



Deusto

Facultad de Ingeniería
Universidad de Deusto

Ingeniaritza Fakultatea
Deustuko Unibertsitatea

Ingeniero en Informática **Informatikako Ingeniaria**

Proyecto fin de carrera
Karrera amaierako proiektua

Summary

The goal of this project is the combination of semantic, web and geospatial technologies for the development of an on-line GPS community.

A web application will be developed using NodeJS, a server side JavaScript programming platform, and AngularJS a single page application building framework based on the model-view-controller pattern. This system will allow users to share trails and points of interest which will be structured according to an ontology developed for this system and stored on a triplestore.

The semantic nature of the data will allow the users for a more intelligent and precise search, beyond the classic key word search. Besides, with the use of GeoSPARQL, a standard for the representation and querying of geospatial linked data the implicit relations among this data, such as which points of interest surround a trail, can be obtained.

The web application will allow the users to share their trails and enrich their information. Besides, a mobile application will give support to the system allowing the users to follow the trails of others, discovering information near them in real time and allowing them to post location aware notes on their actual location.

Descriptors

Semantic Web, GeoSPARQL, Linked Data, NodeJS, AngularJS.

Índice general

1	Introduction	1
1.1	Semantic Web	1
1.1.1	What is it?	1
1.1.2	The fundamentals	1
1.2	Entornos enumerados	4
1.2.1	Listas	4
1.2.2	Enumeraciones	4
1.2.3	Descripciones	4
1.3	Entornos flotantes	4
1.3.1	Imágenes	5
1.3.2	Tablas	5
1.3.3	Algoritmos	5
1.4	Referencias cruzadas y citas bibliográficas	6
A	Normativa	9
A.1	Introducción	9
A.2	Encuadernación	9
A.3	Portada	9
A.4	Papel	10
A.5	Impresión	10
A.6	Organización del contenido	11
A.6.1	Primera página	11
A.6.2	Segunda página	11
A.6.3	Resumen y descriptores	11
A.6.4	Índice	11
A.6.5	Memoria	11
A.6.6	Bibliografía	11
A.6.7	Planos	12

A.6.8 Anexos	12
A.6.9 Otros	12

Índice de figuras

Capítulo 1

1.1	An RDF graph representing Eric Miller	3
1.2	Logotipo de la Universidad de Deusto.	5
1.3	Ejemplo de subfiguras con los logotipos de la universidad, facultad y DeustoTech.	6

Índice de tablas

Capítulo 1

1.1	Ejemplo de un entorno tabulado.	6
1.2	Ejemplo de un entorno tabulado con varias subtablas.	7

Índice de algoritmos

Capítulo 1

1.1	«Hola mundo» en C.	5
1.2	Ejemplo más elaborado en Python.	5

1. INTRODUCTION

The goal of this report is to describe the process of design and implementation of the Final Undergraduate Project "SORELCOM: Design and Development of a semantic geospatial web and mobile application".

This project has been carried by Aimar Rodríguez Soto, 4th year student of an undergraduate degree in computer science engineering.

The different chapters that form this document are the following:

TODO

1.1. SEMANTIC WEB

1.1.1. What is it?

The semantic web is a extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation. [berners2001semantic]

The idea behind the semantic web is to bring meaning and structure to the contents of web pages, creating an environment where software agents can roam the web carrying out sophisticated tasks for end users. Thanks to the semantic web, this agents will be able to understand the relationships among the information on a web page, beyond simply identifying certain keywords.

This machine-readable web of data is based standard languages which allow a uniform representation of the information throughout the different semantic data sources. Through this common infrastructure it is possible to share and process information in a relatively simple way, enabling better solution to the usual problem of searching through the web of information. In short, the Semantic Web attempts to give meaning and structure to the contents on the web, allowing computers to better understand this information to do more useful work.

1.1.2. The fundamentals

The Semantic Web is based on standard formats used for a universal representation and manipulation of data. This standards are the following:

RDF: The Resource Description Framework (RDF) is a standard language for representing information resources in the World Wide Web. It is particularly intended to represent metadata about web resources, however, it by generalizing it can also be used to represent information about things that can be identified on the web. [w3crdf] RDF provides a framework for

1. INTRODUCTION

representing information which can be exchanged between application without it losing its meaning and it is intended for situation in which this information needs to be processed by machines instead of displayed to humans, thus making it fitting for the goals of The Semantic Web. The framework is based on identifying resources by using Uniform Resource Identifiers (URIs), and describing them in terms of properties and property values, which enables to represent statements about resources as a graph of nodes and arcs formed by the resources, their properties and their property values. For example, in figure 1.1 we can find a RDF graph representing a person named Eric Miller. Note that the resources itself as well as the properties and property values may be identified by URIs.

s

s

s

s

Este documento contiene una plantilla para la presentación de los Proyectos Fin de Grado de la Facultad de Ingeniería de la Universidad de Deusto. La plantilla se basa en tecnologías modernas y por lo tanto necesita una distribución de $\text{\LaTeX} 2_{\epsilon}$ reciente (MiKTeX 2.9 [miktex] o TeXLive 2012 [texlive]). En particular usamos:

\XeLaTeX o \LuaLaTeX : Debido a que en esta memoria se usan las fuentes Arial y Bookman Old Style que no incluidas generalmente en las distribuciones $\text{\LaTeX} 2_{\epsilon}$ es necesario recurrir a compiladores que soporte fuentes TTF o OTF como \XeLaTeX o \LuaLaTeX . Estos compiladores solo están presentes en las versiones más modernas de MiKTeX (2.9) y TeXLive (2012).

La fuente Arial viene por defecto en el sistema operativo Windows® mientras que en sistemas *nix se puede encontrar en el paquete `corefonts` [corefonts]. En los sistemas operativos derivados de Debian puede encontrarse en el paquete `ttf-mscorefonts-installer`.

Por otro lado, la fuente Bookman Old Style se puede encontrar en equipos con el sistema operativo Windows al instalar la suite ofimática Office®. En entornos *nix se puede descargar de [fuente]¹. La licencia nos permite redistribuir la fuente y por ello se encuentra en la carpeta `TeX-Gyre-Bonum`. Para instalarlas simplemente hay que copiarlas al directorio `~/ .fonts` creándolo si es necesario. Nótese que `~` representa el directorio del usuario (normalmente `/home/"username"` y que el directorio `.fonts` empieza por un punto y por lo tanto es un directorio oculto.

\BibLaTeX : En la actualidad es muy difícil generar bibliografías de calidad sin usar un gestor de registros bibliográficos. En esta plantilla se decidió usar el moderno gestor de referencias bibliográficas \BibLaTeX . Para su uso es necesario instalar tanto el paquete `biblatex` como el programa `biber`. La versión más reciente de MiKTeX (2.9) permite usar sin problemas estos paquetes. En entornos *nix es necesario instalar estos programas desde el gestor de paquetes.

\BibLaTeX es compatible con \BibTeX así como con los principales gestores bibliográficos comerciales como RefWorks® [refworks] o Mendeley® [mendeley] así como con los formatos de bases de datos bibliográficas RIS, Endnote™ XML and Zotero RDFXML.

¹ Obviamente, también se puede instalar en entornos Windows®.

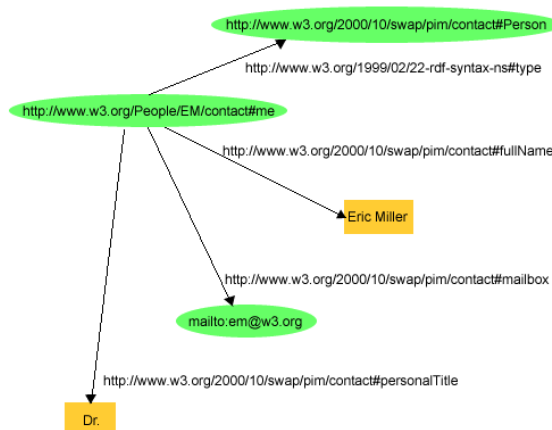


Figura 1.1: An RDF graph representing Eric Miller

source: <http://www.w3.org/TR/2004/REC-rdf-primer-20040210>

Para más información sobre como hacer bibliografías con Bib \LaTeX se puede consultar los diversos tutoriales que hay en la web, como por ejemplo el siguiente video [youtube] de YouTube®, el wikilibro [wikilibro] o este mismo documento.

Pygments: Los proyectos fin de grado de esta facultad contienen en muchas ocasiones código fuente o pseudocódigos. La representación de código fuente es un tema complejo pues se necesita «escapar» el texto a formatear del propio código \LaTeX . En esta plantilla hemos decidido usar programa `pygments` (de la misma manera que con la bibliografía) para realizar esta tarea. Sin embargo, en este caso no es necesario realizar una llamada externa como en el caso de Bib \LaTeX pues se la llamada se realiza al compilar el propio documento. Esto presenta un problema de seguridad en general y es por ello que es necesario introducir el parámetro `-shell-escape` en la llamada al compilador para permitir este comportamiento. Consúltase los manuales del IDE en uso para más información sobre este proceso.

Para su uso es necesario instalar un intérprete Python así como el programa `pygments` y el paquete `minted`. En entornos `*nix` no debería de haber ningún problema pues estas aplicaciones están en los repositorios de paquetes. En los sistemas derivados de Debian se puede instalar de la siguiente forma:

1. Instalar los paquetes: `python-setuptools` y `texlive-latex-extra`
2. Ejecutar el siguiente comando desde una terminal: `sudo easy_install pygments`

Para su instalación en entornos Windows™ se puede seguir la siguiente guía [pygments].

Hyperref: Dado que uno de los objetivos de las plantillas es generar documentos electrónicos de uniformes y de calidad se ha activado la generación de hipervínculos en el documento usando el paquete `hyperref`. De esta forma el fichero pdf resultante contiene un índice además de posibilidad de navegar por el documento como en una página web al ser todas las citas, referencias cruzadas, notas al pie de página y urls hiperenlaces. Además, el contenido de los distintos índices también son hiperenlaces.

En las siguientes secciones están dedicadas a hacer un resumen de las posibilidades de esta plantilla (y de \LaTeX en general).

1.2. ENTORNOS ENUMERADOS

Existen tres entornos enumerados que se pueden anidar de cualquiera de las maneras posibles. A continuación hacemos una ilustración de ellos.

1.2.1. Listas

- Soy un elemento de la lista.
- Soy otro elemento de la lista.
 - Soy un elemento anidado.
 - Soy otro elemento anidado.
- Soy el último elemento de la lista.

1.2.2. Enumeraciones

1. Soy el primer elemento de la enumeración.
 - a) Soy un elemento anidado de una enumeración.
 - b) Soy otro elemento anidado como enumeración.
2. Soy el segundo elemento de la enumeración.
 - Soy un elemento anidado de una lista.
 - Soy otro elemento anidado como lista.
3. Soy el último elemento enumeración.

1.2.3. Descripciones

En el capítulo ?? podemos ver un ejemplo de una descripción. Ponemos aquí otro por completitud.

Corta: Esta es una etiqueta pequeña con con algo de texto que habla sobre ella misma. El texto se tiene la consistencia de un párrafo con sangría francesa.

Etiqueta bastante larga: Esta es una etiqueta más larga. Nótese que el texto se sangra como en la etiqueta anterior y no a la altura de esta etiqueta.

Otra etiqueta: Esta etiqueta esta puesta para ver como quedan los textos que tiene más de un párrafo. Primero ponemos un poco de texto de relleno.

Ahora empezamos otro párrafo para que se vea como se respeta la distancia entre párrafos y la sangría continúa siendo la misma.

1.3. ENTORNOS FLOTANTES

Una de las grandes ventajas de \LaTeX se encuentra en su algoritmo de posicionamiento de objetos flotantes. Un objeto flotante es todo aquel que se le da libertad para moverse dentro del texto hasta ocupar una posición donde encaje. Típicamente los elementos flotantes suelen ser tablas e

imágenes que no puede dividirse entre páginas y por lo tanto deben «flotar» hasta su posición. A continuación presentamos diversos ejemplos de uso.

1.3.1. Imágenes

La figura 1.2 es una figura completamente flotante y por eso el algoritmo la coloca preferentemente al principio de la página. Sin embargo, a la figura 1.3 se le ha indicado que se prefiere que se coloque en la posición en la que está (nótese el `[h]`) y por lo tanto está en medio de la página. Además, esta figura está compuesta por tres subfiguras 1.3a, 1.3b y 1.3c.

1.3.2. Tablas

La tabla 1.1 muestra un ejemplo de como se pueden muestran datos tabulados. Nótese que al igual que antes es un entorno flotante y la tabla se desplazará hasta donde encaje. En la tabla 1.2 podemos ver un ejemplo con subtablas que, aunque le hemos indicado que lo introduzca en esta tabla es demasiado grande y flota hasta la siguiente página. Por supuesto, en este caso también podemos hacer referencias a las subtablas 1.2a y 1.2b.

1.3.3. Algoritmos

Finalmente terminamos con el último entorno flotante que falta: los algoritmos. Este entorno flotante es el más extraño pues necesita interpretar de forma completamente distinta el texto. Los algoritmos 1.1 y 1.2 hay dos ejemplos muy simples de sus posibilidades.

```
1  int main() {
2      printf("hello world");
3      return 0;
4  }
```

Algoritmo 1.1: «Hola mundo» en C.

```
1  # Returns  $\sum_{i=1}^n i$ 
2  def sum_from_one_to(n):
3      r = range(1, n + 1)
4      return sum(r)
```

Algoritmo 1.2: Ejemplo más elaborado en Python.



Figura 1.2: Logotipo de la Universidad de Deusto.



Figura 1.3: Ejemplo de subfiguras con los logotipos de la universidad, facultad y DeustoTech.

Tabla 1.1: Ejemplo de un entorno tabulado.

Nombre	Valor 1	Valor 2	Valor 3
Pedro	100	200	300
Juan	200	100	600
María	300	50	200
Carmen	400	10	7000

1.4. REFERENCIAS CRUZADAS Y CITAS BIBLIOGRÁFICAS

Una de las grandes características de \LaTeX es la facilidad para hacer tanto referencias cruzadas como citas bibliográficas. Como ya se ha visto anteriormente, para referenciar un capítulo, sección, etc. o un elemento flotante solo es necesario asignarle un `\label{}` y posteriormente citarle con `\ref{}`. Por ejemplo, para referenciar el primer capítulo basta con colocar la etiqueta `\label{cha:introduccion}` en cualquier punto del capítulo (preferiblemente después del título) y luego hacer referencia a él con `\ref{cha:introduccion}`. Es decir, hacemos referencia al capítulo ?? . De la misma forma podemos hacer referencia a los elementos flotantes como figuras, tablas y algoritmos.

Finalmente, la bibliografía se genera de forma automática conteniendo únicamente los trabajo explícitamente citados en la memoria². Para ello es necesario contar una base de datos bibliográfica en un formato reconocido. El más tradicional es el formato `.bib`. El fichero `bib.bib` incluido en esta memoria es un ejemplo básico con el que se han construido las citas de este documento.

A modo de ejemplo ponemos aquí una lista de citas de relleno para completar la bibliografía [article, conference, book], [phd] y [master].

²Hay varios mecanismos para incluir también otras obras. Véase la documentación de Bib \LaTeX .

Tabla 1.2: Ejemplo de un entorno tabulado con varias subtablas.

Nombre	Valor 1	Valor 2	Valor 3
Pedro	-100	200	a
Juan	2000	-100	b
María	300	50	c
Carmen	400	-10	f
(a) Valores de algo.			

Nombre	Valor 1	Valor 2	Valor 3
Pedro	100	α	300
Juan	γ	100	600
María	300	50	β
Carmen	ω	10	ε
(b) Valores de otra cosa.			

A. NORMATIVA

En este apéndice mostramos como dividir textos grandes en ficheros más manejables mediante el comando `\input{}`. Para ello incluiremos la normativa en este anexo.

A.1. INTRODUCCIÓN

En este documento se recogen las especificaciones más relevantes relacionadas con el formato de las memorias de los proyectos de la asignatura Proyecto Fin de Grado del Grado en Ingeniería Informática, Grado en Ingeniería en Tecnologías Industriales, Grado en Ingeniería en Electrónica Industrial y Automática, Grado en Ingeniería en Organización Industrial y Grado en Ingeniería en Tecnologías de Telecomunicación.

A.2. ENCUADERNACIÓN

La encuadernación será en cartulina (con cola y cinta negra), del color especificado a continuación según titulación (las tonalidades específicas están disponibles en la cartelera).

Titulación	Color de la encuadernación
Grado en Ingeniería Informática	Azul mediterráneo
Grado en Ingeniería en Organización Industrial	Crema
Grado en Ingeniería en Electrónica Industrial y Automática	Rojo Navidad
Grado en Ingeniería en Tecnologías de Telecomunicación	Verde Hierba
Grado en Ingeniería en Tecnologías Industriales	Butano

A.3. PORTADA

La portada consta de un encabezado que respeta la identidad corporativa de la Universidad de Deusto y que se facilita a los estudiantes a través de la página web de la asignatura. En dicho encabezado figuran en castellano y en euskera los datos relativos a la Universidad, la Facultad y la titulación correspondiente.

Dada la complejidad de definir literalmente toda la distribución de los diversos contenidos de la portada, se facilitan plantillas en formato «pdf editable» por cada titulación y que son de obligado uso.

El orden de los párrafos será:

A. NORMATIVA

- La expresión: Proyecto fin de grado.
- El título completo (sólo se escribirán con mayúscula la primera letra de la primera palabra y los nombres propios; no se pone punto al final)
- El nombre del estudiante
- El director o directora, donde se especificará dicho cargo, tal como figura en el ejemplo, es decir: Director: o Directora:. No se debe especificar el título de éste (doctor, licenciado o ingeniero)
- La fecha, tal como figura en el ejemplo. Bilbao, mes de año (en número)

A.4. PAPEL

Ha de emplearse papel del mismo color (blanco), calidad 80 gr. mínimo y normalmente suficiente y tamaño (UNE A-4), en todas las copias de la memoria.

A.5. IMPRESIÓN

Se empleará la plantilla «PlantillaPFG» disponible en la página web de la asignatura (en la sección de Plantillas) con el fin de respetar las fuentes, los márgenes y los encabezados y pies de páginas. Se suministra un ejemplo de su uso en formato Word y \LaTeX .

El uso de estas plantillas (y no otras) es de obligado cumplimiento para lograr una uniformidad en todas las titulaciones y memorias. Se deberán mantener el formato de las fuentes, títulos y márgenes establecidos en estos documentos.

Los márgenes han de ajustarse a las siguientes medidas: superior e inferior 3 cm, lateral interno 3'5 cm y lateral externo 2'5 cm.

El encabezado de las páginas pares indicará el título del capítulo en curso y el encabezado de las páginas impares llevará los términos «PROYECTO FIN DE GRADO», según plantilla. La numeración de las páginas será exterior. Las páginas en blanco no deben llevar numeración ni encabezado y las páginas posteriores deben seguir la secuencia de la numeración del documento.

Si hay alguna imagen, tabla o gráfico que deban ir en horizontal, se debe rotar la imagen, no la página completa. El encabezado y pie de página debe ser los mismos en toda la memoria, a excepción de los anexos, que pueden llevar un formato distinto, pero debe estar indicado claramente que se trata de anexos.

La presentación de la memoria debe cuidarse con especial esmero, procurando claridad, limpieza, uniformidad y ausencia de erratas. Se imprimirá a doble cara. Se evitarán los saltos de página innecesarios y la presencia de páginas que por una u otra razón estén prácticamente en blanco.

Las tres copias de la memoria que se entregan deben ser idénticas y la impresión se puede realizar en B/N o color.

A.6. ORGANIZACIÓN DEL CONTENIDO

A.6.1. Primera página

Debe ser exactamente igual que la portada, pero en papel blanco de la misma calidad que el resto de la memoria.

Esta página es la que deberá llevar a modo de confirmación la firma del director/a del proyecto. Esta firma deberá estar encima del nombre del director/a.

A.6.2. Segunda página

La segunda página es el reverso de la primera página, irá en blanco (sin número ni encabezado).

A.6.3. Resumen y descriptores

En la tercera página (anverso de la segunda hoja) debe aparecer un resumen del proyecto (de 200 a 250 palabras) y a continuación del mismo, entre tres y cinco descriptores (palabras clave) que ayuden a clasificar adecuadamente el proyecto.

La paginación anterior al capítulo 1 (portada, resumen e índice) se hará con números romanos (i, ii, iii, iv, ...). En la primera página, la portada, no debe aparecer el número.

A.6.4. Índice

El índice empezará en la página 5 (v), es decir, en el anverso de la tercera hoja.

A.6.5. Memoria

El capítulo 1 comenzará en la página 1. El resto de capítulos deberán comenzar también en página impar.

A.6.6. Bibliografía

Este será el último capítulo del documento e irá antes de los anexos. El formato de la bibliografía es el indicado en los ejemplos de memoria y plantillas disponibles, tanto en \LaTeX como en Word. De modo general, el formato a emplear será:

Para referenciar páginas web: «Título/Nombre de la Web», <http://www.google.es>, (consultado el 7/10/12).

Para referenciar libros: Autores separados por comas, «Título de la contribución/Libro», Editorial en cursiva, Año de la publicación.

A.6.7. Planos

Si los hubiera, la presentación de planos se hará de acuerdo a la normativa vigente (UNE-EN ISO 7200:2004) en esta área.

A.6.8. Anexos

Los anexos irán adjuntados al final de la memoria y podrán tener un formato libre diferente al del resto de la memoria.

A.6.9. Otros

No se establece ninguna otra restricción en cuanto al contenido y/o formato de las memorias, a excepción de las que se recogen en el presente documento.