppt:IsDoing	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Person</i> dengan <i>class Project</i> . Hal ini menjelaskan bahwa beberapa orang mengerjakan suatu proyek.
10.	bppt:IsHad	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:Has, digunakan untuk membuat relasi antara <i>class Competency</i> dengan <i>class Person</i> . Hal ini menjelaskan bahwa keahlian atau <i>skill</i> dimiliki oleh seseorang.
11.	bppt:IsProduceFrom	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:Produce, digunakan untuk membuat relasi antara <i>class Product</i> dengan <i>class Project</i> . Hal ini menjelaskan suatu produk dihasilkan dari pengembangan suatu proyek.

Tabel 4.4 Object Properties Ontologi BPPT (lanjutan)

12.	bppt:IsWorkOnBy	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:WorksOn, digunakan untuk membuat relasi antara <i>class Organization</i> dengan <i>class Person</i> . Hal ini menjelaskan suatu organisasi dijalankan oleh beberapa orang.
13.	bppt:Participate	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Person</i> dengan <i>subclass Participation</i> dari <i>class Project</i> . Hal ini menjelaskan bahwa seseorang berpartisipasi sebagai narasumber ahli dari suatu proyek yang berjenis <i>Participation Project</i> .
14.	bppt:ParticipatedBy	<i>Property</i> ini merupakan <i>inverse property</i> dari bppt:Participate, digunakan untuk membuat relasi antara <i>class Project</i> dengan <i>class Person</i> . Hal ini menjelaskan suatu organisasi dijalankan oleh beberapa orang.
15.	bppt:Produce	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Project</i> dengan <i>class Product</i> . Hal ini menjelaskan suatu proyek yang dikembangkan akan menghasilkan suatu produk (baik berupa dokumen, perangkat lunak, atau prototipe).
16.	bppt:WorksOn	<i>Property</i> ini digunakan untuk membuat relasi antara <i>class Person</i> dengan <i>class Organization</i> . Hal ini menjelaskan bahwa seseorang bekerja pada satu bidang dalam suatu struktur organisasi.

Tabel 4.5 merupakan *datatype properties* yang dimiliki oleh ontologi BPPT. Terdiri dari 17 *datatype properties* yang masing-masing dijelaskan lebih rinci pada keterangan tabel berikut.

Tabel 4.5 Datatype Properties Ontologi BPPT

No.	Datatype Properties	Keterangan
1.	bppt:alamat	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Person</i> , atribut ini diisi dengan <i>instance</i> alamat seseorang.
2.	bppt:email	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Person</i> , atribut ini diisi dengan <i>instance e-mail</i> seseorang.
3.	bppt:deskripsi	Merupakan <i>property</i> dengan tipe data <i>String</i> , atribut ini diisi dengan deskripsi atau penjelasan dari nama suatu produk dan proyek.
4.	bppt:foto	Merupakan <i>property</i> yang menjadi salah satu atribut pada <i>class Person</i> , atribut ini diisi dengan gambar foto <i>knowledge worker</i> .
5.	bppt:homepage	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Person</i> , atribut ini diisi dengan <i>instance</i> alamat web atau <i>blog</i> atau <i>homepage</i> seseorang.
6.	bppt:link	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Document</i> , atribut ini diisi dengan <i>instance</i> nama <i>file</i> dokumen.

Tabel 4.5 Datatype Properties Ontologi BPPT (lanjutan)

7.	bppt:nama	Merupakan <i>property</i> dengan tipe data <i>String</i> yang berfungsi sebagai label.
8	bppt:nip	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Person</i> , atribut ini diisi dengan <i>instance</i> NIP seseorang.
9	bppt:pendidikan	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Person</i> , atribut ini diisi dengan <i>instance</i> riwayat pendidikan seseorang.
10.	bppt:penerbit	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>subclass Book</i> dan <i>Paper</i> dari <i>class Product</i> , atribut ini diisi dengan <i>instance</i> penerbit.
11.	bppt:pemilikproyek	Merupakan <i>property</i> diisi dengan <i>instance</i> nama pemilik proyek.
12.	bppt:statusproyek	Merupakan <i>property</i> diisi dengan <i>instance</i> status proyek apakah telah selesai atau masih aktif berjalan.
13.	bppt:tahun	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Project</i> diisi dengan <i>instance</i> tahun dikembangkan atau diselesaikannya suatu proyek.
14.	bppt:tanggal	Merupakan <i>property</i> dengan tipe data <i>Date</i> yang menjadi salah satu atribut pada <i>class Product</i> diisi dengan <i>instance</i> tanggal rilis produk.

Tabel 4.5 Datatype Properties Ontologi BPPT (lanjutan)

15.	bppt:telepon	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Person</i> diisi dengan <i>instance</i> nomor telepon seseorang.
16.	bppt:versi	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>class Product</i> diisi dengan <i>instance</i> versi produk.
17.	bppt:volume	Merupakan <i>property</i> dengan tipe data <i>String</i> yang menjadi salah satu atribut pada <i>subclass Book</i> dan <i>Paper</i> dari <i>class Product</i> , atribut ini diisi dengan <i>instance</i> volume keberapakah produk itu diterbitkan.

Berikut ini merupakan penjelasan *quantifier restriction* yang diberikan pada relasi antara *classes* dengan menggunakan *object properties* yang telah dijelaskan di tabel sebelumnya. Terdapat dua jenis *quantifier restriction* dalam Protégé 3.4, yaitu **only** dan **some**. Pada *quantifier only* merupakan *quantifier universal* $\forall(\text{allValuesFrom})$, sedangkan *quantifier some* merupakan *quantifier existential* $\exists(\text{someValuesFrom})$. Ontologi yang dimiliki BPPT terdiri atas *quantifier restriction* yang dijelaskan pada tabel 4.6 berikut. Pada tabel ini terdapat 16 *quantifier restriction* untuk ontologi BPPT.

Tabel 4.6 Quantifier Restriction Ontologi BPPT

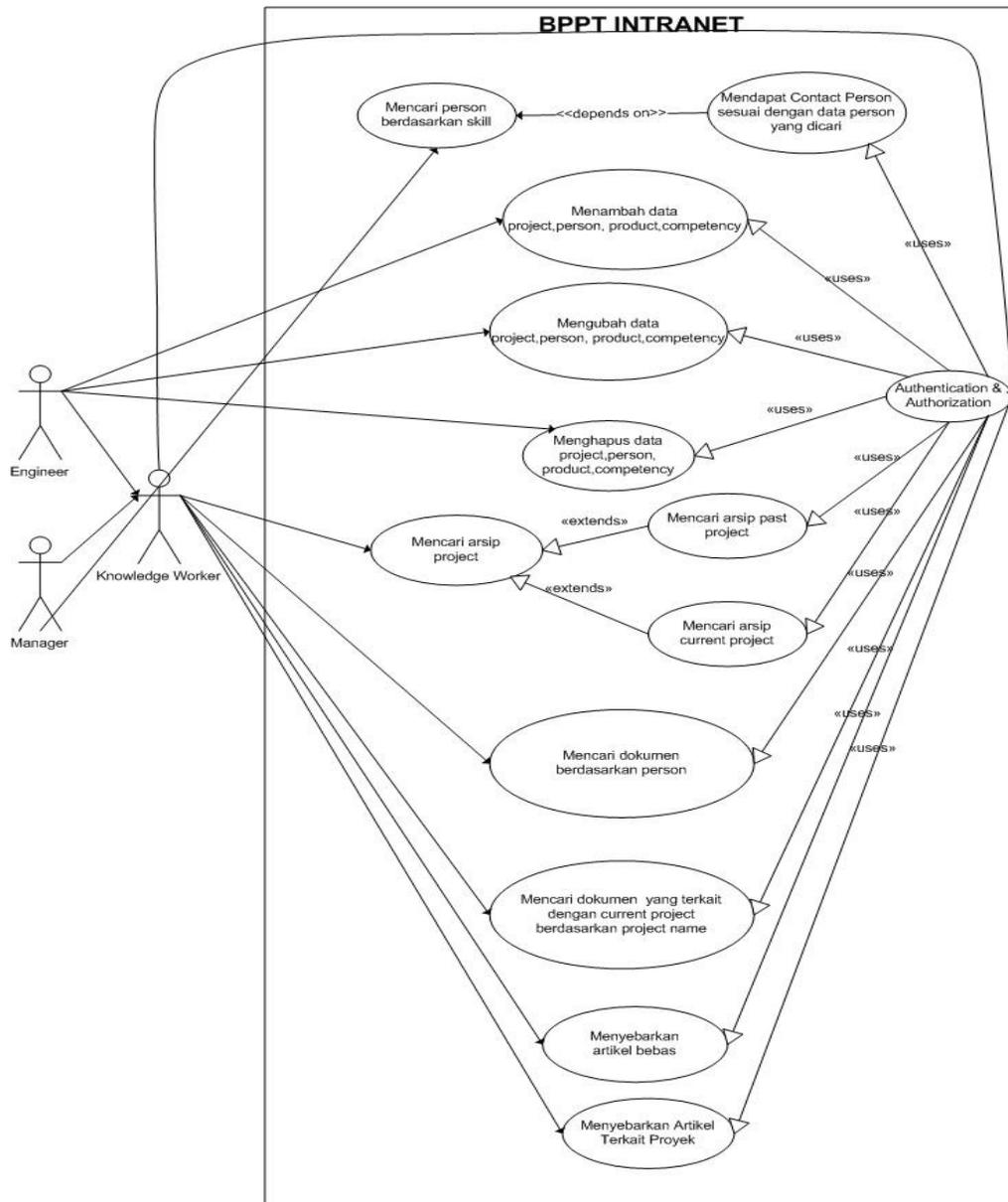
No.	Quantifier Restriction	Keterangan
1.	bppt:Competency bppt:IsHad only bppt:Person	Kompetensi hanya dimiliki oleh orang.
2.	bppt:Organization bppt:Developes only bppt:Project	Organisasi hanya mengembangkan proyek.
3.	bppt:Organization bppt:IsWorkOnBy some bppt:Person	Organisasi dijalankan oleh beberapa orang.
4.	bppt:Person bppt:AuthorOf some bppt:Product	Orang dapat menulis atau mengembangkan beberapa produk.
5.	bppt:Person bppt:Has some bppt:Competency	Orang dapat memiliki beberapa keahlian.
6.	bppt:Person bppt:HeadOf some bppt:Project	Orang dapat memimpin beberapa proyek.
7.	bppt:Person bppt:IsDoing some bppt:Project	Orang dapat menjalankan beberapa proyek.
8.	bppt:Person bppt:Participate some bppt:Participation	Orang dapat berpartisipasi dalam beberapa Participation Project.
9.	bppt:Person bppt:WorksOn only bppt:Organization	Orang hanya bekerja di suatu organisasi.
10.	bppt:Product bppt:IsAuthorBy only bppt:Person	Produk hanya dikembangkan atau ditulis oleh orang.

Tabel 4.6 Quantifier Restriction Ontologi BPPT (lanjutan)

11.	bppt:Product bppt:IsProduceFrom only bppt:Project	Produk hanya dihasilkan dari proyek.
12.	bppt:Project bppt:DevelopedBy only bppt:Organization	Proyek hanya dikembangkan oleh organisasi.
13.	bppt:Project bppt:DoneBy only bppt:Person	Proyek hanya dikerjakan oleh orang.
14.	bppt:Project bppt:HeadBy only bppt:Person	Proyek hanya dipimpin oleh orang.
15.	bppt:Project bppt:Produce some bppt:Product	Proyek dapat menghasilkan beberapa produk.
16.	bppt:Participation bppt:ParticipatedBy only bppt:Person	Participation Project hanya diikuti oleh orang.

4.4 Use Case

Berikut merupakan *use case* yang dihasilkan dari *requirement* yang telah dijelaskan pada tujuan dari pengembangan *knowledge management system* di lembaga riset BPPT.



Gambar 4.7 Use Case BPPT Intranet

4.5 Use Case Specification

Berikut di bawah ini merupakan penjelasan dari *use case* yang telah ditampilkan pada subbab sebelumnya.

4.5.1 Use Case Mencari Person Berdasarkan Skill

Use case ini digunakan untuk mencari data *person* atau orang-orang yang bekerja di lembaga riset BPPT. Pencarian *person* ini dilakukan oleh aktor *manager*. Terdapat beberapa *facet* yang dimiliki oleh *class person*, yaitu *facet name*, *product*, *field area*, dan *competency*. Seluruh *facet* inilah yang dapat digunakan aktor dalam mencari *person*. Jika pencarian *person* berdasarkan *skill* ingin dilakukan, maka pemilihan *browse* pada *facet competency*. Berikut tabel penjelasan *use case* Mencari Person Berdasarkan Skill.

Tabel 4.7 Use Case Mencari Person Berdasarkan Skill

Nama Use Case	Use Case Mencari Person Berdasarkan Skill.
Aktor utama	<i>Manager</i>
Kondisi awal	Aktor utama memilih <i>class Person</i>
Kondisi akhir	Aktor utama telah mendapatkan data <i>person</i> , misalnya NIP, <i>contact person</i> , riwayat pendidikan, bekerja pada bidang mana, proyek apa saja yang dikerjakan dan produk apa yang telah dihasilkan.
Skenario sukses utama	<ol style="list-style-type: none">1. Aktor berada pada halaman utama sistem.2. Aktor memilih <i>class Person</i>.3. Untuk mencari berdasarkan <i>skill</i>, aktor memilih <i>facet Competency</i> dengan cara memilih salah satu <i>instance</i> dari <i>competency</i> yang muncul.4. Aktor memperoleh data <i>person</i> yang dicarinya.
Skenario alternatif	<ol style="list-style-type: none">1. Aktor berada pada halaman utama sistem.2. Aktor memilih pencarian "All".3. Aktor memilih <i>class Competency</i>.4. Aktor dapat memilih langsung <i>instance</i> dari <i>Competency</i> atau dapat pula memilih pencarian <i>filter by subclass</i> dari <i>Competency</i> berupa <i>Hard Skill</i> dan <i>Soft Skill</i> yang berada pada <i>sidebar</i> sebelah kiri.
Prioritas	1

4.5.2 Use Case Mencari Arsip Project

Use case ini digunakan untuk mencari data proyek yang dikembangkan di lembaga riset BPPT. Pencarian *project* ini dilakukan oleh aktor *manager* dan *engineer* yang keduanya merupakan *knowledge worker*. Terdapat beberapa *facet* yang dimiliki oleh *class project*, yaitu *facet name*, *field* (satuan kerja yang mengembangkan proyek tersebut), *type* (jenis proyek yang ada di BPPT), *year*, *person*, serta *product* yang dihasilkan dari proyek tersebut. Seluruh *facet* inilah yang dapat digunakan aktor dalam mencari proyek. Berikut tabel penjelasan *use case* Mencari Arsip *Project*.

Tabel 4.8 Use Case Mencari Arsip Project

Nama Use Case	Use Case Mencari Arsip Project
Aktor utama	<i>Knowledge worker</i> , yaitu <i>manager</i> dan <i>engineer</i>
Kondisi awal	Aktor utama memilih <i>class Project</i>
Kondisi akhir	Aktor utama telah mendapatkan data <i>project</i> , misalnya deksripsi proyek, pemilik proyek, kepala proyek, anggota pengembang, tahun dikerjakannya proyek, status proyek, satuan kerja yang berpartisipasi dalam proyek, produk-produk yang dihasilkan, serta tipe atau jenis proyek.
Skenario sukses utama	<ol style="list-style-type: none"> 1. Aktor berada pada halaman utama sistem. 2. Aktor memilih <i>class Project</i>. 3. Aktor dapat mencari proyek yang dikembangkan BPPT dengan memilih salah satu dari <i>facet</i> yang tersedia. <i>Facets</i> tersebut yaitu <i>name</i>, <i>field</i>, <i>type</i>, <i>year</i>, <i>person</i>, dan <i>product</i>. 4. Aktor memilih <i>instance</i> proyek yang ingin dicarinya.
Skenario alternatif	<ol style="list-style-type: none"> 1. Aktor berada pada halaman utama sistem. 2. Aktor memilih pencarian "All". 3. Aktor memilih <i>class Project</i>. 4. Aktor dapat memilih langsung <i>instance</i> dari <i>Project</i> atau dapat pula memilih pencarian <i>filter by subclass</i> dari <i>Project</i> berupa <i>BPPT Project</i>, <i>Cooperation</i>, dan <i>Participation</i> yang berada pada <i>sidebar</i> sebelah kiri.
Prioritas	1

4.5.3 Use Case Mencari Dokumen Berdasarkan Person

Use case ini digunakan oleh pengguna jika ia ingin mencari dokumen-dokumen yang ditulis oleh seseorang. Pencarian suatu dokumen dapat dilakukan dengan cara *filter by person* dan *project name*. Akan tetapi, pada pembahasan subbab ini terlebih dahulu dijelaskan mengenai pencarian berdasarkan *person*.

Tabel 4.9 Use Case Mencari Dokumen Berdasarkan Person

Nama Use Case	Use Case Mencari Dokumen Berdasarkan Person
Aktor utama	<i>Knowledge worker</i> , yaitu <i>manager</i> dan <i>engineer</i>
Kondisi awal	Aktor utama memilih <i>class Product</i>
Kondisi akhir	Aktor utama telah mendapatkan data dokumen terkait, misalnya jenis dokumen, penulis, dokumen hasil dari proyek mana, tanggal rilis, dan versi dokumen. Selain itu, aktor juga mendapatkan dokumen terkait yang dapat di <i>download</i> .
Skenario sukses utama	<ol style="list-style-type: none"> 1. Aktor berada pada halaman utama sistem. 2. Aktor memilih <i>class Product</i>. 3. Aktor dapat mencari dokumen yang ditulis oleh seseorang dengan memilih salah satu dari <i>facet</i> yang tersedia yaitu <i>facet author</i>. 4. Aktor memilih <i>instance</i> dokumen yang ingin dicarinya dengan memilih <i>facet Type "Document"</i>. Hal ini dikarenakan jika memilih langsung <i>instance</i> yang tampil tanpa memilih <i>facet</i>, aktor dapat memilih <i>instance product</i> selain dokumen. 5. Pilih <i>instance</i> dokumen kemudian aktor dapat melihat data dokumen sekaligus <i>download</i> dokumen tersebut.
Skenario alternatif	<ol style="list-style-type: none"> 1. Aktor berada pada halaman utama sistem. 2. Aktor memilih pencarian "All". 3. Aktor memilih <i>class Person</i>. 4. Aktor memilih <i>instance</i> nama penulis dari <i>Person</i>. 5. Akan tampil data-data <i>person</i> termasuk dokumen apa yang telah ditulisnya. Kemudian aktor memilih salah satu dokumen yang dicarinya. 6. Setelah memilih dokumen, akan tampilan data-data dokumen yang dicari dan aktor pun dapat <i>download</i> dokumen.
Prioritas	1

4.5.4 Use Case Mencari Dokumen Terkait Current Project Berdasarkan Project Name

Use case ini diperlukan ketika aktor ingin mencari dokumen yang berkaitan dengan proyek yang sedang dikembangkannya. Hal ini dikarenakan proyek dilanjutkan ke tahap berikutnya atau proyek yang baru dijalankan tetapi masih berkaitan sehingga dokumen pada sistem ini dapat berperan sebagai *knowledge reuse* bagi para pengembang. Berikut penjelasan *use case*.

Tabel 4.10 Use Case Mencari Dokumen Terkait Current Project Berdasarkan Project Name

Nama Use Case	Use Case Mencari Dokumen Terkait Current Project Berdasarkan Project Name
Aktor utama	<i>Knowledge worker</i> , yaitu <i>manager</i> dan <i>engineer</i>
Kondisi awal	Aktor utama memilih <i>class Product</i>
Kondisi akhir	Aktor utama telah mendapatkan data dokumen terkait, misalnya jenis dokumen, penulis, dokumen hasil dari proyek mana, tanggal rilis, dan versi dokumen. Selain itu, aktor juga mendapatkan dokumen terkait yang dapat di <i>download</i> .
Skenario sukses utama	<ol style="list-style-type: none"> 1. Aktor berada pada halaman utama sistem. 2. Aktor memilih <i>class Product</i>. 3. Aktor dapat mencari dokumen yang dihasilkan oleh proyek BPPT dengan memilih salah satu dari <i>facet</i> yang tersedia yaitu <i>facet project name</i>. 4. Aktor memilih <i>instance</i> dokumen yang ingin dicarinya dengan memilih <i>facet Type "Document"</i>. Hal ini dikarenakan jika memilih langsung <i>instance</i> yang tampil tanpa memilih <i>facet</i>, aktor dapat memilih <i>instance product</i> selain dokumen. 5. Pilih <i>instance</i> dokumen kemudian aktor dapat melihat data dokumen sekaligus <i>download</i> dokumen tersebut

Tabel 4.10 Use Case Mencari Dokumen Terkait *Current Project* Berdasarkan *Project Name* (lanjutan)

Skenario alternatif	<ol style="list-style-type: none"> 1. Aktor berada pada halaman utama sistem. 2. Aktor memilih pencarian “All”. 3. Aktor memilih <i>class Project</i>. 4. Aktor memilih <i>instance</i> nama proyek dari <i>Project</i>. 5. Akan tampil data-data <i>project</i> termasuk dokumen apa yang telah dihasilkannya. Kemudian aktor memilih salah satu dokumen yang dicarinya. 6. Setelah memilih dokumen, akan tampilan data-data dokumen yang dicari dan aktor pun dapat <i>download</i> dokumen.
Prioritas	1

4.5.5 Use Case Menambah Data *Person*, *Project*, *Product*, dan *Competency*.

Menambah data *class* yang ditampilkan pada sistem dilakukan oleh *engineer*. *Use case* ini merupakan salah satu *use case* untuk mengupdate data. Penambahan dilakukan secara otomatis pada sistem. Berikut penjelasan *use case*.

Tabel 4.11 Use Case Menambah Data *Person*, *Project*, *Product*, dan *Competency*

Nama Use Case	<i>Use Case Menambah Data Person, Project, Product, dan Competency.</i>
Aktor utama	<i>Engineer</i>
Kondisi awal	Aktor utama memilih <i>link</i> “Update Data”
Kondisi akhir	Aktor utama telah berhasil menambah data <i>person</i> , <i>product</i> , <i>project</i> , dan <i>competency</i> ke dalam sistem.
Skenario sukses utama	<ol style="list-style-type: none"> 1. Aktor berada pada halaman <i>administration area</i>. 2. Aktor memilih <i>link</i> “Update Data”, kemudian pilih “Add Data”. 3. Sistem menampilkan empat <i>class</i> yang dapat ditambah datanya. Aktor dapat memilih <i>class</i> manapun untuk melakukan penambahan data. 4. Setelah aktor masuk ke dalam <i>class</i> yang dipilih, sistem akan menampilkan atribut-atribut yang dimiliki oleh <i>class</i> tersebut. Berikut tampilan atribut masing-masing <i>class</i>: <ol style="list-style-type: none"> 4.a <i>Class Person</i> : <i>upload</i> foto, nama, e-mail, homepage, telepon, alamat, area bidang, anggota dari proyek apa, produk yang ditulis atau dikembangkan olehnya, kompetensi yang

Tabel 4.11 Use Case Menambah Data Person, Project, Product, dan Competency (lanjutan)

	<p>dimiliki, dan riwayat pendidikan.</p> <p>4.b <i>Class Project</i>: deskripsi proyek, pemilik proyek, kepala proyek, anggota proyek, tahun dikerjakan, status proyek, satuan kerja yang mengembangkan, produk yang dihasilkan, jenis proyek.</p> <p>4.c <i>Class Product</i>: deskripsi produk, jenis produk, penulis/pengembang, berasal dari proyek apa, tanggal rilis, versi, penerbit dan <i>volume</i> (jika tipe yang dipilih buku atau karya tulis), <i>upload</i> dokumen.</p> <p>4.d <i>Class Competency</i> : deskripsi kompetensi, orang yang memiliki kompetensi tersebut, jenis kompetensi.</p> <p>5. Setelah pengisian form <i>class</i> yang dipilihnya selesai, aktor memilih tombol “Add” dan data otomatis berhasil ditambahkan pada sistem.</p>
Skenario alternatif	-
Prioritas	1

4.5.6 Use Case Mengubah Data Person, Project, Product, dan Competency

Selain menambah data, melakukan *update* data dapat dilakukan dengan cara mengubah data pada empat *class* yang tersedia. Aktor yang dapat mengakses fitur ini adalah *engineer*. Berikut penjelasan *use case* mengubah data.

Tabel 4.12 Use Case Mengubah Data Person, Project, Product, dan Competency

Nama Use Case	Use Case Mengubah Data Person, Project, Product, dan Competency.
Aktor utama	<i>Engineer</i>
Kondisi awal	Aktor utama memilih <i>link</i> “Update Data”
Kondisi akhir	Aktor utama telah berhasil mengubah sebagian data <i>person</i> , <i>product</i> , <i>project</i> , dan <i>competency</i> ke dalam sistem.
Skenario sukses utama	<ol style="list-style-type: none"> 1. Aktor berada pada halaman <i>administration area</i>. 2. Aktor memilih <i>link</i> “Update Data”, kemudian pilih “Edit Data”. 3. Sistem menampilkan empat <i>class</i> yang dapat diubah datanya. Aktor dapat memilih <i>class</i> manapun untuk

Tabel 4.12 Use Case Mengubah Data Person, Project, Product, dan Competency (lanjutan)

	<p>melakukan perubahan data.</p> <ol style="list-style-type: none"> 4. Setelah aktor masuk ke dalam <i>class</i> yang dipilih, sistem akan menampilkan atribut-atribut yang dimiliki oleh <i>class</i> tersebut. Berikut tampilan atribut masing-masing <i>class</i>: <ol style="list-style-type: none"> 4.a <i>Class Person</i>: <i>upload</i> foto, nama, e-mail, homepage, telepon, alamat, area bidang, anggota dari proyek apa, produk yang ditulis atau dikembangkan olehnya, kompetensi yang dimiliki, dan riwayat pendidikan. 4.b <i>Class Project</i>: deskripsi proyek, pemilik proyek, kepala proyek, anggota proyek, tahun dikerjakan, status proyek, satuan kerja yang mengembangkan, produk yang dihasilkan, jenis proyek. 4.c <i>Class Product</i>: deskripsi produk, jenis produk, penulis/pengembang, berasal dari proyek apa, tanggal rilis, versi, penerbit dan <i>volume</i> (jika tipe yang dipilih buku atau karya tulis), <i>upload</i> dokumen. 4.d <i>Class Competency</i>: deskripsi kompetensi, orang yang memiliki kompetensi tersebut, jenis kompetensi. 5. Untuk melakukan perubahan, aktor dapat memilih data mana yang ingin diubahnya. <ol style="list-style-type: none"> 5.a Jika <i>class</i> yang dipilih adalah <i>class person</i>, aktor memilih dulu nama orang yang ingin diubahnya. Kemudian baru atribut lain selain nama dapat diubah. 5.b Jika <i>class</i> yang dipilih adalah <i>class project</i>, aktor memilih dulu proyek apa yang ingin diubah datanya. Kemudian baru atribut lain selain nama proyek dapat diubah. 5.c Jika <i>class</i> yang dipilih adalah <i>class product</i>, aktor memilih dulu nama produk yang ingin diubah. Lalu, atribut lain bisa diubah. 5.d Jika <i>class</i> yang dipilih adalah <i>class competency</i>, aktor memilih nama kompetensi yang ingin diubah, diikuti dengan mengubah atribut lainnya. 6. Setelah perubahan terhadap form <i>class</i> yang dipilihnya selesai, aktor memilih tombol “Edit” dan data otomatis berhasil diubah.
Skenario alternatif	-
Prioritas	1

4.5.7 Use Case Menghapus Data Person, Project, Product, dan Competency

Penghapusan data pada empat *class* dilakukan menyeluruh. *Use case* ini dilakukan oleh *engineer*. Sistem akan meng-*update* otomatis data yang telah dihapus pada halaman *web*. Berikut merupakan penjelasan *use case*.

Tabel 4.13 Use Case Menghapus Data Person, Project, Product, dan Competency

Nama Use Case	Use Case Menghapus Data Person, Project, Product, dan Competency.
Aktor utama	<i>Engineer</i>
Kondisi awal	Aktor utama memilih <i>link</i> "Update Data"
Kondisi akhir	Aktor utama telah berhasil menghapus seluruh data <i>person</i> , <i>product</i> , <i>project</i> , dan <i>competency</i> ke dalam sistem.
Skenario sukses utama	<ol style="list-style-type: none">1. Aktor berada pada halaman <i>administration area</i>.2. Aktor memilih <i>link</i> "Update Data", kemudian pilih "Delete Data".3. Sistem menampilkan empat <i>class</i> yang dapat dihapus datanya. Aktor dapat memilih <i>class</i> manapun untuk melakukan pengurangan data.4. Untuk melakukan penghapusan data, aktor dapat memilih data mana yang ingin dihapusnya.<ol style="list-style-type: none">4.a Jika <i>class</i> yang dipilih adalah <i>class person</i>, aktor memilih dulu nama orang yang ingin dihapusnya.4.b Jika <i>class</i> yang dipilih adalah <i>class project</i>, aktor memilih dulu proyek apa yang ingin dihapus datanya.4.c Jika <i>class</i> yang dipilih adalah <i>class product</i>, aktor memilih dulu nama produk yang ingin dihapus.4.d Jika <i>class</i> yang dipilih adalah <i>class competency</i>, aktor memilih nama kompetensi yang ingin dihapus.5. Setelah penghapusan data sebagian pada form <i>class</i> yang dipilihnya selesai, aktor memilih tombol "Delete" dan data otomatis berhasil dikurangi.
Skenario alternatif	-
Prioritas	1

4.5.8 Use Case Menyebarkan Artikel Bebas

Use case ini digunakan oleh seluruh *knowledge worker* dan berfungsi sebagai media komunikasi diantara mereka. Penyebaran (*sharing*) artikel bebas melalui sebuah *blog* yang berjalan secara *independent* . Dengan menggunakan *blog* ini, para pengguna dapat *posting*, menyebarkan, melihat serta membuat komentar artikel-artikel yang dapat dikategorikan sebagai artikel bebas ataupun artikel terkait dengan proyek yang dimiliki dan diikuti oleh BPPT. Berikut ini penjelasan *use case* menyebarkan artikel bebas.

Tabel 4.14 Use Case Menyebarkan Artikel Bebas

Nama Use Case	Use case menyebarkan artikel bebas
Aktor utama	<i>Knowledge worker</i> , yaitu <i>manager</i> dan <i>engineer</i>
Kondisi awal	Aktor utama memilih <i>link</i> BPPT Blog pada halaman sistem BPPT Intranet
Kondisi akhir	Aktor utama telah berhasil <i>posting</i> , menyebarkan dan melihat artikel-artikel bebas, serta memberikan komentar pada artikel tersebut.
Skenario sukses utama	<ol style="list-style-type: none">1. Pilih <i>link</i> BPPT <i>blog</i>, kemudian aktor memilih <i>login</i> yang terdapat pada <i>right-sidebar</i>.2. Pilih menu “Post” diikuti dengan “Add New”.Kemudian aktor menulis artikel bebas yang tidak terkait dengan proyek.3. Setelah selesai menulis, pilih kategori “Articles” agar artikel tersebut termasuk ke dalam kelompok artikel bebas.4. Pilih “Publish” untuk mempublikasikan tulisan di <i>blog</i> tersebut.

Tabel 4.14 Use Case Mengubah Data Person, Project, Product, dan Competency (lanjutan)

Skenario alternatif	<p>Setelah berhasil <i>posting</i> dan menyebarkan/mempublikasikan (<i>sharing info</i>), aktor pun dapat melihat artikel-artikel tersebut dan menambahkan komentar (<i>sharing komunikasi</i>) sesama <i>knowledge worker</i>.</p> <ol style="list-style-type: none"> 1. Pilih <i>link</i> BPPT <i>blog</i>, kemudian aktor dapat langsung melihat artikel-artikel yang telah di-<i>posting</i> oleh dirinya sendiri ataupun <i>knowledge worker</i> lainnya. 2. Pilih kategori “Articles” pada <i>right-sidebar</i> untuk melihat artikel-artikel yang dikategorikan sebagai artikel bebas (tidak berkaitan dengan proyek yang diikuti dan dikembangkan oleh BPPT). 3. Untuk memberikan komentar, pilih menu “Comments” yang terdapat di setiap artikel yang dipublikasi. Kemudian, aktor dapat menulis komentar (dapat berupa informasi, pertanyaan, ide, dll) sehingga komunikasi dapat terjadi melalui media <i>blog</i> ini.
Prioritas	1

4.5.9 Use Case Menyebarkan Artikel Terkait Proyek

Sama seperti *use case* menyebarkan artikel bebas, hanya saja konten yang berbeda yaitu artikel yang dapat di-*posting*, disebar, dilihat dan diberikan komentar adalah konten terkait dengan proyek yang dikerjakan dan diikuti oleh BPPT. Berikut penjelasannya pada tabel berikut ini.

Tabel 4.15 Use Case Menyebarkan Artikel Terkait Proyek

Nama Use Case	Use case menyebarkan artikel terkait proyek
Aktor utama	<i>Knowledge worker</i> , yaitu <i>manager</i> dan <i>engineer</i>
Kondisi awal	Aktor utama memilih <i>link</i> BPPT Blog pada halaman sistem BPPT Intranet
Kondisi akhir	Aktor utama telah berhasil <i>posting</i> , menyebarkan dan melihat artikel-artikel terkait proyek, serta memberikan komentar pada artikel tersebut.

Tabel 4.15 Use Case Menyebarkan Artikel Terkait Proyek (lanjutan)

Skenario sukses utama	<ol style="list-style-type: none"> 1. Pilih <i>link</i> BPPT <i>blog</i>, kemudian aktor memilih <i>login</i> yang terdapat pada <i>right-sidebar</i>. 2. Pilih menu “Post” diikuti dengan “Add New”.Kemudian aktor menulis artikel terkait dengan proyek. 3. Setelah selesai menulis, pilih kategori “Related Project” agar artikel tersebut termasuk ke dalam kelompok artikel yang berhubungan dengan proyek. 4. Pilih “Publish” untuk mempublikasikan tulisan di <i>blog</i> tersebut.
Skenario alternatif	<p>Setelah berhasil <i>posting</i> dan menyebarkan/mempublikasikan (<i>sharing info</i>), aktor pun dapat melihat artikel-artikel tersebut dan menambahkan komentar (<i>sharing komunikasi</i>) sesama <i>knowledge worker</i>.</p> <ol style="list-style-type: none"> 1. Pilih <i>link</i> BPPT <i>blog</i>, kemudian aktor dapat langsung melihat artikel-artikel yang telah di-<i>posting</i> oleh dirinya sendiri ataupun <i>knowledge worker</i> lainnya. 2. Pilih kategori “Related Project” pada <i>right-sidebar</i> untuk melihat artikel-artikel yang dikategorikan sebagai artikel terkait dengan proyek yang diikuti dan dikembangkan oleh BPPT. 3. Untuk memberikan komentar, pilih menu “Comments” yang terdapat di setiap artikel yang dipublikasi. Kemudian, aktor dapat menulis komentar (dapat berupa informasi, pertanyaan, ide, dll) sehingga komunikasi dapat terjadi melalui media <i>blog</i> ini.
Prioritas	1

4.5.10 Use Case Authentication dan Authorization

Use case ini digunakan sebagai otentikasi dan otorisasi (hak level akses) pada sistem BPPT intranet. Semua *knowledge worker* memiliki hak akses yang berbeda. Berikut penjelasannya.

Tabel 4.16 Use Case Authentication dan Authorization

Nama Use Case	Use Case Authentication dan Authorization
Aktor utama	<i>Knowledge worker</i> , yaitu <i>manager</i> dan <i>engineer</i>
Kondisi awal	Aktor utama masuk ke halaman “Administration Area”
Kondisi akhir	<i>Login</i> berhasil dan aktor memiliki hak akses yang berbeda.
Skenario sukses utama	<ol style="list-style-type: none">1. Sistem akan menampilkan isian data <i>login</i>.2. Aktor utama akan mengisi isian data <i>login</i> yang berisi <i>username</i> dan <i>password</i>.3. Sistem akan melakukan validasi isian data yang berupa <i>username</i> dan <i>password</i>.4. Aktor berhasil <i>login</i> dan mempunyai hak akses penuh terhadap sistem sesuai perannya masing-masing.
Skenario alternatif	Jika data <i>login</i> aktor tidak valid: <ol style="list-style-type: none">1. Sistem akan menampilkan pesan kesalahan yang berupa kesalahan <i>username</i> atau <i>password</i> ataupun kesalahan format input <i>username</i> atau <i>password</i> tersebut.2. Aktor dapat keluar <i>use case</i> atau mencoba memasukkan <i>username</i> dan <i>password</i> lagi
Prioritas	1

5. IMPLEMENTASI DAN EVALUASI

Bab ini akan menjelaskan mengenai beberapa langkah yang dikerjakan pada tahap implementasi pengembangan *knowledge management system*, yaitu BPPT Intranet, *use case* apa saja yang telah diimplementasikan kedalam sistem dan tampilan prototipe berbasis disain GUI, *testing* atau evaluasi berdasarkan pemenuhan analisis *requirement tracking*. Penjelasan langkah-langkah akan dirincikan ke dalam subbab-subbab berikut ini.

5.1 Persiapan *Instance Data*

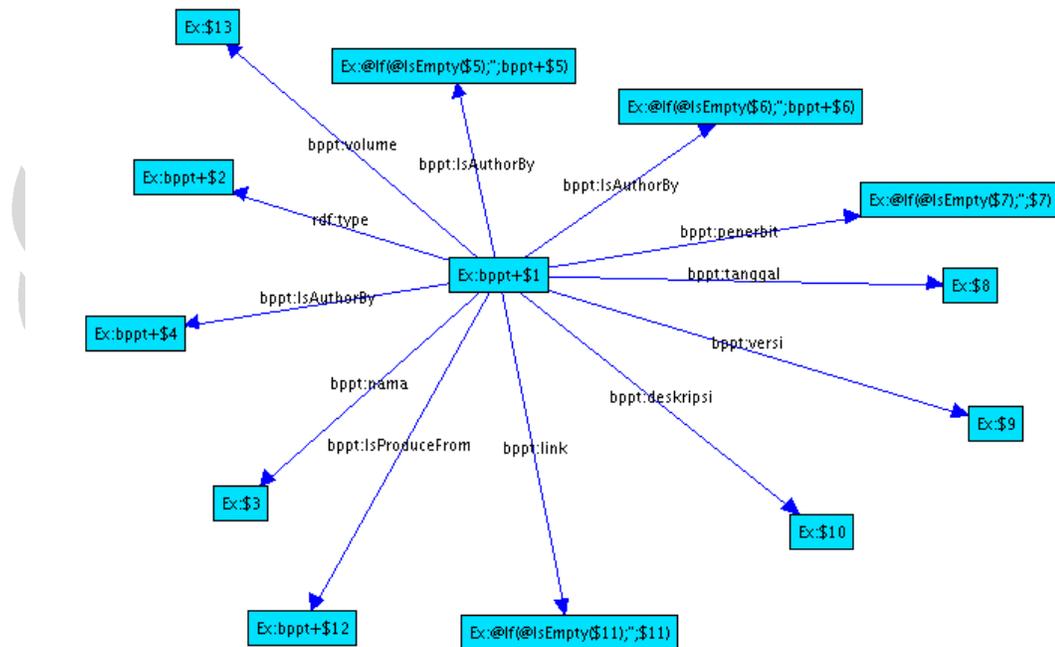
Setelah membuat disain ontologi BPPT, hal yang dilakukan adalah menyiapkan *instance data*. Dalam portalCore, *instance data* ditampilkan menggunakan format RDF dengan sintaks *.n3*. Pertama-tama, *instance* disimpan dalam file dengan format *spreadsheet* (*.xls*) yang diubah kembali formatnya menjadi (*.csv*), kemudian dengan menggunakan RDF123 yang berfungsi sebagai *converter* ke dalam format RDF. Akan tetapi, terlebih dahulu membuat *mapping graph* menggunakan RDF123 tersebut sehingga hasilnya disimpan dalam *output* berupa *.n3*.

Berdasarkan ontologi BPPT yang telah dibuat, terdapat lima *class* utama yang seluruhnya akan dibuatkan *instance data*, yaitu *class Person, Project, Product, Competency, dan Organization* sehingga akan ada lima *file* dengan tipe *.csv*. Kelima *file* ini berisikan atribut-atribut yang ditampilkan dalam suatu tabel. Masing-masing atribut inilah yang akan diisi dengan *instance data*. Tiap satu baris pada tabel menghasilkan satu *instance data*. Berikut ini merupakan atribut pada tiap-tiap *class* yang digunakan.

1. *Person* terdiri dari 14 atribut, yaitu id, nama, nip, email, homepage, telepon, alamat, bidang, kompetensi1, kompetensi2, kompetensi3, kompetensi4, foto, pendidikan.

2. *Project* terdiri dari 13 atribut, yaitu id, nama, deskripsi, kepala, pemilikproyek, tahun, anggota1, anggota2, anggota3, satuankerja, partisipan, statusproyek, tipe.
3. *Product* terdiri dari 13 atribut, yaitu id, tipe, nama, pengarang1, pengarang2, pengarang3, tanggal, versi, penerbit, volume, deskripsi, proyek, link.
4. *Competency* terdiri dari 3 atribut, yaitu id, tipe, nama.
5. *Organization*, terdiri dari 3 atribut, yaitu id, tipe, nama.

Kemudian, dibuatlah *map graph* dengan menggunakan RDF123. Berikut salah satu contoh *map graph* untuk class *Product*.



Gambar 5.1 Map Graph Class Product

Pengerjaan *map graph* tentunya memiliki format tertentu dalam penggunaan *namespace*, yaitu Ex:, \$n, *logical expression* @If(B;E;E) dan @IsEmpty(E). *Map graph* disimpan dengan tipe *file* .xgmml dan dapat di-convert menjadi format RDF.[2] Konversi dilakukan dengan memilih menu "Output" pada RDF123 dan

diikuti dengan memilih "View SpreadSheet in RDF". Lalu, simpan file dengan memilih "Display type: N3". Pada ontologi BPPT dihasilkan lima file dengan tipe .xgml dan .n3. Sintaks .n3 dapat dilihat pada contoh di bawah ini.

```
# Base: http://mybase
@prefix :      <#> .
@prefix bppt:  <http://www.owl-ontologies.com/Ontology1240287328.owl#>.
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdf123: <http://rdf123.umbc.edu/ns/> .

bppt:BSE
  a          bppt:Bidang_Sistem_Elektronika ;
  bppt:nama "Bidang Sistem Elektronika" .
```

5.2 Pendefinisian Rules

Pendefinisian *rules* bertujuan untuk proses *inference* disimpan dalam file bernama "BPPT.rules". Selain ontologi dan *instance* data, *rule* juga merupakan input pada portal BPPT Intranet. Pada portalCore, terdapat RDFS *Closure Rules* yang telah didefinisikan. Berikut beberapa *rules* yang telah terdefiniskan, antara lain:

```
[rdfs2: (?x ?p ?y), (?p rdfs:domain ?c) -> (?x rdf:type ?c)]
[rdfs3: (?x ?p ?y), (?p rdfs:range ?c) -> (?y rdf:type ?c)]
[rdfs5a: (?a rdfs:subPropertyOf ?b), (?b rdfs:subPropertyOf ?c) -> (?a
rdfs:subPropertyOf ?c)]
[rdfs5b: (?a rdf:type rdf:Property) -> (?a rdfs:subPropertyOf ?a)]
[rdfs6: (?a ?p ?b), (?p rdfs:subPropertyOf ?q) -> (?a ?q ?b)]
[rdfs7: (?a rdf:type rdfs:Class) -> (?a rdfs:subClassOf ?a)]
[rdfs8: (?a rdfs:subClassOf ?b), (?b rdfs:subClassOf ?c) -> (?a
rdfs:subClassOf ?c)]
[rdfs9: (?x rdfs:subClassOf ?y), (?a rdf:type ?x) -> (?a rdf:type ?y)]
```

- rdfs2 menjelaskan bahwa suatu relasi p menghubungkan x dengan y, jika relasi p memiliki batasan domain c, maka x *instance* dari c.
- rdfs3 menjelaskan bahwa suatu relasi p menghubungkan x dengan y, jika relasi p memiliki batasan *range* c, maka x *instance* dari c.
- rdfs5a menjelaskan bahwa jika a *subproperty* dari b, dan b *subproperty* dari c, maka a juga *subproperty* dari c.

- `rdfs5b` menjelaskan bahwa suatu *property* adalah *subproperty* dari dirinya sendiri.
- `rdfs6` menjelaskan bahwa suatu relasi *p* menghubungkan *a* dengan *b*, jika relasi *p* *subproperty* dari *q*, maka relasi *q* pun menghubungkan *a* dengan *b*.
- `rdfs7` menjelaskan bahwa suatu *class* adalah *subclass* dari dirinya sendiri.
- `rdfs8` menjelaskan bahwa jika *a* *subclass* *b*, dan *b* *subclass* *c*, maka *a* juga *subclass* *c*.
- `rdfs9` menjelaskan bahwa jika *x* *subclass* *y*, a *instance* *x*, maka *a* juga *instances* *y*.

Sedangkan *rules* yang didefinisikan khusus pada BPPT Intranet sesuai dengan disain ontologi yang *instance* data memiliki dua jenis *rules*, yaitu *transitive* dan *inverse*. Pendefinisian *rules* berikut disesuaikan dengan lima *class* yang ada. Berikut masing-masing penjelasannya, yaitu:

1. *Rule* untuk menyatakan seseorang bekerja pada bidang tertentu.

```
(?A bppt:IsDoing ?B), (?B bppt:DevelopedBy ?C) -> (?C bppt:IsWorkOnBy ?A) .
(?X bppt:IsWorkOnBy ?Y) -> (?Y bppt:WorksOn ?X) .
```

Rule ini menjelaskan bahwa jika *A* mengerjakan proyek *B*, dan proyek *B* dikembangkan oleh bidang *C*, maka bidang *C* tersebut dijalankan oleh *A*.

2. *Rule* untuk menyatakan proyek dilakukan oleh seseorang.

```
(?A bppt:Produce ?B), (?B bppt:IsAuthorBy ?C) -> (?C bppt:IsDoing ?A) .
(?X bppt:IsDoing ?Y) -> (?Y bppt:DoneBy ?X) .
(?A bppt:HeadBy ?B)-> (?A bppt:DoneBy ?B) .
(?A bppt:ParticipatedBy ?B)-> (?A bppt:DoneBy ?B) .
```

Rule ini menjelaskan bahwa jika proyek A menghasilkan produk B, dan produk B ditulis atau dikembangkan oleh seorang C, maka seorang C tersebut melakukan proyek A. *Rules* tambahan ada pada baris 3 dan 4 yaitu jika proyek A dikepalai oleh B, maka proyek A tersebut juga dilakukan oleh B. Lalu, jika proyek A diikuti oleh B, maka proyek A tersebut juga dilakukan oleh B. Dengan kata lain, *rules* tambahan ini menyatakan bahwa jika orang yang mengepalai atau berperan sebagai narasumber pada suatu proyek, ia pun juga mengerjakan proyek tersebut.

3. *Rule* untuk menyatakan seseorang menulis atau mengembangkan suatu produk.

```
(?A bppt:IsProduceFrom ?B), (?B bppt:DoneBy ?C) -> (?C bppt:IsAuthorBy ?A) .
(?X bppt:IsAuthorBy ?Y) -> (?Y bppt:AuthorOf ?X) .
```

Rule ini menjelaskan bahwa jika produk A dihasilkan dari suatu proyek B, dan proyek B dikerjakan oleh seorang C, maka seorang C menulis atau mengembangkan produk A.

4. *Rule* untuk membantu visualisasi dalam penamaan (*labeling*) pada *class Product, Project, Competency, dan Organization*.

```
(?A bppt:nama ?B) -> (?A rdfs:label ?B) .
(?A rdf:type bppt:Project), (?A bppt:nama ?B) -> (?A rdfs:label ?B) .
(?A rdf:type bppt:Product), (?A bppt:nama ?B) -> (?A rdfs:label ?B) .
(?A rdf:type bppt:Competency), (?A bppt:nama ?B) -> (?A rdfs:label ?B) .
(?A rdf:type bppt:Organization), (?A bppt:nama ?B) -> (?A rdfs:label ?B) .
```

5. Rule untuk inverse property.

```
(?A bppt:AuthorOf ?B) -> (?B bppt:IsAuthorBy ?A) .  
(?A bppt:Developes ?B) -> (?B bppt:DevelopedBy ?A) .  
(?A bppt:IsWorkOnBy ?B) -> (?B bppt:WorksOn ?A) .  
(?A bppt:Has ?B) -> (?B bppt:IsHad ?A) .  
(?A bppt:HeadOf ?B) -> (?B bppt:HeadBy ?A) .  
(?A bppt:IsProduceFrom ?B) -> (?B bppt:Produce ?A) .  
(?A bppt:ParticipatedBy ?B)-> (?B bppt:Participate ?A) .
```

5.3 Konfigurasi Portal BPPT Intranet

Konfigurasi utama yang dilakukan dalam penggunaan *component reuse* portalCore untuk pengembangan BPPT Intranet ini adalah mendefinisikan *datasource* yang dapat dilihat dalam `WEB-INF/config/sources.n3`, yaitu sebuah *file* RDF dengan sintaks N3. [19]

5.3.1 Pendefinisian *Datasource*

Setiap *datasource* harus didefinisikan sebagai *resource* dengan tipe `pcv:DataSource` beserta penjelasan *properties*. Deskripsi penamaan pada *datasource* juga dapat dilakukan pada konfigurasi file, sehingga akan tampil seperti ini: [19]

```
rdf:type pcv:DataSource ;  
  
    rdfs:label "Project" ;  
  
    pcv:encoding "p2";  
  
    pcv:order "20"^^xsd:integer ;  
  
    dc:description "Prototype Research Portal" ;
```

Pada konfigurasi *datasource* ini juga membutuhkan beberapa data. Data dikategorikan ke dalam tiga kelompok, yaitu ontologi, *instance* data, dan *closure rule*.

```
pcv:sourceURL <portal://data/personbppt.n3> ;  
  
pcv:sourceURL <portal://data/orgbppt.n3> ;  
  
pcv:sourceURL <portal://data/productbppt.n3> ;  
  
pcv:sourceURL <portal://data/competencybppt.n3> ;  
  
pcv:sourceURL <portal://data/projectbppt.n3> ;  
  
pcv:ontologySourceURL <portal://data/ontologyBPPT.owl> ;  
  
pcv:closureRulesURL <portal://data/BPPT.rules> ;
```

Pendefinisian *datasource* pada konfigurasi *datasource* BPPT Intranet diperlukan tiga *class*, yaitu *Person*, *Project*, dan *Product*. Selain itu, ada *datasource* tambahan yang menampilkan semua *class*. *Source code* dapat dilihat secara lengkap pada lampiran D.

1.3.2 Pendefinisian *Facets*

Setelah mendefinisikan *datasource*, konfigurasi *portalCore* dilakukan pada pendefinisian *facet* untuk navigasi data. Spesifikasi *facet* menggunakan *pcv:facet property* sedangkan *pcv:linkProp* untuk menampilkan objek yang dicari berdasarkan *property*. Terdapat dua *subclass* dari *pcv:Facet*, yaitu *pcv:HierarchicalFacet* (*resources* yang terindeks diklasifikasikan dengan pengaturan sktruktur hirarki dan *pcv:AlphaRangeFacet* (*resources* dikelompokkan berdasarkan huruf pertama dari *literal-valued property*). Sebuah *facet* yang dideklarasikan hanya *pcv:Facet* akan berfungsi sebagai *flat facet* yang akan menampilkan *instance* yang dihubungkan oleh sebuah *property*. [19] Ketiga macam *facet* tersebut ditampilkan dalam bentuk seperti ini

_ Competency [What is this facet?]

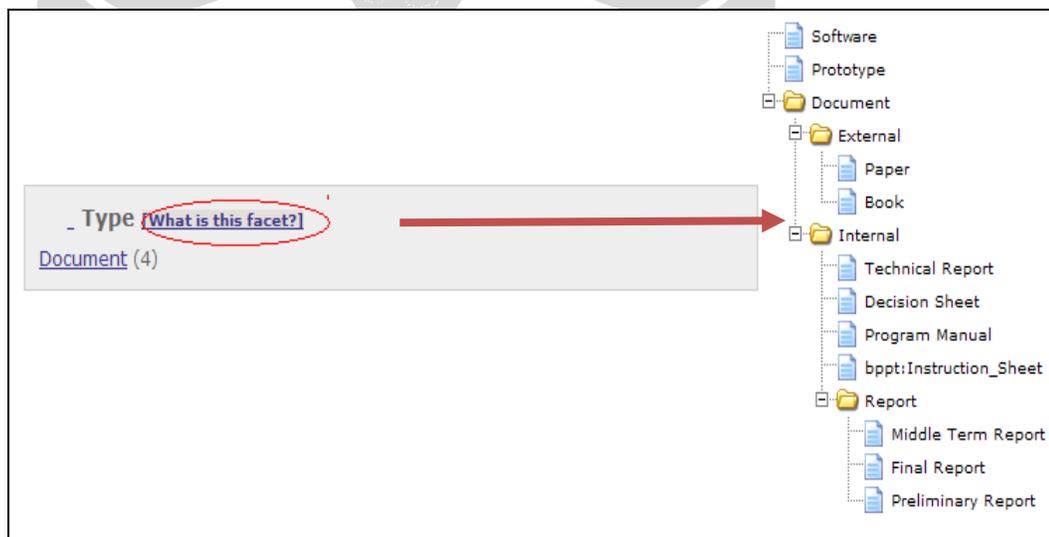
[Elektronika Analog](#) (6) | [OS:Windows, Linux \(instalasi konfigurasi\)](#) (4) | [PC Troubleshooting](#) (5) | [Pemrograman Perl](#) (4) | [Pemrograman Web](#) (4) | [Pemrograman java](#) (1) | [Troubleshooting PC \(Beginner\)](#) (3) **More options**

Gambar 5.2 Flat Facet

_ Name [What is this facet?]

[A*](#) (5) | [B*](#) (2) | [D*](#) (27) | [E*](#) (1) | [F*](#) (1) | [G*](#) (1) | [H*](#) (1) **More options**

Gambar 5.3 AlphaRangeFacet



Gambar 5.4 HierarchicalFacet

Pada BPPT Intranet pendefinisian *facet* dilakukan untuk tiga entitas, yaitu *Person*, *Project*, dan *Product*. Berikut salah satu contoh sintaks dalam pendefinisian *facet Project*.

```

pcv:facet bppt:p2titleFacet;
pcv:facet bppt:p2typeFacet;
pcv:facet bppt:p2personFacet;
pcv:facet bppt:p2fieldFacet;
pcv:facet bppt:p2yearFacet;
pcv:facet bppt:p2productFacet;

bppt:p2titleFacet a pcv:AlphaRangeFacet;
  rdfs:label "Title" ;
  pcv:linkProp bppt:nama;
  pcv:order "1"^^xsd:integer;
.

bppt:p2typeFacet a pcv:HierarchicalFacet;
  rdfs:label "Type";
  pcv:linkProp rdf:type;
  pcv:widenP rdfs:subClassOf ;
  pcv:facetBase bppt:Project ;
  pcv:order "2"^^xsd:integer;
.

```

Jumlah seluruh *facet* pada BPPT Intranet ada sebanyak 15 *flat facet*, 1 *alpharangefacet*, dan 1 *hierarchicalfacet* pada *datasource*. Tabel 5.1 berikut ini akan menerangkan *facet-facet* pada tiga entitas yang akan ditampilkan.

Tabel 5.1 Facets Untuk Person, Project, dan Product

<i>Classes</i>	<i>Facets</i>
<i>Person</i>	<i>Name, Product, Field, Competency</i>
<i>Project</i>	<i>Title, Field, Type, Year, Person, Product</i>
<i>Product</i>	<i>Type, Release Date, Title, Project, Author</i>

1.3.3 Pendefinisian *Templates*

Pada portalCore menggunakan *velocity template engine* untuk membangun tampilan *web* dari data RDF. *Templates* ini digunakan untuk memungkinkan tiap *datasource* menggunakan *template* yang berbeda untuk setiap fungsi. Pendefinisian *templates* sebanyak jumlah *class utama*, yaitu *Person*, *Project*, *Product*, *Competency*, dan *Organization*. Berikut tampilan pendefinisian *templates*.

```
pcv:template [a pcv:Template;
pcv:templateContext "page" ;
pcv:templatePath      <portal://templates/pagePersonBPPT.vm> ;
pcv:templateClass     bppt:Person;
];

pcv:template [a pcv:Template;
pcv:templateContext "page" ;
pcv:templatePath      <portal://templates/pageProjectBPPT.vm> ;
pcv:templateClass     bppt:Project;
];

pcv:template [a pcv:Template;
pcv:templateContext "page" ;
pcv:templatePath      <portal://templates/pageProductBPPT.vm> ;
pcv:templateClass     bppt:Product;
];

pcv:template [a pcv:Template;
pcv:templateContext "page" ;
pcv:templatePath      <portal://templates/pageOrgBPPT.vm> ;
pcv:templateClass     bppt:Organization;
];

pcv:template [a pcv:Template;
pcv:templateContext "page" ;
pcv:templatePath      <portal://templates/pageCompetencyBPPT.vm> ;
pcv:templateClass     bppt:Competency;
];
```

5.4 Visualisasi *Templates* BPPT Intranet

Pada tahap visualisasi *templates*, portalCore membagi tampilan menjadi tiga, yaitu *browse page*, *results page*, dan *resource page*. *Browse page* merupakan halaman yang menampilkan *facets* dari suatu *class*, *results page* adalah halaman hasil pemilihan *browse-facets*, sedangkan *resources page* menampilkan deskripsi dari suatu *resource*. Modifikasi *code* dilakukan pada *resource page*. [2] *Resources page* untuk BPPT Intranet sebanyak lima *file* sesuai dengan *class utama* pada disain

ontologi serta pada saat pendefinisian *facet* (dapat dilihat pada subbab 4.3), sebagai salah satu contoh *filenya* `portal://templates/pageCompetency.vm`. Berikut merupakan daftar *templates* dan *properties* yang ditampilkan pada BPPT Intranet yang dijelaskan pada tabel 5.2.

Tabel 5.2 Daftar *Templates*

<i>Templates</i>	<i>Object Properties</i>	<i>Datatypes Properties</i>
pagePersonBPPT.vm	bppt:WorksOn, bppt:HeadOf, bppt:IsDoing, bppt:AuthorOf, bppt:Has	bppt:foto, bppt:nama, bppt:email, bppt:homepage, bppt:telepon, bppt:alamat, bppt:pendidikan
pageProjectBPPT.vm	bppt:HeadBy, bppt:DoneBy, bppt:DevelopedBy, bppt:Produce, rdf:type, bppt:ParticipatedBy	bppt:nama, bppt:deskripsi, bppt:pemilikproyek, bppt:statusproyek, bppt:tahun
pageProductBPPT.vm	rdf:type, bppt:IsAuthorBy, bppt:IsProduceFrom	bppt:nama, bppt:deskripsi, bppt:tanggal, bppt:versi, bppt:penerbit, bppt:volume, bppt:link,
pageCompetencyBPPT.vm	bppt:IsHad, rdf:type	bppt:nama
pageOrgBPPT.vm	rdf:type	bppt:nama

Salah satu bagian dari *code* pada *resources page* yang telah dimodifikasi dapat dilihat di bawah ini. *Code* yang akan dijelaskan diambil dari *template file* `pageProjectBPPT.vm`.

```

<tr> <td width="10%"><b> Status: </b> </td>
      <td width="90%">
        $resource.getPropertyValue("bppt:statusproyek")
      </td>
    </tr>
<tr height="8"></tr>

#if ($resource.hasProperty("bppt:DevelopedBy"))
  <tr>
    <td><b> Field Area: </b></td>
    <td>
      #foreach ($p in $resource.findProperties("bppt:DevelopedBy"))
        #foreach ($v in $p.values)
          $v.render("leaf", $request)
        #end
      #end
    </td>
  </tr>
<tr height="8"></tr>
#end

```

Pada *code* di atas, *statement* yang akan ditampilkan berupa tabel. Terdapat *statement* yang menyatakan adanya tambahan dua baris pada tampilan *resources page*, yaitu *Status* dan *Field Area*. Pada bagian yang berisi atribut *Status*, jika *resource Project* memiliki *property* “*statusproyek*”, maka *status* akan ditambahkan dalam satu baris. Sedangkan pada bagian yang berisi atribut *Field Area*, jika *resource Project* memiliki *property* “*DevelopedBy*”, maka *field area* akan ditambahkan dalam satu baris. Masih dalam bagian *Field Area*, terdapat proses *recursive* untuk mendapatkan *value* dari *resource* yang dihubungkan oleh *property* “*DevelopedBy*” kemudian membuat *link* menuju halaman *resource* tersebut.

5.5 Hasil Prototipe BPPT Intranet

Berdasarkan Bab III Analisis dan Disain, terdapat beberapa *use case* yang sudah terimplementasi dengan baik dan ada pula *use case* yang belum terimplementasi tetapi telah dibuat prototipe berdasarkan disain GUI (*Graphic User Interface*).

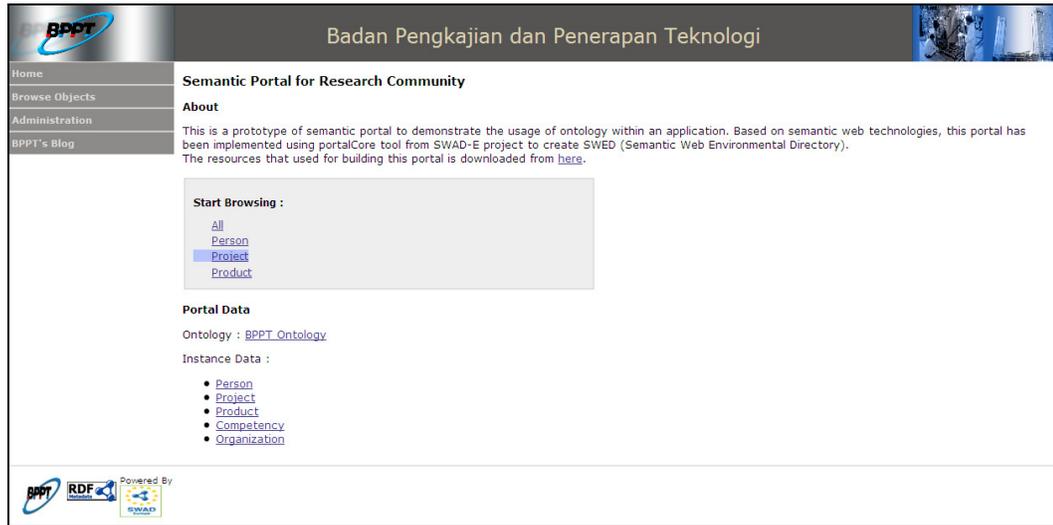
1.5.1 Prototipe Berdasarkan Implementasi *Component Reuse*

Implementasi *component reuse* portalCore dan *blog* Wordpress untuk BPPT Intranet diantaranya memiliki fungsi-fungsi *faceted browse*, *text search*, *refined search*, *download document*, *posting* artikel dan komentar. Semua fungsi-fungsi ini telah dapat digunakan dan dilihat melalui *use cases* berikut yang telah diimplementasikan. Berdasarkan tujuan pengembangan *knowledge management system* di BPPT, tujuan yang berhasil diimplementasikan dalam sistem ini adalah mendukung pekerjaan atau proyek tim yang terdistribusi, mendukung komunikasi antara individu (*knowledge worker*), *sharing* informasi dan pengetahuan berdasarkan proyek yang dikembangkan di BPPT (sebagian diimplementasi dan sebagian masih berupa prototipe GUI).

Pada tujuan mendukung pekerjaan atau proyek tim yang terdistribusi, *use cases* diantaranya *Use Case Mencari Arsip Project*, *Use Case Mencari Dokumen Berdasarkan Person*, *Use Case Mencari Person Berdasarkan Skill*. Sedangkan tujuan mendukung komunikasi antara individu, yaitu *Use Case Menyebarkan Artikel Bebas*, dan *Use Case Menyebarkan Artikel Terkait Proyek*. Lalu untuk tujuan *sharing* informasi dan pengetahuan baru satu *use case* yang sudah diimplementasikan, yaitu *Use Case Mencari Dokumen Terkait Current Project Berdasarkan Project Name*. Penjelasan *flow* berdasarkan skenario utama analisis *use case* dapat dilihat berikut ini.

➤ Use Case Mencari Arsip Project

- Aktor berada pada halaman utama sistem dan kemudian memilih *class Project*.



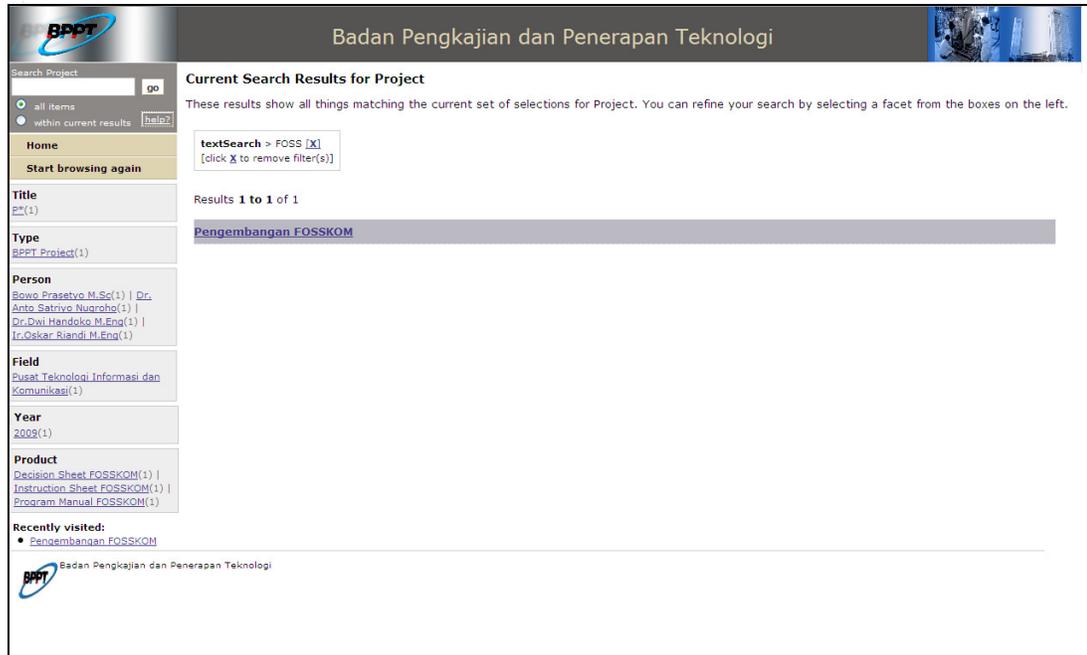
Gambar 5.5 Halaman Utama Pilih Class Project

- Aktor dapat mencari proyek yang dikembangkan BPPT dengan memilih salah satu dari *facet* yang tersedia. *Facets* tersebut yaitu *name*, *field*, *type*, *year*, *person*, dan *product*. Pencarian inilah yang disebut juga dengan *faceted browse*.



Gambar 5.6 Project's Facets

- Aktor memilih *instance* proyek yang ingin dicarinya dengan masuk ke *results page*.



Gambar 5.7 Result Page Project

- Aktor berhasil mendapatkan arsip proyek pada halaman *resource page*.



Gambar 5.8 Resource Page Project

➤ *Use Case Mencari Dokumen Berdasarkan Person*

- Aktor berada pada halaman utama sistem kemudian memilih *class Product*.



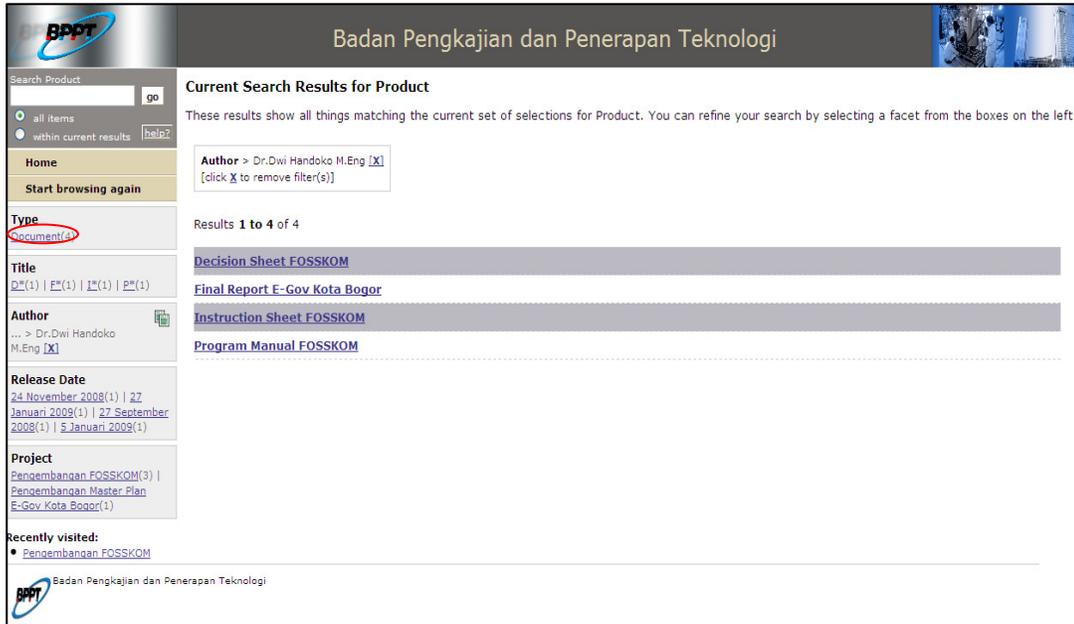
Gambar 5.9 Halaman Utama Pilih *Class Product*

- Aktor dapat mencari dokumen yang ditulis oleh seseorang dengan memilih salah satu dari *facet* yang tersedia yaitu *facet author*.



Gambar 5.10 Pilih *Facet Author*

- Aktor memilih *instance* dokumen yang ingin dicarinya dengan memilih *facet Type "Document"*. Hal ini dikarenakan jika memilih langsung *instance* yang tampil tanpa memilih *facet*, aktor dapat memilih *instance product* selain dokumen. Inilah yang dimaksud dengan *refined search*.



Gambar 5.11 Pilih *Type Document* pada *Result Page*

- Pilih *instance* dokumen kemudian aktor dapat melihat data dokumen sekaligus *download* dokumen tersebut.



Gambar 5.12 *Resource Page Download Document*

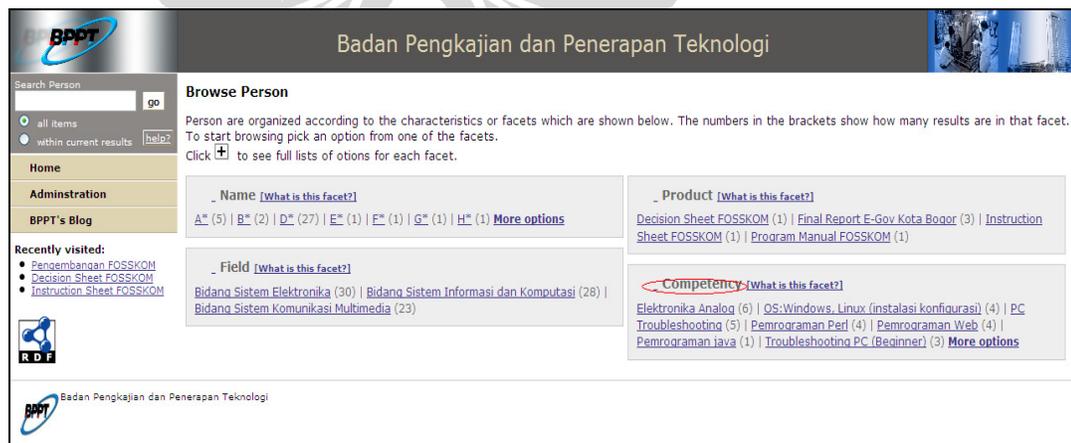
➤ Use Case Mencari Person Berdasarkan Skill

- Aktor berada pada halaman utama sistem, kemudian memilih *class Person*.



Gambar 5.13 Halaman Utama Pilih *Person*

- Untuk mencari berdasarkan *skill*, aktor memilih *facet Competency* dengan cara memilih salah satu *instance* dari *competency* yang muncul.



Gambar 5.14 Facet *Competency* pada *Browse Page*

- Aktor memperoleh data *person* yang dicarinya.



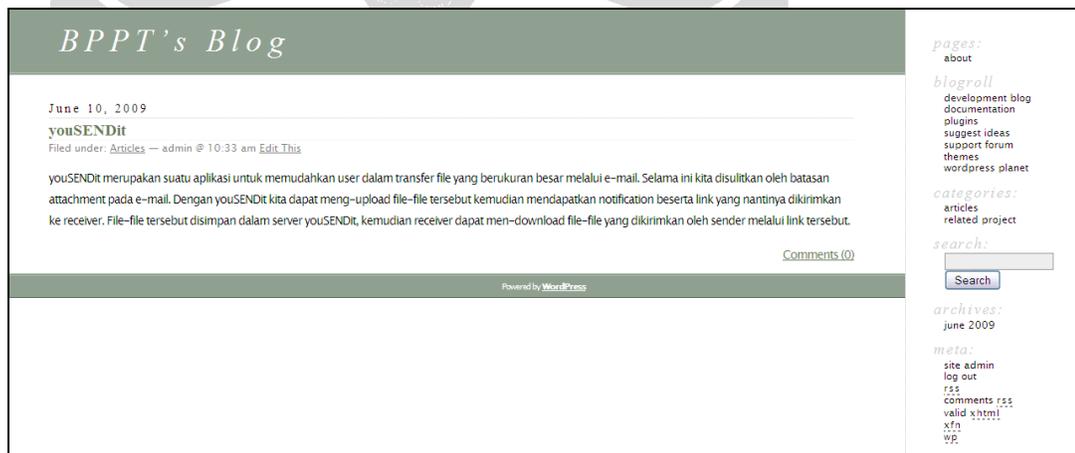
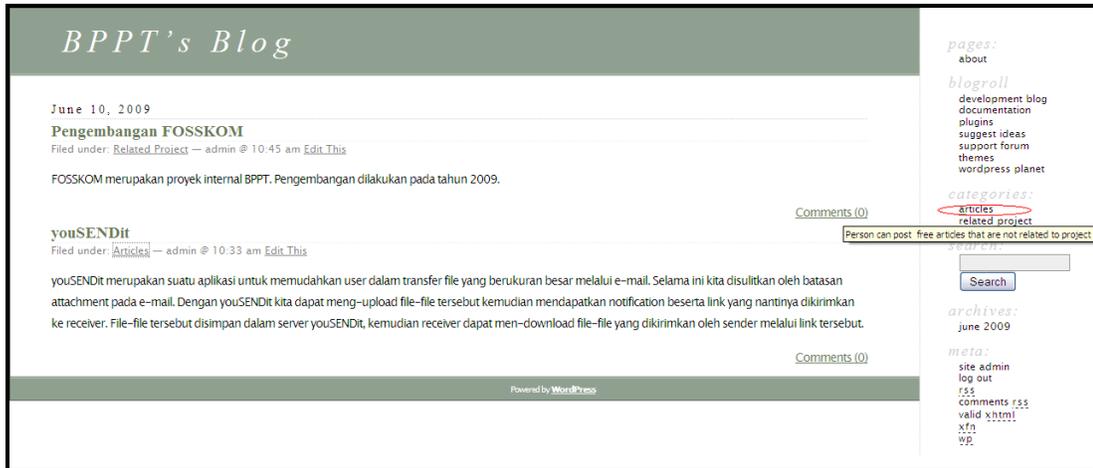
Gambar 5.15 Resource Page Person

- Use Case Menyebarkan Artikel Bebas (*flow* dari skenario alternatif *use case*)
 - Pilih *link* BPPT *blog*, kemudian aktor dapat langsung melihat artikel-artikel yang telah di-*posting* oleh dirinya sendiri ataupun *knowledge worker* lainnya.



Gambar 5.16 Pilih Link "BPPT Blog"

- Pilih kategori “Articles” pada *right-sidebar* untuk melihat artikel-artikel yang dikategorikan sebagai artikel bebas (tidak berkaitan dengan proyek yang diikuti dan dikembangkan oleh BPPT).



Gambar 5.17 Pilih Kategori "Articles"

- Untuk memberikan komentar, pilih menu “Comments” yang terdapat di setiap artikel yang dipublikasi. Kemudian, aktor dapat menulis komentar (dapat berupa informasi, pertanyaan, ide, dll) sehingga komunikasi dapat terjadi melalui media *blog* ini.



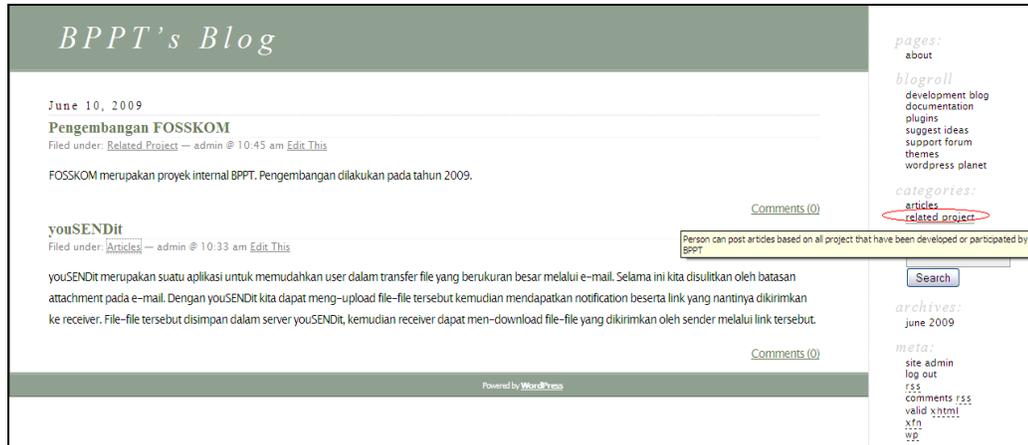
Gambar 5.19 Posting Komentar pada Artikel Bebas

- Use Case Menyebarkan Artikel Terkait Proyek (*flow* dari skenario alternatif *use case*)
 - Pilih *link* BPPT *blog*, kemudian aktor dapat langsung melihat artikel-artikel yang telah di-*posting* oleh dirinya sendiri ataupun *knowledge worker* lainnya.

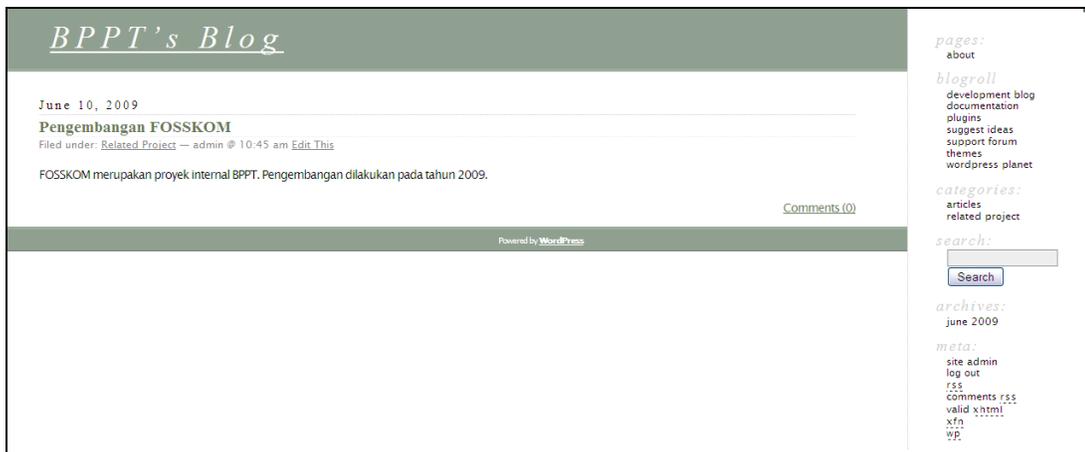


Gambar 5.20 Pilih *Link* "BPPT Blog"

- Pilih kategori “Related Project” pada *right-sidebar* untuk melihat artikel-artikel yang dikategorikan sebagai artikel terkait dengan proyek yang diikuti dan dikembangkan oleh BPPT.



Gambar 5.21 Pilih Kategori "Related Project"



Gambar 5.22 Hasil Pilihan Kategori "Related Project"

- Untuk memberikan komentar, pilih menu “Comments” yang terdapat di setiap artikel yang dipublikasi. Kemudian, aktor dapat menulis komentar (dapat berupa informasi, pertanyaan, ide, dll) sehingga komunikasi dapat terjadi melalui media *blog* ini.



Gambar 5.23 Posting Komentar pada Artikel Terkait Proyek

- *Use Case* Mencari Dokumen Terkait *Current Project* Berdasarkan *Project Name*
 - Aktor berada pada halaman utama sistem kemudian memilih *class Product*.



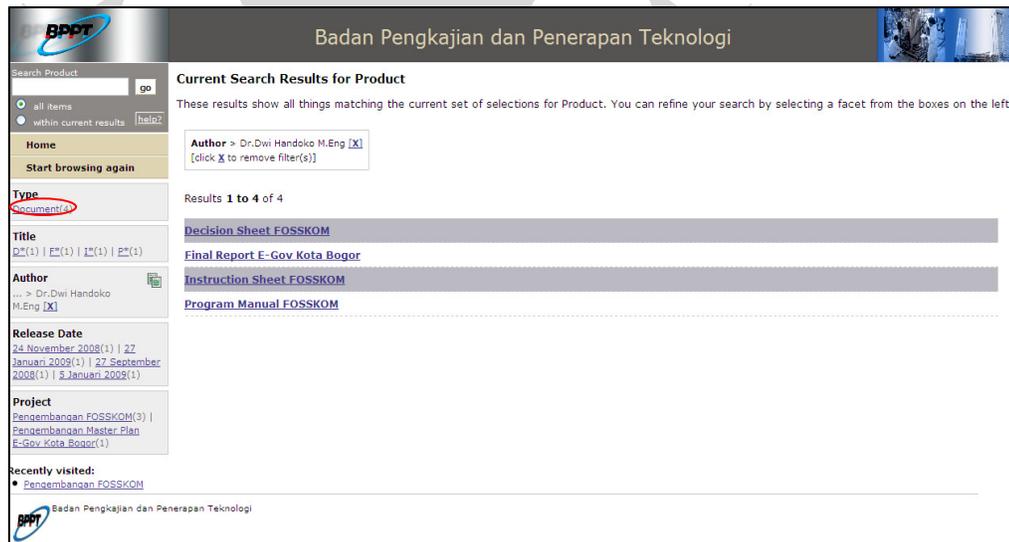
Gambar 5.24 Halaman Utama Pilih *Product*

- Aktor dapat mencari dokumen yang dihasilkan oleh proyek BPPT dengan memilih salah satu dari *facet* yang tersedia yaitu *facet project name*.



Gambar 5.25 Pilih Facet Project

- Aktor memilih *instance* dokumen yang ingin dicarinya dengan memilih *facet Type "Document"*. Hal ini dikarenakan jika memilih langsung *instance* yang tampil tanpa memilih *facet*, aktor dapat memilih *instance product* selain dokumen.



Gambar 5.26 Pilih Type Document

- Pilih *instance* dokumen kemudian aktor dapat melihat data dokumen sekaligus *download* dokumen tersebut .



Gambar 5.27 Tampilan *Resource Page Product*

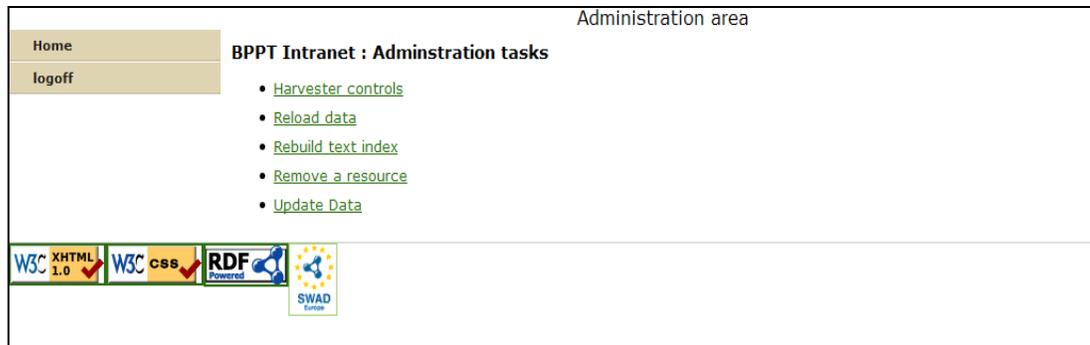
5.5.2 Prototipe Berdasarkan Disain GUI

Terdapat beberapa *use cases* pula yang masih belum diimplementasikan. Seperti yang sudah dijelaskan sebelumnya pada tujuan pengembangan, yaitu *sharing* informasi dan pengetahuan berdasarkan proyek yang dikembangkan di BPPT baru satu *use case* yang diimplementasikan, yaitu *Use Case Mencari Dokumen Terkait Current Project Berdasarkan Project Name* . Sedangkan *Use Case Update* (Tambah, Ubah, Hapus) Data *Person, Project, Product, dan Competency* baru berupa disain GUI. Berikut *flow* masing-masing *use cases*.

➤ Use Case Menambah Data *Person*, *Project*, *Product*, dan *Competency*

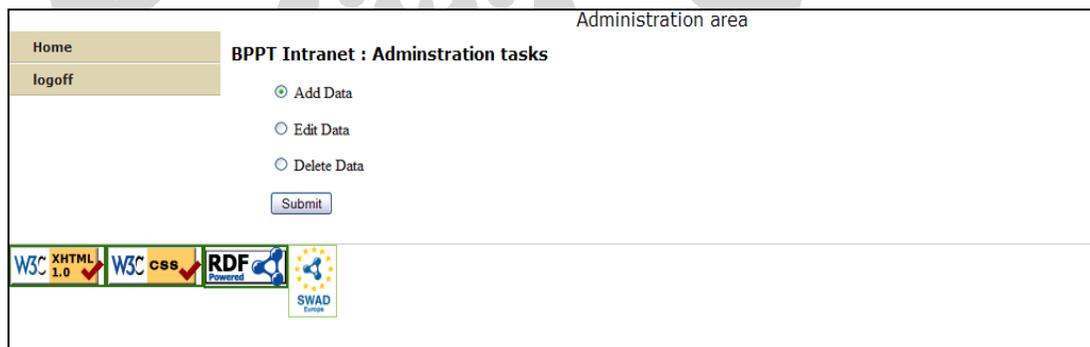
Contoh *flow* pada salah satu *class*, yaitu *class Project*.

- Aktor berada pada halaman administration area. Kemudian aktor memilih *link* “Update Data”.



Gambar 5.28 Pilih *Link* "Update Data"

- Lalu, aktor pilih “Add Data”.



Gambar 5.29 Pilih *Link* "Add Data"

- Sistem menampilkan empat *class* yang dapat ditambah datanya. Aktor dapat memilih *class Project* untuk melakukan penambahan data.

Gambar 5.30 Pilih Class Project

- Setelah aktor masuk ke dalam class *Project* yang dipilih, sistem akan menampilkan atribut-atribut deskripsi proyek, pemilik proyek, kepala proyek, anggota proyek, tahun dikerjakan, satuan kerja yang mengembangkan, produk yang dihasilkan, status proyek, jenis proyek, partisipan (diisi ketika memilih tipe *Participation Project*).

Gambar 5.31 Form Tambah Data Project

- Setelah pengisian form class yang dipilihnya selesai, aktor memilih tombol “Add” dan data otomatis berhasil ditambahkan pada sistem.

The screenshot shows a web interface for BPPT (Badan Pengkajian dan Penerapan Teknologi). The page title is 'Badan Pengkajian dan Penerapan Teknologi'. On the left, there is a navigation menu with 'Home', 'administration', and 'logout'. The main content area is titled 'Project' and contains a form with the following fields:

- Name: Pengembangan FOSS
- Description: FOSS merupakan proyek
- Owner: BPPT
- Head: Dr.Ali Badarudin (dropdown menu)
- Member: select member (dropdown menu)
- Year: 2008
- Status: Finish (dropdown menu)
- Field Area: select field area (dropdown menu)
- Product: select product (dropdown menu)
- Type: BPPT Project (dropdown menu)
- Participants: select participants (dropdown menu)

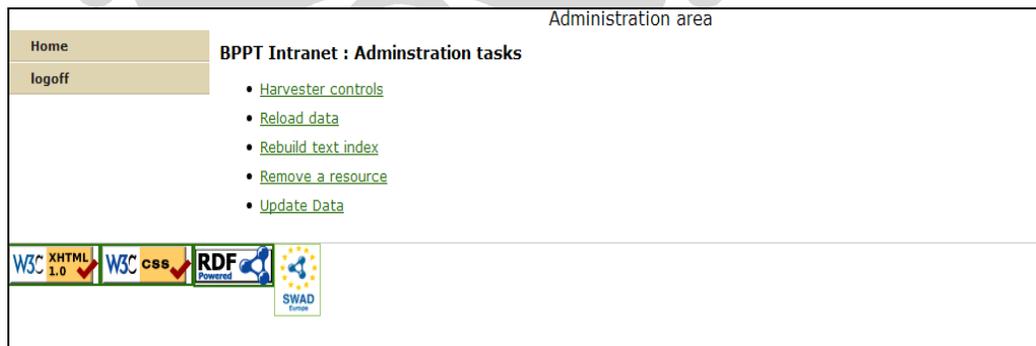
An 'Add' button is located at the bottom right of the form.

Gambar 5.32 Pengisian Form

- Use Case Mengubah Data *Person*, *Project*, *Product*, dan *Competency*

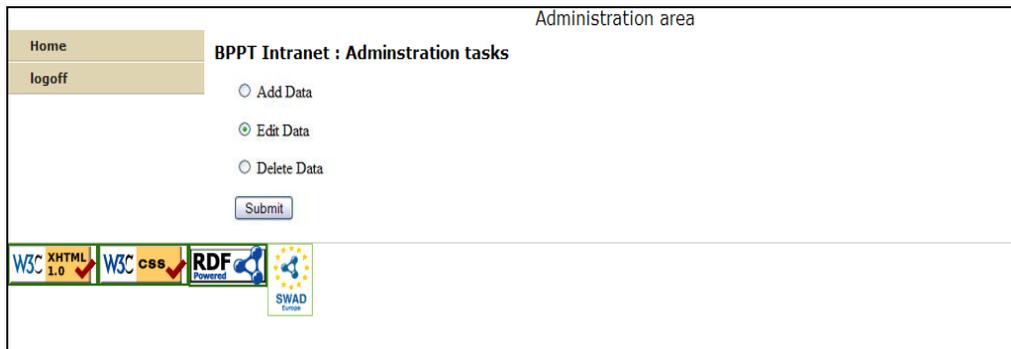
Contoh *flow* pada salah satu *class*, yaitu *class Project*.

- Aktor berada pada halaman administration area. Lalu, aktor memilih *link* “Update Data”.



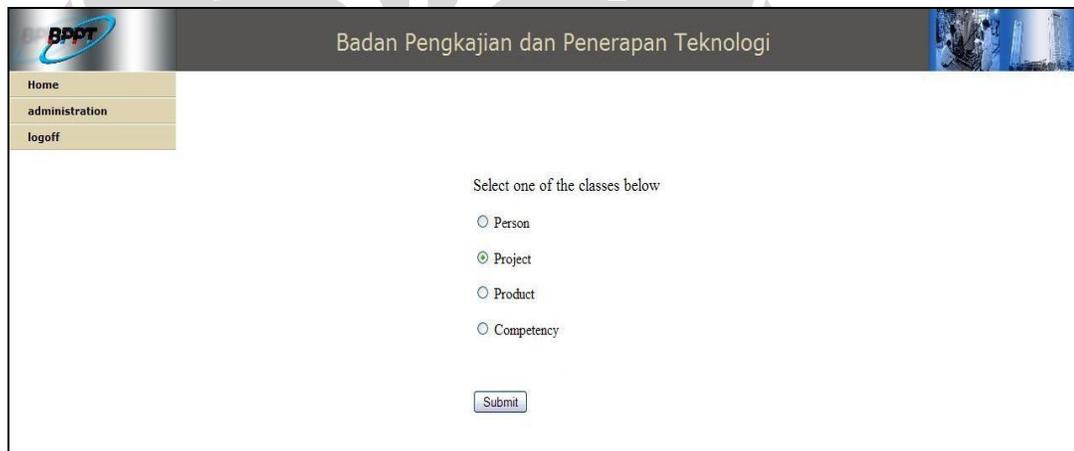
Gambar 5.33 Pilih *Link* "Update Data"

- Kemudian aktor pilih “Edit Data”.



Gambar 5.34 Pilih Link "Edit Data"

- Sistem menampilkan empat *class* yang dapat diubah datanya. Aktor dapat memilih *class Project* untuk melakukan perubahan data.



Gambar 5.35 Pilih Class Project

- Setelah aktor masuk ke dalam *class Project* yang dipilih, sistem akan menampilkan atribut-atribut deskripsi proyek, kepala proyek, pemilik proyek, anggota proyek, tahun dikerjakan, satuan kerja yang mengembangkan, produk yang dihasilkan, status proyek, jenis proyek, partisipan.

Gambar 5.36 Form Ubah Data Project

- Untuk melakukan pengubahan, aktor dapat memilih data mana yang ingin diubahnya. Jika *class* yang dipilih adalah *class project*, aktor memilih dulu proyek apa yang ingin diubah datanya. Kemudian baru atribut lain selain nama proyek dapat diubah.

Gambar 5.37 Pilih Project Name

Project

Name: Pengembangan FOSS

Description: Foss merupakan proyek...

Owner: BPPPT

Head: Dr.Ali Badarudin

Member: **select member**

Year: 2008

Status: Finish

Field Area: [select field area](#)

Product: [select product](#)

Type: BPPPT Project

Participants: [select participants](#)

[Edit](#)

Gambar 5.38 Ubah Data Member

Choose one or more person from the list below who work/works in the project

Dr.Dwi Handoko M.Eng Bowo Prasetyo M.Sc Dr. Anto Satriyo Nugroho

Ir.Oskar Riandi M.Eng Dr.Ali Badarudin

[Add](#)

Gambar 5.39 Pilih *Knowledge Worker* yang Terkait

- Setelah perubahan terhadap form *class* yang dipilihnya selesai, aktor memilih tombol “Edit” dan data otomatis berhasil diubah.

Gambar 5.40 Pilih *Button* "Edit" Jika Telah Selesai Mengubah Data

- *Use Case* Menghapus Data *Person*, *Project*, *Product*, dan *Competency*

Contoh *flow* pada salah satu *class*, yaitu *class Project*.

- Aktor berada pada halaman administration area diikuti dengan memilih *link* “Update Data”.

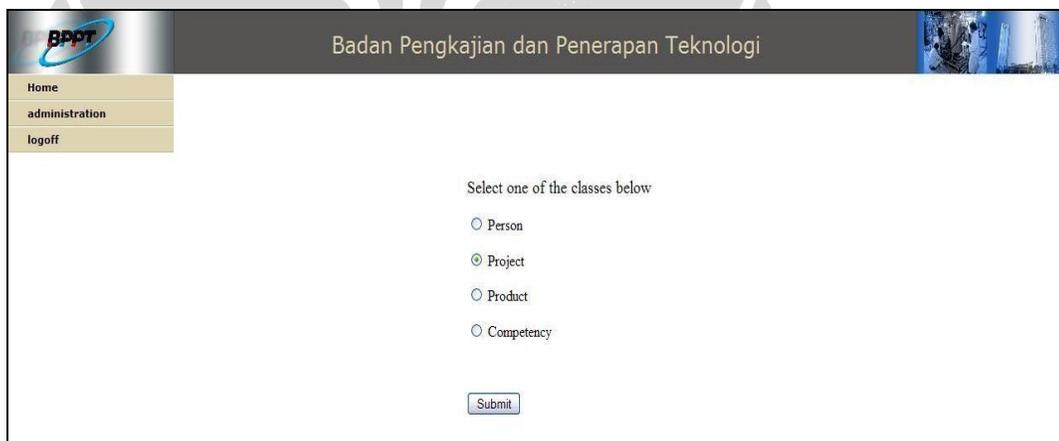
Gambar 5.41 Pilih *Link* "Update Data"

- Kemudian pilih “Delete Data”.



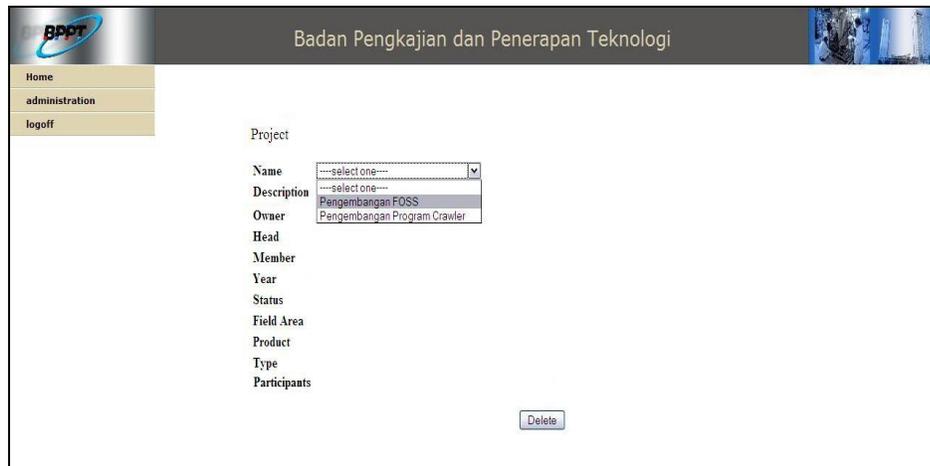
Gambar 5.42 Pilih Link "Delete Date"

- Sistem menampilkan empat *class* yang dapat dihapus datanya. Aktor dapat memilih *class Project* untuk melakukan pengurangan data.



Gambar 5.43 Pilih Class Project

- Untuk melakukan penghapusan data, aktor dapat memilih data mana yang ingin dihapus. Jika *class* yang dipilih adalah *class Project*, aktor memilih dulu proyek apa yang ingin dihapus datanya.



Gambar 5.44 Pilih Project Name yang Ingin Dihapus Datanya

- Setelah penghapusan data sebagian pada form *class* yang dipilihnya selesai, aktor memilih tombol "Delete" dan data otomatis berhasil dikurangi.



Gambar 5.45 Pilih Button "Delete"

Dari deskripsi-deskripsi ini hasil prototipe berdasarkan pada *existing framework* yang telah diimplementasikan dan juga prototipe yang masih berdasarkan disain GUI dapat dijelaskan dalam sebuah tabel yang berisikan pemetaan antara tujuan pengembangan *knowledge management system* dengan *use cases* yang didisain serta ketersediaan disain GUI dan implementasi.

Tabel 5.3 Hasil Prototipe BPPT Intranet

Tujuan KMS	Use Cases	Disain GUI	Implementasi
Mendukung pekerjaan atau proyek tim yang terdistribusi	• <i>Use case</i> mencari arsip <i>project</i>	Ya	Ya
	• <i>Use case</i> mencari dokumen berdasarkan <i>person</i>	Ya	Ya
	• <i>Use case</i> mencari <i>person</i> berdasarkan <i>skill</i>	Ya	Ya
Mendukung komunikasi antara individu	• <i>Use case</i> menyebarkan artikel bebas	Ya	Ya
	• <i>Use case</i> menyebarkan artikel terkait proyek	Ya	Ya
<i>Sharing</i> informasi dan pengetahuan berdasarkan proyek yang dikembangkan di BPPT.	• <i>Use case</i> mencari dokumen terkait <i>current project</i> berdasarkan <i>project name</i>	Ya	Ya
	• <i>Use case update</i> (menambah, mengubah, dan menghapus) data <i>person, project, product, dan competency</i> .	Ya	Tidak

1.6 *Functional Test*

Tahapan terakhir berupa *functional test* (uji fungsionalitas) yang bertujuan untuk pengujian terhadap fitur-fitur apakah sudah berfungsi atau belum dan apakah sudah memenuhi *requirement* atau belum. Tabel berikut berisi mengenai daftar

fitur-fitur yang diuji apakah sudah diimplementasi atau berfungsi dengan baik atau masih dalam bentuk disain GUI saja.

Tabel 5.4 Uji Fungsionalitas Fitur

<i>List of Features</i>	<i>Availability of GUI Design</i>	<i>Functioned</i>
<i>Searching Class Person</i> berdasarkan <i>facets Name, Product, Field, dan Competency</i>	√	√
<i>Searching Class Project</i> berdasarkan <i>facets Title, Field, Type, Year, Person, dan Product</i>	√	√
<i>Searching Class Product</i> berdasarkan <i>facets Type, Release Date, Title, Project, dan Author</i>	√	√
<i>Searching All Classes (Person, Project, Product, Competency, dan Organization)</i>	√	√
<i>Refined Search All Classes</i>	√	√
<i>Text Search</i> dengan aturan pencarian melalui mesin pencari	√	√
<i>Tree Search</i> pada <i>classes</i> yang memiliki <i>browse facet Type</i> , yaitu <i>class Project dan Product</i>	√	√
<i>Download</i> dokumen	√	√
<i>Update</i> yang meliputi menambah, mengubah, dan menghapus data <i>Person, Project, Product, dan Competency</i> (termasuk <i>upload</i> foto dan dokumen)	√	-
<i>Posting</i> artikel melalui <i>blog</i> sesuai kategori	√	√
Mencari Artikel berdasarkan kategori atau dengan <i>text search</i> melalui mesin pencari	√	√
Menambah komentar tiap artikel yang telah dipublikasikan	√	√