ar

Autor: Francisco José Díaz Romero

Tutor: María Teresa Ariza Gómez

Plataforma Web para la creación de rutinas de entrenamiento para la App RutinaApp

Trabajo Fin de Grado

Grado en Ingeniería de las Tecnologías de

Telecomunicación

Departamento de Ingeniería Telemática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2017

Trabajo Fin de Grado

Grado en Ingeniería de Tecnologías de Telecomunicación

**Plataforma Web**

Autor:

Francisco José Díaz Romero

Tutor:

María Teresa Ariza Gómez

Profesor titular

Departamento de Ingeniería Telemática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2017

Trabajo Fin de Grado: Plataforma Web

|  |  |
| --- | --- |
| Autor: | Francisco José Díaz Romero |
| Tutor: | María Teresa Ariza Gómez |

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2017

El Secretario del Tribunal

*A mi familia*

*A mis amigos*

*A mis maestros*

Agradecimientos

Los estilos adoptados por nuestra Escuela y utilizada en este texto es una versión y adaptación a Word® del la versión LATEX que el Prof. Payán realizó para un libro que desde hace tiempo viene escribiendo para su asignatura. Por ello, la Escuela le está agradecida. Por otro lado, la adaptación se hizo sobre un formato que el prof. Aguilera arregló, basándose en su tesis doctoral. Su aportación ha sido muy relevante para que este formato vea la luz. Esta adaptación la llevamos a cabo el alumno Silvio Fernández, becario del Centro de Cálculo, y yo mismo, sobre un trabajo preliminar del alumno Julián José Pérez Arias.

A esta hoja de estilos se le incluyó unos nuevos diseños de portada. El diseño gráfico de las portadas para proyectos fin de grado, carrera y máster, está basado en el que el prof. Fernando García García, de la Facultad de Bellas Artes de nuestra Universidad, hiciera para los libros, o tesis, de la sección de publicación de nuestra Escuela. Nuestra Escuela le agradece que pusiera su arte y su trabajo a nuestra disposición.

*Juan José Murillo Fuentes*

*Subdirección de Comunicaciones y Recursos Comunes*

*Sevilla, 2013*

Resumen

En nuestra Escuela se producen un número considerable de documentos, tantos docentes como investigadores. Nuestros alumnos también contribuyen a esta producción a través de sus trabajos de fin de grado, máster y tesis. El objetivo de este material es facilitar la edición de todos estos documentos y a la vez fomentar nuestra imagen corporativa, facilitando la visibilidad y el reconocimiento de nuestro Centro.

Por otro lado, los alumnos muestras en la mayoría de las ocasiones, un desconocimiento absoluto de los programas informáticos de edición de texto. Esto es, de los elementos de un texto y de las herramientas para editarlos.

El objetivo de este documento es doble. Por un lado establecer un formato uniforme, mediante la adaptación de este fichero en formato electrónico, y por otro iniciar al usuario en las herramientas de edición de Microsoft Word.

Índice

Agradecimientos 9

Resumen 11

Índice 13

Índice de Tablas 15

Índice de Figuras 17

1 Introducción 19

1.1 Motivación y Objetivos 20

1.2 Contexto 20

1.3 Presentación Del Problema 23

1.4 Antecedentes 24

1.5 Descripción de la Solución 27

1.5.1 Objetivos específicos 27

1.5.2 Funcionalidades 27

1.5.3 Arquitectura 28

1.6 Estructura de la memoria 30

2 Tecnologías utilizadas 31

2.1 Servicio Web REST 31

2.1.1 Arquitectura REST 31

2.1.2 Uso en nuestro proyecto (REST) y ventajas respecto a otras arquitecturas 32

2.2 Formato para el intercambio de datos (JSON) 33

2.2.1 Introducción 33

2.2.2 Otros formatos (XML) 33

2.2.3 Uso en el proyecto 33

2.3 Tecnologías en el Servidor: Java Spring Framework 34

2.3.1 Introducción: Spring 34

2.3.2 Spring Boot 35

2.3.3 Spring Security 35

2.3.4 Spring Data 36

2.3.5 Maven y Gradle 37

2.3.6 Spring Beans 38

2.3.7 Inyección de dependencias (IoC) 38

2.3.8 Inversion de control 38

2.4 Persistencia: Base de datos 38

2.4.1 Introducción: Modelo Relacional 38

2.4.2 Gestor MySQL y uso en nuestro proyecto 38

2.5 Tecnologías para la interfaz web de usuario 39

2.5.1 Patron Modelo-Vista-Controlado (MVC) 39

2.5.2 Bootstrap (HTML5 + CSS3 + JavaScript) 39

2.5.3 Controlador de cliente: JQuery/Ajax 40

3 Estructura del Trabajo 42

Preliminares 43

3.1.1 Cubierta 43

3.1.2 Portada 43

3.1.3 Resumen 43

3.1.4 Prefacio o Introducción 43

3.1.5 Índice y/o Índice general 43

3.1.6 Lista de ilustraciones y tablas 44

3.1.7 Lista de abreviaturas y símbolos 44

3.1.8 Glosario 44

3.2 Texto principal 44

3.3 Bibliografía 44

3.4 Anexos 44

4 Uso de Estilos 45

4.1 Secciones 45

4.1.1 Subsección 45

4.1.2 Otra subsección 45

4.2 Otra sección 46

5 Otro Capítulo 11

5.1 Estilos de un documento 11

5.2 Títulos y Referencias Cruzadas 12

5.3 Versiones y Sistemas Operativos 12

5.3.1 Macintosh 12

5.3.2 Linux 12

5.4 Texto en inglés 13

5.5 Elementos básicos de un libro 13

5.6 Símbolos y fórmulas 13

5.7 Ecuaciones y MathType® 13

5.7.1 Fuentes 14

5.7.2 Epígrafes o citas célebres 14

5.7.3 Figuras y tablas 15

5.7.4 Hiperenlaces 15

5.7.5 Tabla de contenido 16

5.7.6 Índice de figuras, tablas y otros elementos 16

5.7.7 Formatos de páginas 16

5.7.8 Teoremas y otros elementos similares 16

5.7.9 Ejemplos 17

5.7.10 Índices de palabras y glosarios 17

5.8 Antes del documento 17

5.9 Fuente del texto 18

5.10 Cubierta 18

Referencias 20

Índice de Conceptos 22

Glosario 24

# **Índice de Tablas**

Tabla 2–1. Tipos de transmisión y frecuencia central 36

Tabla 3–1 Tipos de transmisión y frecuencia central 15

# **Índice de Figuras**

[**Figura 1.4‑1.1:** Aplicación Web de Runtastic. 25](#_Toc484270149)

[**Figura 1.1.4‑2:** Aplicación Móvil de Runtastic. 25](#_Toc484270150)

[**Figura 1.1.4‑3:** Aplicación Web de Strava. 26](#_Toc484270151)

[**Figura 1.1.4‑4:** Aplicación Móvil de Strava 26](#_Toc484270152)

[**Figura 1.0‑1:** Arquitectura de la aplicación Web para RutinaApp. 28](#_Toc484270153)

[Ilustración 2.1‑1 31](#_Toc484270154)

[Ilustración 2.2‑1 33](#_Toc484270155)

[Ilustración 2.2‑2 33](#_Toc484270156)

[Ilustración 2.4‑1 37](#_Toc484270157)

# Introducción

La tecnología no es en sí el fin sino el medio entre la sociedad del conocimiento y el desarrollo mundial. Anónimo

A

ctualmente, el desarrollo software se encuentra en un proceso revolucionario cuyo principal objetivo y tendencia es la implantación y desarrollo de software distribuido, donde se marcan como principales objetivos, la disponibilidad, usabilidad y funcionalidad de cualquier aplicación a través de internet. Es aquí donde entran en juego el desarrollo de aplicaciones o servicios Web, sustituyendo en gran medida al desarrollo de aplicaciones y servicios de escritorio.

Muchos son los aspectos a tener en cuenta, entre los que cabe destacar la eficiencia, la velocidad de respuesta y de navegación, los cuales para algunos servicios Web son excelentes, pero para otros aún se encuentran en un proceso de mejora. Esto es debido a que gran cantidad de aplicaciones requieren de un buen diseño donde se optimicen el uso de los recursos disponibles, como el ancho de banda, memoria, carga del servidor, respaldo, seguridad, etc.

## 1.1 Motivación y Objetivos

Este proyecto es una línea de mejora y continuación de otro proyecto en el que desarrolla una aplicación móvil para la gestión de rutinas de entrenamiento. El proyecto que nos ocupa, pretende dar soporte web a dicha aplicación móvil, proporcionando las funcionalidades necesarias para la creación y gestión de rutinas de entrenamiento, las cuales serán usadas por la aplicación móvil.

Dicha aplicación Web estará compuesta por un conjunto de servicios Web que proporcionaran las funcionalidades necesarias para tal fin, cuyo desarrollo web se implementara con el uso de un framework llamado Spring, que permitirá el despliegue, tanto en máquinas locales como en la Cloud, de nuestra aplicación web desde un solo fichero, que detallaremos más adelante.

Toda la aplicación web se desarrolla siguiendo los patrones que se esperan cumplir para un proyecto de esta universidad, aunque hay ciertos aspectos a tener en cuenta que provocan algunas variaciones respecto al estándar de un proyecto software.

Atendiendo a la mención anterior, dichas variaciones centran su atención en el uso de un servicio Web REST desarrollado con Spring framework, usando como lenguaje de programación Java. Durante el desarrollo de este servicio Web, se han hecho uso de ciertas librerías nativas de Java, además del manejo de tipos de datos simples, aprovechando al máximo las capacidades que nos proporciona Java y su framework de Spring.

Finalmente, son muchas las razones por las cuales he decidido adentrarme en el mundo de las aplicaciones web, ya que a lo largo del grado, ha sido lo que más curiosidad e interés me ha generado, viendo en este proyecto una oportunidad para aprender e investigar en tecnologías orientadas a los servicios Web. Otra de las razones es mi pasión y dedicación hacia el deporte en mi vida, lo cual ha hecho que este interés sea mayor debido a la relación directa que guarda este proyecto con el mismo.

Una vez mostradas las motivaciones y objetivos que nos han llevado a realizar este proyecto, detallaremos a continuación el problema a resolver y la solución que nosotros proponemos, además de los antecedentes relacionados con dicha tarea

## Contexto

Durante los últimos años, la evolución tecnológica gira principalmente en torno a Internet, hasta tal punto de que dicha herramienta está presente cada vez más en nuestras vidas y en todas las actividades que desempeñamos en el día a día.

Poco a poco, la necesidad de disponer de toda la diversidad funcional y tecnológica que esta herramienta nos proporciona, ligado a la necesidad de comunicarnos y estar conectados continuamente, evoluciona en paralelo, entre otras, la tecnología móvil. Es aquí donde ocupa la mayor parte del foco de atención, los dispositivos móviles, principalmente los Smartphones.

Ambas herramientas en conjunto nos permiten el desarrollo de muchas de nuestras actividades desde cualquier lugar, ya que a día de hoy muchas de estas actividades están ligadas a la tecnología y principalmente a Internet, ya sean actividades de ocio, de trabajo, de docencia, etc.

Atendiendo a estas necesidades y expectativas de la sociedad, nace el concepto de aplicación web o servicio web, donde se permite la interacción con servicios o herramientas tecnológicas ligadas a nuestras actividades cotidianas desde cualquier lugar y con una disponibilidad de recursos continua y eficiente.

Observando la tendencia y los intereses sociales en la actualidad, podemos observar que la imagen juega un papel primordial en muchos aspectos sociales, lo cual nos lleva a tender hacia un hábito de vida e imagen física saludables, donde como medio para alcanzar dicho fin aparecen la alimentación y el deporte.

Por otro lado, el rimo de vida de la sociedad, cada vez es más acelerado y exigente, y la precariedad económica que se ha sufrido durante los últimos años a nivel mundial nos impide a veces poder acudir a centros de entrenamiento y de nutrición especializados. Estos obstáculos nos llevan a buscar alternativas.

Aquí nacen entre muchas otras, las aplicaciones móviles o plataformas web para el seguimiento de nuestro entrenamiento físico, las cuales nos permiten personalizar nuestros entrenamientos y poderlos realizar en casa sin necesidad de acudir al gimnasio.

Estas aplicaciones nos enseñan además a seguir un hábito de entrenamiento saludable, ya que la correcta ejecución de los ejercicios físicos es muy importante para evitar lesiones futuras. También aportan funcionalidades de monitorización y seguimiento del ejercicio físico, de forma que para conseguir un mínimo estado de forma no es necesaria la presencia de un entrenador o monitor.

|  |
| --- |
| Z:\home\fran\Descargas\mobile-vs-desktop-web-users.png  **Figura 1.1:** Fuente de datos, Morgan Stanley |
|  |

## Presentación Del Problema

Como principal problema que se plantea en este proyecto es proponer una solución Web a una aplicación móvil desarrollada en Android para la gestión y creación de rutinas de entrenamiento.

La gran mayoría de aplicaciones existentes en el mercado, dedicadas al seguimiento de tareas físicas, imponen ciertas limitaciones en el número de rutinas de entrenamiento y de ejercicios, además de que son aplicaciones muy genéricas en cuanto al área de entrenamiento.

Es por tanto que se desea mejorar o enfatizar en estas carencias de otras aplicaciones para intentar mejorarlas en nuestra solución.

Persiguiendo lo mencionado previamente, en nuestra plataforma Web, el propio usuario que hace uso de la plataforma es el encargado de crear, elegir y descargar las rutinas que desee. Puede elegir si solo desea rutinas especializadas en una zona muscular o en varias, el nivel de forma del usuario y otros aspectos descriptivos del ejercicio, como una descripción detallada o videos relacionados con el ejercicio físico.

En resumen, la idea de esta página sería que cualquier usuario pueda ser productor y consumidor de rutinas, el límite en el número sólo lo imponen los propios usuarios. Además, en esta plataforma se permitirá al usuario la descarga de las rutinas que desee para poder reproducirlas en la aplicación móvil, en formato .zip.

Con la combinación de la aplicación móvil y la plataforma web, conseguimos que mediante una aplicación web podamos crear nuestras propias rutinas y personalizarlas, para posteriormente descargarlas y reproducirlas en la aplicación móvil.

En comparación con otras opciones existentes en el mercado, la mayoría de las aplicaciones móviles requieren de conexión a internet para reproducir las rutinas de entrenamiento desde un servidor externo, por lo que si no se dispone de buena conexión a internet, no se consigue una calidad mínima del servicio.

El objetivo es que la aplicación funcione siempre en tiempo real y en cualquier lugar donde se encuentre el usuario, además de poder crear sus propias rutinas de entrenamiento y que estas estén disponible siempre y desde cualquier lugar para poder ser descargadas a través de la aplicación Web, y reproducidas localmente desde cualquier dispositivo móvil Android.

## Antecedentes

Como antecedentes principales a nuestro proyecto, debemos destacar principalmente la siguiente línea de trabajo:

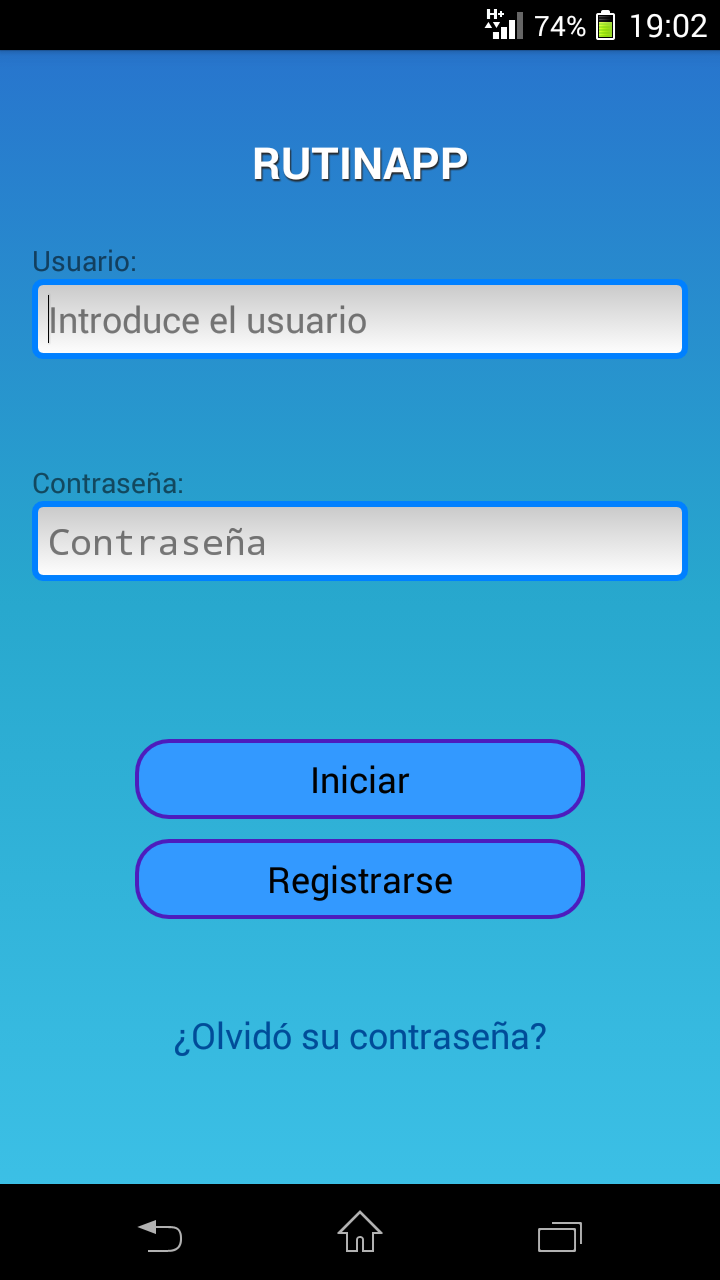
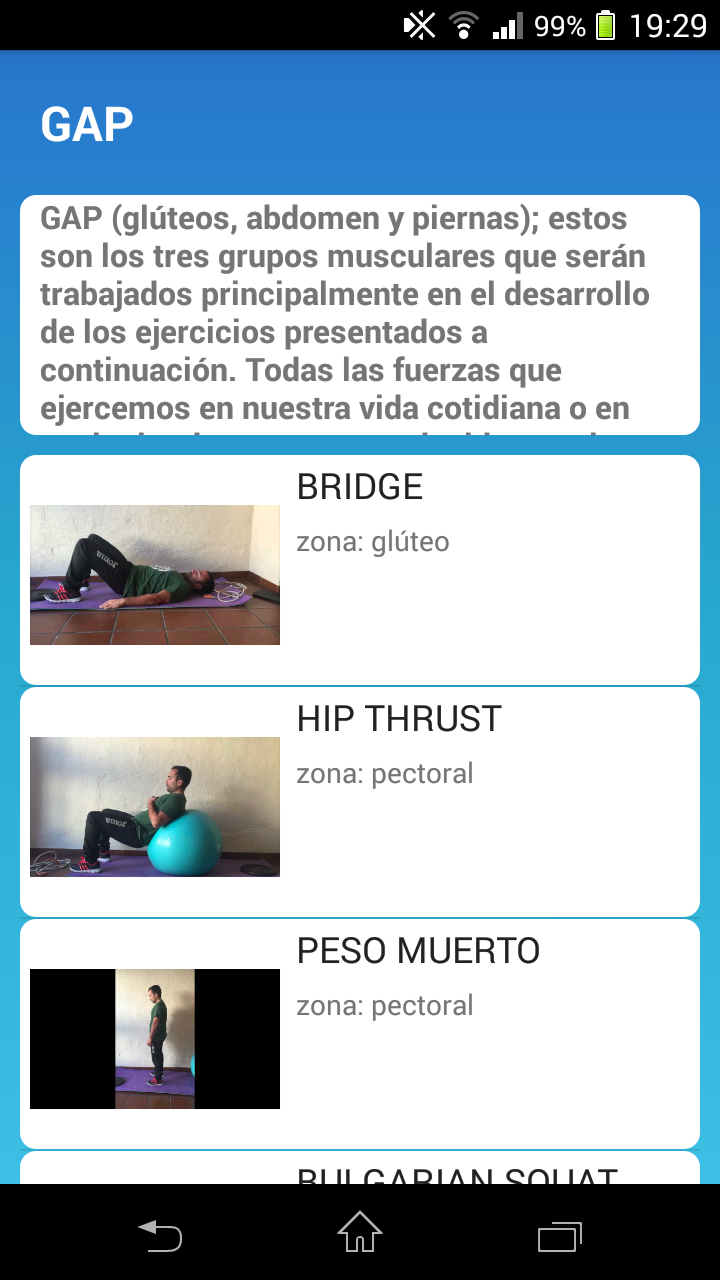
**Aplicación Android de rutinas de entrenamiento adaptadas al usuario usando SQLite y JSON**

Este es un proyecto desarrollado en la Escuela Técnica Superior de Ingeniería de Sevilla por la alumna Mirian Franco Maireles en el año 2015.

El objetivo principal de este proyecto es el desarrollo de una aplicación móvil en Android que permita al usuario tener sus propias rutinas de entrenamiento en el lugar y el momento que lo desee, con la mayor calidad y exactitud posible, y con la ausencia de un profesional especializado, tales como un entrenador personal o fisioterapeuta.

Se tienen en cuenta además especificaciones tales como la forma física del usuario, además de una descripción detallada de las rutinas y los ejercicios que agrupan, destacando como principal característica la disponibilidad de videos explicativos para los ejercicios de dichas rutinas.

El funcionamiento de dicha aplicación está totalmente desvinculado a la existencia de una red de datos, lo cual no permite la gestión de los recursos que maneja dicha aplicación a través de internet mediante alguna plataforma o servicio Web, dejándose esta como línea de desarrollo.

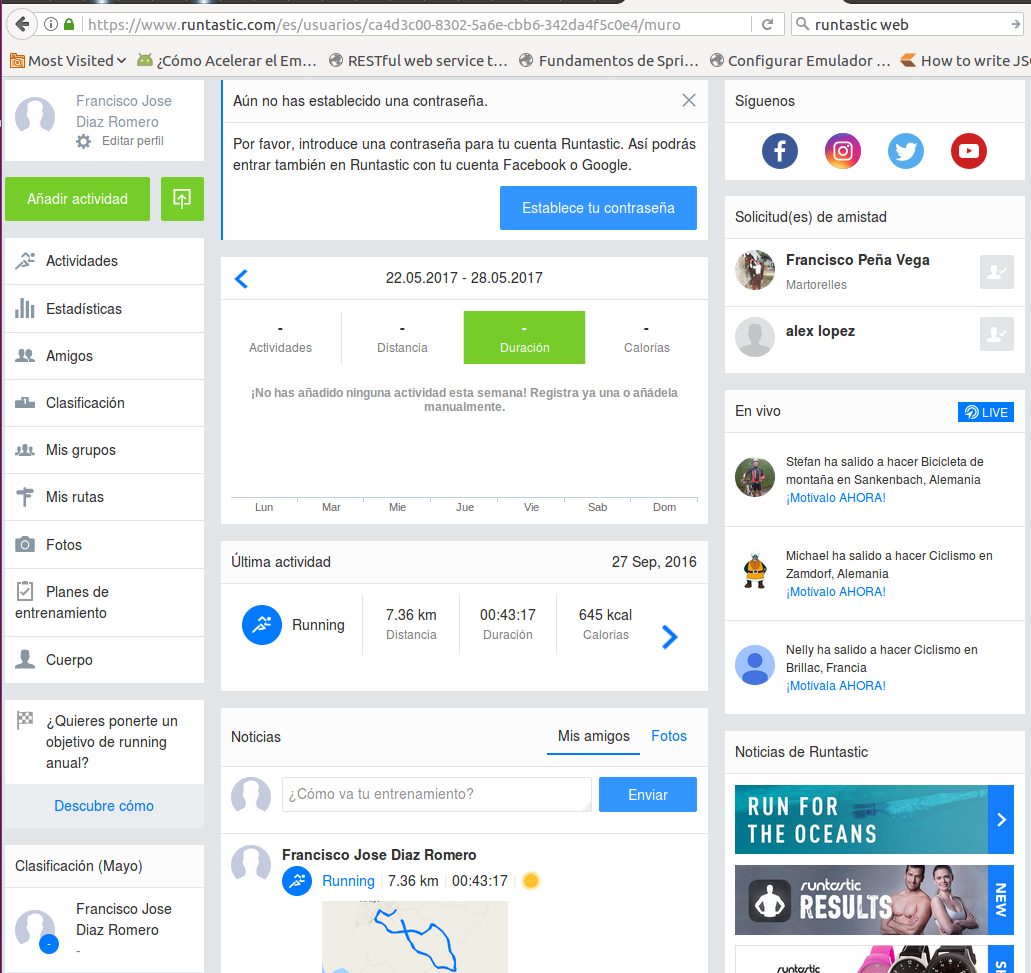
**Figura 1.2:** Aplicación Móvil Rutina App

**Otras Referencias**

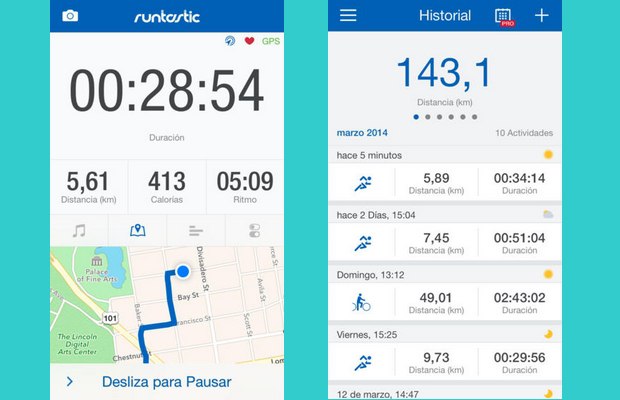
Son muchas las aplicaciones móviles que disponen además de una a plataforma web para su gestión remota o para su gestión y almacenamiento de datos en la Cloud, permitiendo tener la información distribuida y disponible siempre, además de poder compartirla con más usuarios de la plataforma si se desea. Existen aplicaciones móviles orientadas al deporte que disponen además de plataformas web, las cuales algunas de estas son:

**Runtastic**

Es una plataforma dedicada al seguimiento de la actividad física, la cual permite multitud de funcionalidades, como seguimiento de la actividad física, estadísticas, almacenamiento de rutas para actividades como el running o ciclismo, red social y sincronización con Facebook, etc. A día de hoy es una de las más completas existentes en el mercado, y cuenta tanto con aplicación web como aplicación móvil.



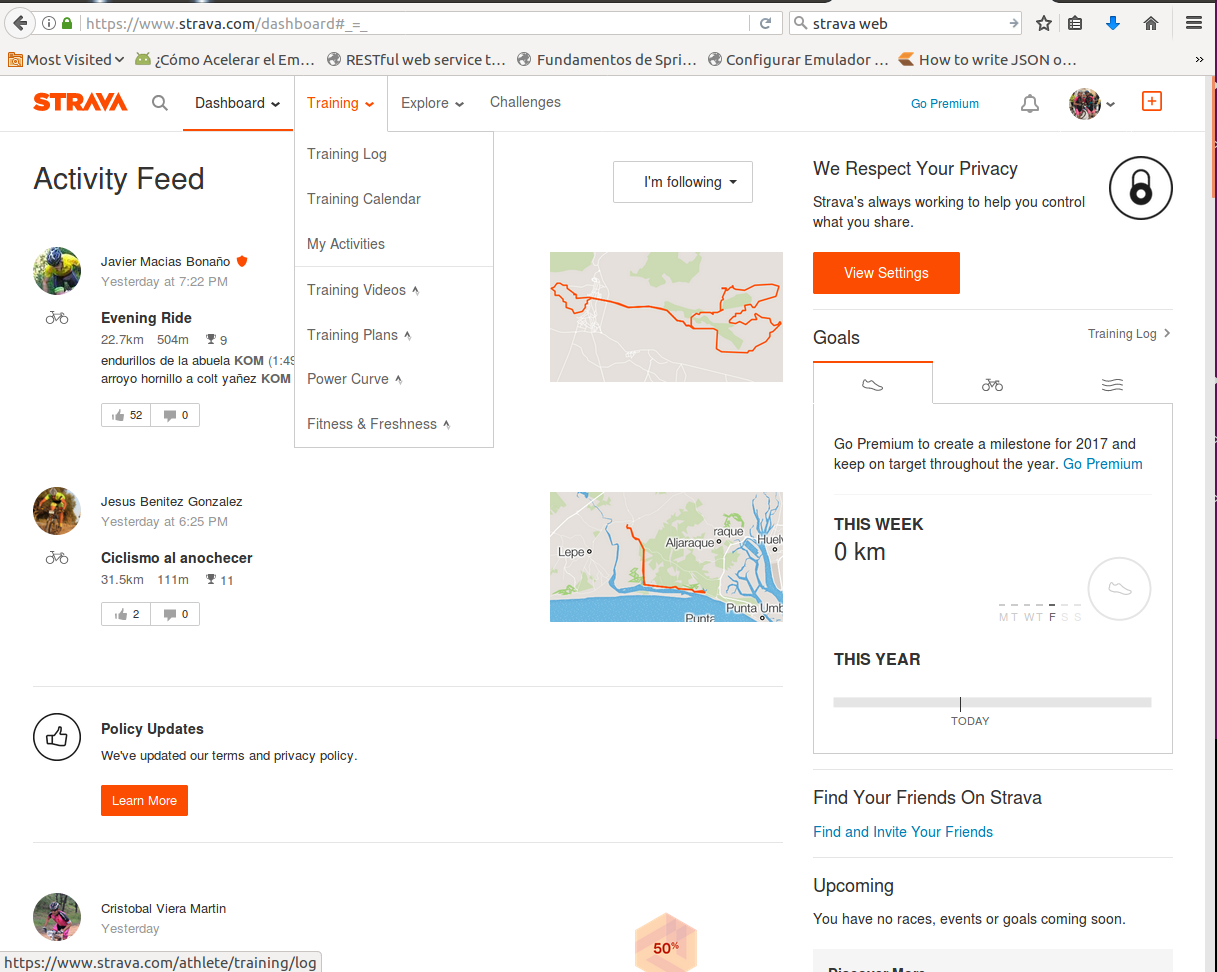
**Figura** ¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑1.1:** Aplicación Web de Runtastic.



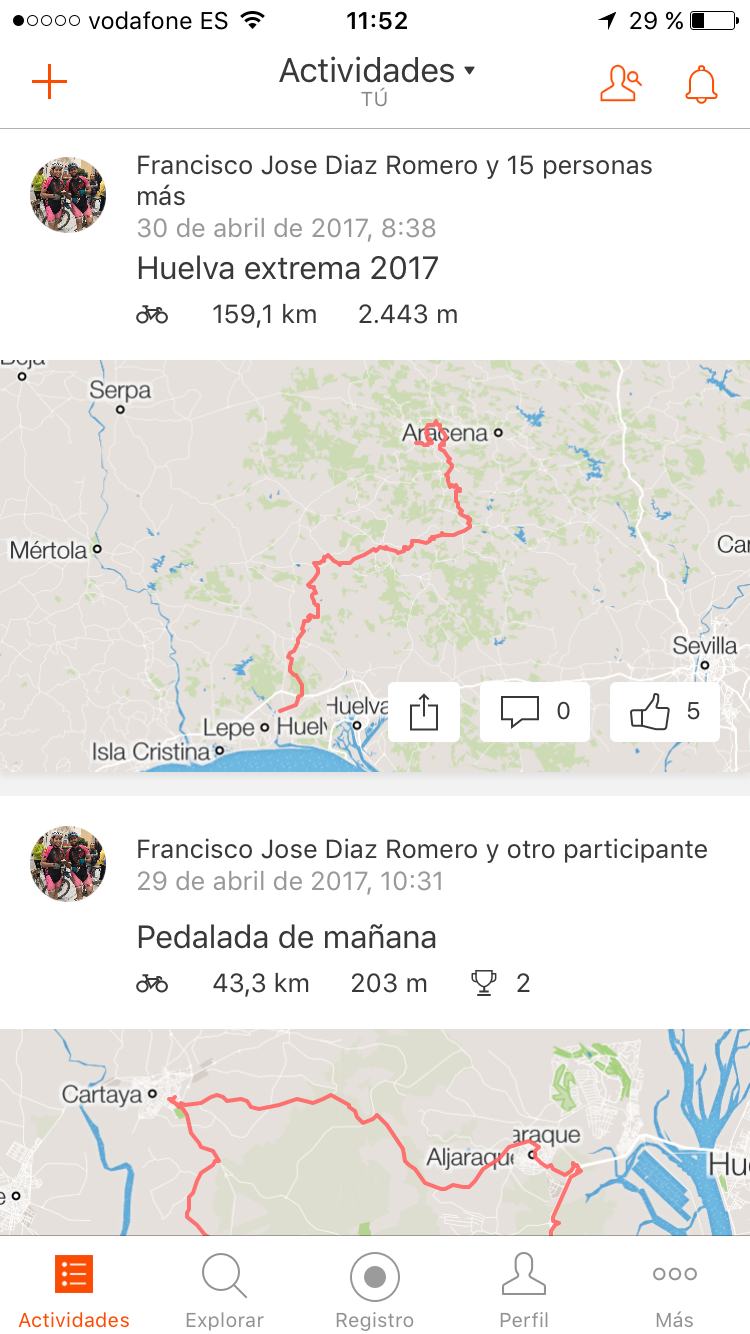
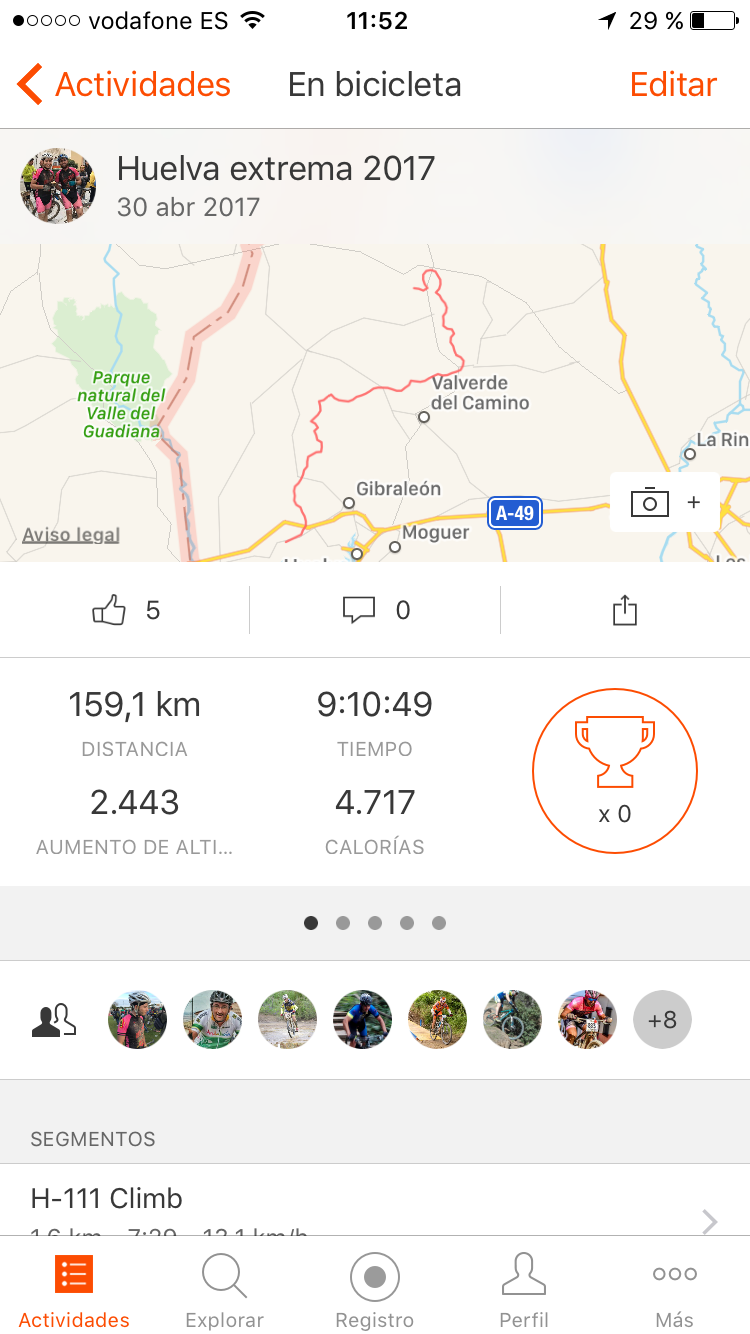
**Figura 1.**¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑2:** Aplicación Móvil de Runtastic.

**Strava**

Esta se trata de otra plataforma, la cual está dedicada solamente a dos actividades de fondo como el running y el ciclismo, mientras que la anterior permite su uso para muchos otros deportes. Esta aplicación destaca porque en el trazado de rutas en el mapa, esta reconoce tramos en el mapa transitados por más usuarios, y permite realizar rankings y estadísticas de tiempos.



**Figura 1.**¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑3:** Aplicación Web de Strava.

**Figura 1.**¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑4:** Aplicación Móvil de Strava

## 1.5 Descripción de la Solución

En esta sección abordaremos las características principales, funcionalidades y arquitectura que seguiremos en el desarrollo del proyecto.

### 1.5.1 Objetivos específicos

Como objetivo principal se busca facilitar al usuario la creación y gestión de un auto-entrenamiento a través de una plataforma web, y su posterior reproducción y realización desde cualquier lugar y en cualquier momento a través de la aplicación móvil que ya se proporciona.

A través de la aplicación Web, el usuario podrá descargar tanto sus rutinas como las rutinas que el resto de usuarios hayan decidido publicar, de forma que estas se podrán reproducir en la aplicación móvil Rutina App.

### 1.5.2 Funcionalidades

Al iniciar el navegador del usuario y acceder a la aplicación Web, en la página principal se proporcionan varias alternativas:

* Autenticación de un usuario (Usuario, Contraseña) que previamente se ha registrado.
* Acceso a la vista Web donde se permitirá el registro de un nuevo usuario.
* Acceso a la gestión del olvido de la contraseña (No implementado pero es una posibilidad).

Una vez iniciada la sesión con un usuario, se accederá a la vista principal de usuario, en la que se dispondrá de un menú con diversas funcionalidades:

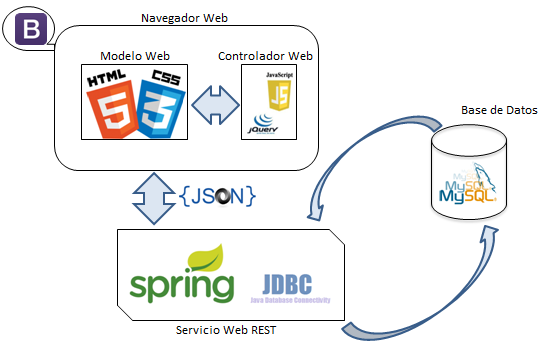
* Acceso a la modificación de los datos de usuario.
* Acceso a la creación, modificación y listado de rutinas de entrenamiento. Además, se permitirá la asociación de ejercicios ya existentes a las rutinas propias del usuario. Por otro lado, se permitirá la descarga de las rutinas existentes en formato “.Zip” para su posterior interpretación en la aplicación Móvil Rutina App.
* Acceso a la creación, modificación y listado de ejercicios. Además, se permitirá añadir videos en formato “.MP4” a los ejercicios permitiendo la subida desde la máquina local del usuario. Además, se permitirá la visualización de los videos disponibles a través de la propia plataforma web.
* Acceso a soporte de ayuda en línea, en la que como opciones se dan la baja del usuario en el sistema y una vía de contacto a través de correo electrónico con los responsables del mantenimiento y soporte de la Web.
* Cabe destacar, que los usuarios podrán hacer uso de las rutinas y ejercicios que haya creado, además de darse la opción de poder listar y descargar las rutinas y ejercicios de otros usuarios que hayan decidido hacer públicos.
* Opción para el cierre de sesión por parte el usuario.

### 1.5.3 Arquitectura

Como la mayoría de servicios Web REST, para el diseño y desarrollo de nuestro servicio hemos seguido una arquitectura basada en el modelo vista controlador (MVC), el cual separa los datos de la aplicación de la lógica de negocio que implementa y de la interfaz de usuario para acceder al servicio.

Esto permite el desarrollo por separado de cada parte, incrementando entre otras, su reutilización y flexibilidad.

Aplicando este patrón de diseño a nuestra solución, podemos visualizar el diseño de nuestro servicio en la siguiente ilustración, donde a continuación se detallará cada parte con las tecnologías y modelos de datos utilizados para ello.



**Figura 1.**¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑5:** Arquitectura de la aplicación Web para RutinaApp.

El usuario interactúa con la aplicación Web a través del Navegador, el cual obtiene datos del servidor mediante el servicio Web y los muestra en la vista, que se genera modificando el modelo Web. Este modelo se implementa con tecnologías de desarrollo y diseño Web (HTML5 y CSS3), con los controladores correspondientes para obtener y enviar datos con servicio Web REST compuestos por ficheros JavaScript. Se hará por tanto uso en su conjunto de BootStrap, el cual es un framework que proporciona este conjunto de tecnologías.

Se implementará un servicio Web tipo REST, el cual se desarrolla mediante Java Spring framework. La función principal de este servicio es la de gestionar la autenticación de usuarios, seguridad de los datos y la realización de operaciones CRUD (Create, Read, Update, Delete), sobre la información de usuarios, rutinas, ejercicios y videos. Como patrón de modelado para intercambiar datos entre el servicio Web y el controlador del cliente se hará uso de JSON, ya que su estructura es estándar y permite una buena integración entre ambas partes.

Nuestro servicio implementara la persistencia de los datos mediante una base de datos relacional. Esta se ha diseñado siguiendo un modelo Entidad-Relación, usando como gestor de BBDD MySQL. Cabe destacar que, el servicio Web REST accederá a los datos mediante la implantación del patrón DAO (Data Access Model), que detallaremos más adelante.

## Estructura de la memoria

El siguiente esquema resume de forma genérica los puntos tratados en esta memoria para que el lector pueda tener una idea orientativa de la organización de este documento:

1. **Introducción:** En este punto se abordan temas genéricos como los objetivos, ámbito y contexto, antecedentes o funcionalidades, de forma que pueda adquirirse una breve aproximación a los detalles del proyecto.
2. **Tecnologías Utilizadas:** A lo largo de esta sección se entrara en detalle sobre todas las tecnologías utilizadas e implementadas para la realización de este proyecto, con un breve razonamiento de porque se usan.
3. **Herramientas Utilizadas:** Consta de un resumen de las herramientas que se han usado para implementar y llevar a cabo las tecnologías usadas, y funcionalidades de nuestra aplicación Web.
4. **Arquitectura y Análisis:** Se realiza un análisis de diseño de la aplicación Web, mediante el uso de diagramas UML de diversos tipos.
5. **Interfaz de usuario y Funcionalidad:** Consta de un detalle sobre la funcionalidad visual de la aplicación, de forma que el usuario pueda entender como interactuar con el servicio.
6. **Conclusión:** Se incluye una conclusión final tras la realización del proyecto y una valoración del mismo a nivel general y personal.
7. **Líneas de Mejora:** Se nombraran en esta sección algunas tecnologías para mejorar este servicio y un razonamiento de las mismas. Además, se abarcan mejoras en cuanto a la propia funcionalidad de la aplicación que podrían ser implementadas y las ventajas de las mismas.

Además, se realizara un anexo, en el que se trataran puntos adicionales sobre el proyecto:

* **Anexo A:** Despliegue de servicio Web en Cualquier servidor Tomcat. (Raspberry).
* **Anexo B:** Presupuesto y Diagrama de Gantt.

# Tecnologías utilizadas

La tecnología no es en sí el fin sino el medio entre la sociedad del conocimiento y el desarrollo mundial. Anónimo

E

n este punto, mostraremos en detalle el conjunto de tecnologías usadas para llevar a cabo la realización de la aplicación Web. Realizaremos una explicación de cada tecnología y un razonamiento de la finalidad con la que se ha usado. Sera en la próxima sección donde se entrada en detalle sobre la forma en la que se han implementado dichas tecnologías en nuestra solución.

## Servicio Web REST

### Arquitectura REST

La arquitectura REST o Transferencia de estado representacional es un estilo de arquitectura introducido y definido en el año 2000 por Roy Fielding en su tesis doctoral. Fielding es uno de los principales autores del protocolo de transferencia de hipertexto (HTTP).

Es por tanto, que REST usa dicho protocolo de transferencia, lo que le da gran facilidad para poder ser usada prácticamente por cualquier lenguaje de programación y que sea fácil de testear. Además es un requisito de un servicio REST que el cliente y el servidor sean independientes entre sí.

Esta arquitectura hace énfasis en que las interacciones entre los clientes y los servicios se mejoran al tener un número limitado de operaciones (verbos). La flexibilidad se obtiene asignando recursos a sus propios identificadores de recursos universales únicos (URI). Debido a que cada verbo tiene un significado específico (GET, POST, PUT y DELETE), REST evita la ambigüedad:

* GET: Obtener un recurso.
* POST: Crear un recurso en el servidor.
* PUT: Cambiar el estado de un recurso o actualizarlo.
* DETELE: Eliminar un recurso.



Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑6

REST se ha convertido en una de las tecnologías más importantes para las aplicaciones Web. En el futuro es probable que su importancia siga creciendo rápidamente a medida que las empresas buscan proporcionar interfaces abiertas y bien definidas para los servicios de aplicaciones e infraestructura.

### Uso en nuestro proyecto (REST) y ventajas respecto a otras arquitecturas

En nuestro proyecto, hemos hecho uso de esta arquitectura como modelo de diseño para el servicio Web a implantar, ya que hoy día es el más usado por los desarrolladores, lo cual se dispone de gran cantidad de documentación en internet para aprender a usarlo

El uso de REST es mayormente preferida que con el estilo SOAP (Simple Object Access Protocol) que en comparación es más pesado, porque REST no aprovecha tanto ancho de banda, lo que hace que sea un mejor ajuste para su uso a través de Internet. Por tanto, REST nos proporciona soluciones más simples, sacrificando a su vez la potencia de dar soluciones para todo, como es el caso de SOAP.

## Formato para el intercambio de datos (JSON)

### Introducción

Cuando intercambiamos datos entre el navegador Web del cliente y el servidor necesitamos gestionar de forma sencilla y eficaz mucha información. JSON es un estándar para el intercambio de datos siguiendo una sintaxis específica y definida de forma que podemos intercambiar información entre diferentes aplicaciones independientemente de la plataforma o lenguaje en el que hayan sido desarrolladas. JSON nació como una alternativa a XML, debido a su gran facilidad de uso con JavaScript.

Se destaca principalmente porque este formato puede ser leído por cualquier lenguaje de programación, lo cual nos proporcionar mayor integridad y flexibilidad.

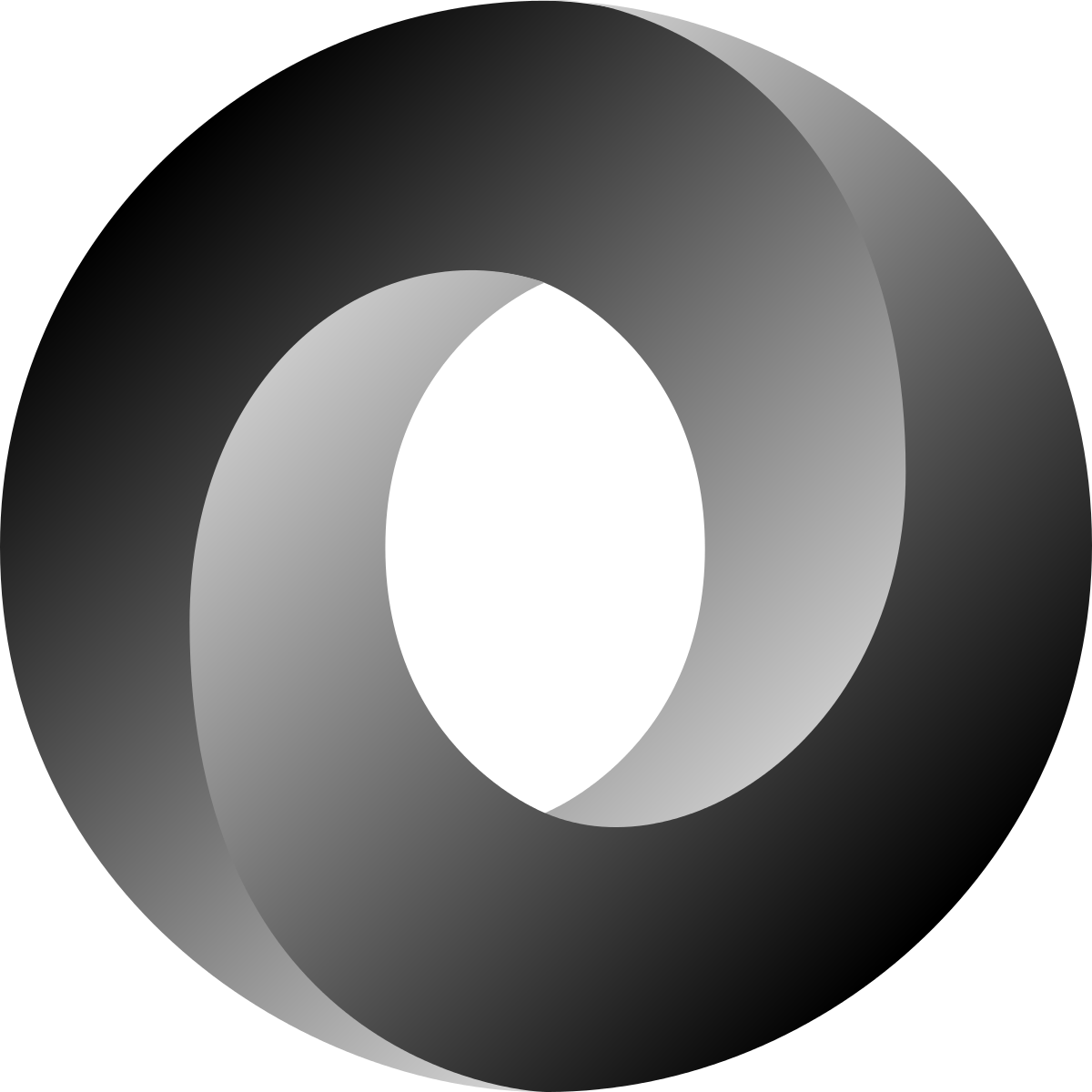


Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑7

### Otros formatos (XML)

XML es un lenguaje de marcado que define un conjunto de reglas para codificar información de forma que sea legible por un ser humano o por un ordenador. Gracias a la flexibilidad que proporciona, carece de las limitaciones que tiene HTML y puede ser usada para representar cualquier estructura de datos. Como ventaja principal de este lenguaje, cabe destacar que proporciona soporte Unicode, lo cual permite escribir la información en cualquier idioma del mundo. También cabe mencionar que su complejidad hace que JSON vaya ganando terreno respecto a XML.



Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑8

### 2.2.3 Uso en el proyecto

En nuestro proyecto, principalmente se ha hecho uso de JSON como formato para el intercambio de datos entre el cliente y el servicio web. Esto se debe principalmente a que la generación e interpretación de estructuras JSON a partir de objetos JavaScript es sumamente sencilla en el lado del cliente.

Por otro lado, cabe destacar que Spring framework también soporta XML para la representación de los datos, aunque solo hemos hecho uso de este para las directivas y ficheros de configuración tanto del servicio Web Spring REST como para el despliegue del mismo en el servidor de Tomcat.

## Tecnologías en el Servidor: Java Spring Framework

### Introducción: Spring

Este es un framework que se usa para el desarrollo de Servicios y aplicaciones y como contenedor de inversión de control de código abierto para la plataforma Java.

La primera versión de este framework fue escrita por Rod Johnson en Octubre de 2002, el cual fue lanzado inicialmente bajo licencia de Apache 2.0 en Junio de 2003. En los años posteriores, se han ido lanzando más versiones, hasta llegar a la versión 4.0 la cual se encuentra en desarrollo desde enero de 2013.

Spring fue escrito para la plataforma J2EE de Java, plataforma orientada al desarrollo de aplicaciones web y ha ido evolucionando rápidamente hasta el día de hoy, donde podemos encontrar diferentes ramas de desarrollo de la mano de SpringSource y todo su equipo de desarrolladores.



Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑9

Spring Framework nos provee de un conjunto de conceptos, técnicas y una metodología de programación para el desarrollo de aplicaciones informáticas de alta calidad, pudiendo ser usada en cualquier aplicación desarrollada en Java. Se trata pues de un framework que impulsa una metodología de trabajo ágil y eficiente, dando como resultado la creación de software de elevada calidad.

Además, tiene  amplia compatibilidad para la integración con otros frameworks y librerías de uso común para la creación de aplicaciones web, desde Composite Views como Velocity o Tiles, APIs en capa de persistencia como Hibernate o JDO, y otros muchos recursos como JavaMail, Quartz, etc…

En la actualidad, es la referencia mundial como framework de programación orientado al desarrollo Web, y su éxito se basa en la constante labor de investigación e innovación que lleva a cabo su equipo de desarrollo.

En el desarrollo de nuestro proyecto hemos hecho uso de este framework para la construcción de nuestro servicio Web REST, debido a la agilidad y diversidad de herramientas que provee, lo cual hace que su implementación y uso sea más sencillo de llevar a cabo. De todos los módulos que posee este framework, hemos hecho uso de los siguientes: Spring Web Services, Spring Boot, Spring Security Spring Data.

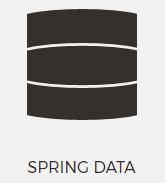
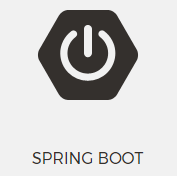


Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑10

### Spring Boot

Spring Boot es un módulo de Springque nos permite crear diferentes tipos de aplicaciones autónomas de una manera rápida y sencilla, donde simplemente hay que ejecutarlas, simplificando la creación de dependencias y el despliegue en un servidor de aplicaciones.

Principalmente se trata de un mecanismo centrado en una clase principal, la cual tiene una configuración por defecto y que podemos personalizar. Este es compatible con Maven y Gradle, y su formato por defecto para el despliegue es “.JAR”, aunque también soporta “.WAR”.

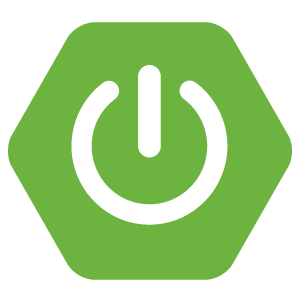


Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑11

Haremos uso de este módulo para la creación y despliegue de nuestra aplicación Web, configurada mediante una clase principal que se encargará de configurar el resto de parámetros del despliegue de la aplicación de forma automática.

Con esto se consigue que la aplicación completa quede contenida en un fichero “.JAR” o “.WAR” y solo ejecutando dicho fichero como un fichero normal de clases Java, conseguimos que la aplicación se lance de forma automática, incluyéndose un servidor web Tomcat embebido.

### Spring Security

**Spring Security** es un módulo que permitirá gestionar todo lo relativo a la seguridad de nuestra aplicación web, desde el protocolo de seguridad, hasta los roles que necesitan los usuarios para acceder a los diferentes recursos de la aplicación. Desde el punto de vista de la autenticación, nos permite comprobar que un usuario exista y permitir el acceso a los recursos de la aplicación, dando autorización a unos recursos u otros según sus credenciales y sus privilegios.

Este módulo consta de una serie de elementos, como clases y parámetros en Java, los cuales hay que personalizar para que se adapte de la mejor manera posible al comportamiento y finalidad que queremos llegar a conseguir en nuestra aplicación Web.



Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑12

### Spring Data

Este módulo tiene como propósito principal el de unificar y facilitar el acceso a distintos tipos de tecnologías de persistencia, tanto a bases de datos relacionales como no relacionales.

Por defecto, Spring Framework ya proporcionaba soporte para el acceso a bases de datos como JDBC, Hibernate o JPA, simplificando la implementación de la capa de acceso a datos, unificando la configuración y creando una jerarquía de excepciones común para todas ellas.

Spring Data permite cubrir el soporte necesario para todas las tecnologías de persistencia y además, integra las tecnologías de acceso tradicionales, simplificando el trabajo a la hora de crear las implementaciones concretas.

Con cada tipo de tecnología de persistencia, los DAO (Data Object Access) ofrecen funcionalidades típicas de un CRUD (Create, Read, Update y Delete) para objetos de dominio propios, métodos de búsqueda, ordenación y paginación. Además, proporciona interfaces genéricas para estos aspectos e implementaciones específicas para cada tipo de tecnología de persistencia.



Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑13

#### Patrón DAO (Data Object Access)

Este patrón de diseño software pertenece al catálogo de Core J2EE Patterns de Java, por lo que no es un patrón exclusivo de Spring, aunque es el utilizado para realizar aplicaciones con este framework que hagan uso de la persistencia.

El patrón DAO muestra una forma de envolver el conocimiento sobre el acceso a datos, mediante el uso de objetos de acceso a datos para conseguir abstraer y encapsular el acceso a dichos datos.

Un objeto DAO permite obtener y guardar datos, manejando la conexión con el sistema de persistencia que se haya implementado. Todo ello lo realiza mediante operaciones atómicas (Creación, actualización, borrado, obtención de registros) en la base de datos, por lo que nunca son necesarias las transacciones. Normalmente, se crea un objeto DAO por cada Objeto que tengamos en nuestra aplicación.

Esto nos permite que nuestra aplicación sea lo más transparente posible al sistema de almacenamiento usado, permitiendo la migración entre diferentes sistemas de diferentes fabricantes, diversos tipos de almacenamiento y diferentes almacenamientos de datos.

En nuestro proyecto, hemos hecho uso de este patrón para generar objetos DAO por cada objeto disponible en nuestra aplicación (Usuarios, Rutinas, Ejercicios y Videos), de forma que se usarán para realizar operaciones sobre los datos almacenados en la base de datos.

Cabe destacar que el uso de este patrón hace posible que si en un futuro deseáramos migrar nuestra base de datos a otro tipo, las modificaciones que serían necesarias en nuestra implementación no afectarían a la implementación de este modelo, sino que simplemente sería necesario ajustar la sintaxis de las consultas que realizamos para la implementación de las operaciones atómicas, hacia la sintaxis del nuevo modelo a implantar.

#### JDBC (Java Database Connectivity)

JDBC es una API que describe o define una librería estándar para el acceso a Fuentes de datos desde el lenguaje de programación Java, independientemente del Sistema operativo donde se ejecute o de la base de datos a la que se acceda, principalmente orientado a bases de datos relacionales.

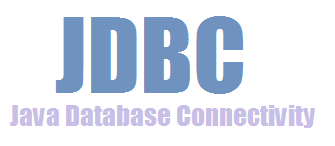


Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑14

Esta API consiste en un conjunto de interfaces y clases escritas en Java. Con estas intefaces y clases estándar, los programadores pueden escribir aplicaciones que conecten con la base de datos, enviar consultas escritas en el lenguaje de consulta estructurada “SQL” y posteriormente procesar los resultados obtenidos.

Nosotros usamos esta para para acceder a nuestro sistema de gestión de bases de datos (MySQL), y así permitir que desde nuestra aplicación Java podamos realizar operaciones sobre dicha base de datos, consiguiendo la persistencia necesaria y la disponibilidad de los datos que gestiona la aplicación.

### Maven y Gradle

### Spring Beans

### Inyección de dependencias (IoC)

### Inversion de control

## Persistencia: Base de datos

### Introducción: Modelo Relacional

El modelo relacional es una alternativa para la organización y representación de la información que se pretende almacenar en una base de datos. Se trata de un modelo teórico matemático que, además de proporcionarnos los elementos básicos de modelado (las relaciones), incluye un conjunto de operadores (definidos en forma de un álgebra relacional) para su manipulación, sin ambigüedad posible.

Este modelo, hace relativamente sencilla su representación y gestión por medio de herramientas informáticas, siendo elegido como referencia para la gran mayoría de los sistemas de Gestión de Bases de Datos disponibles en el mercado. Además, también es seleccionado como referencia para la elaboración del esquema lógico de una base de datos en su diseño, junto con herramientas como el álgebra relacional.

### Gestor MySQL y uso en nuestro proyecto

Es un sistema de gestión de bases de datos relacional desarrollado por Oracle Corporation, considerado como el más popular del mundo, además de ser desarrollado y distribuido libremente (Open Source).

MySQL es multiplataforma, lo cual da soporte a una amplia lista de sistemas operativos y trae soporte para aproximadamente diez motores de almacenamiento, en los cuales se encuentran InnoDB que es el motor predeterminado, el cual soporta transacciones y bloqueo de registros. Esto hace de este gestor que posea un alto rendimiento comparado con sistemas similares.



Ilustración **¡Error! Utilice la pestaña Inicio para aplicar 0 al texto que desea que aparezca aquí.**‑15

En este proyecto se ha hecho uso de MySQL para implementar la persistencia de los datos de la aplicación, es decir, para gestionar el almacenamiento de usuarios, rutinas, ejercicios y videos de nuestro servicio Web.

Principalmente hemos hecho uso de esta tecnología debido a la gran cantidad de información disponible en internet para el aprendizaje de su uso, ya que es el más usado en el mundo de la informática. Esta tecnología también provee de una interfaz de conexión a una base de datos MySQL compatible con Spring, llamada JDBC, que detallaremos más adelante.

Por último, este sistema de gestión de bases de datos provee de varias herramientas de gestión y de diseño de bases de datos de un interactiva y gráfica, como son XAMPP y MySQLWorkbench. Estas herramientas nos han facilitado el diseño y montaje de la base de datos, además de su gestión.

## Tecnologías para la interfaz web de usuario

### Patron Modelo-Vista-Controlado (MVC)

### Bootstrap (HTML5 + CSS3 + JavaScript)

#### HTML5

#### CSS3

#### JavaScript

### Controlador de cliente: JQuery/Ajax

#### JQuery Validation Plugin

a cubierta es la tapa del proyecto, mientras que la portada es la primera hoja que aparece al abrirlo. En Junta de Escuela de 25 de abril de 2014 se aprobó la obligatoriedad de utilizar la cubierta y portada que se incluyen en este ejemplo de formato y siguiendo las siguientes instrucciones. Debe modificar, en su caso y para la cubierta

* la titulación,
* el tipo de proyecto, atendiendo a si es fin de carrera, grado o máster.
* el título del proyecto,
* el autor,
* el tutor o tutores,
* el departamento,
* y la fecha (año).

Para la portada[[1]](#footnote-1), además de los anteriores, deberá cambiar el cargo del tutor. Por otro lado si el tutor no es docente de la ETSI entonces tendrá que añadir la figura de tutor ponente, que es un profesor de la ETSI encargado de realizar la gestión de la defensa.

El proyecto se escribirá en A4. En la cubierta, los diferentes campos se localizarán siguiendo el ejemplo de la cubierta en este documento. Tendrán los tamaños de letras y la posición orientativa siguientes, esta última dada en coorodenadas en cm tomando como referencia la esquina inferior izquierda,

* la titulación 21 pt, (4.2,27)
* el tipo de proyecto 21 pt, (4.2,25.9)
* el título del proyecto 21 pt, (4.2,16.6), podrá subirse si el título excede dos líneas
* el autor y tutor/es 15 pt, (4.2,13)
* el departamento, nombre de la ETSI y de la US, 14 pt y negrita, (centrado, 7.8). Si el nombre del departamento no cupiese en una línea, se utilizaría la siguiente, desplazando el texto inferior convenientemente.
* para el texto “Sevilla, año” 11 pt, (centrado, 5.5)

La cubierta deberá incluir la imagen de fondo que se incluye en la cubierta de este texto, con las dos bandas vertical y horizontal en el color de la fachada del edificio plaza américa y la pequeña imagen de un detalle del edificio en la zona de cruce de las bandas. Incluirá el logotipo de la ETSI a la derecha del nombre de departamento. A pie de cubierta aparecerá el logo de la Universidad de Sevilla. El logo del departamento es opcional. Si no se incluyese, el de la Universidad de Sevilla se centraría en la hoja.

# Estructura del Trabajo

Texto elaborado por Marísa Balsa, de la Biblioteca de la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla

S

E recomienda consultar la norma UNE 50136:1997: Documentación, presentación de tesis y documentos similares. Por otro lado, existen recursos en línea que ayudan a organizar todo el proceso de elaboración del Trabajo. El **personal de la Biblioteca** imparte formación y asesoramiento sobre su uso.

Para establecer la estructura y orden de los datos y partes del Trabajo Fin de Grado nos basamos en la ***Normativa de los Trabajos Fin de Grado*** de la Escuela Técnica Superior de Ingeniería de Sevilla (Aprobada en la sesión de la Junta de Escuela de 12 de julio de 2013, modificada en Junta de Escuela de 05 de febrero de 2014) y en la norma *UNE 50136:1997: Documentación, presentación de tesis y documentos similares.*

## Preliminares

### Cubierta

Deberá utilizarse **la plantilla** aprobada por la Junta de Escuela del 25/4/2014 que se describe en este texto y cuyo modelo puede tomarse de este documento. Debe contener todos los datos que se especifican y en el orden y modelo que se establece: nombre completo del autor, título del Trabajo, nombre completo del tutor, nombre del Departamento, fecha, Grado al que se opta, etc.

### Portada

Igulamente deberá utilizarse **la plantilla**… y contener todos los datos que figuran en ésta, en el orden especificado: el nombre de la Escuela, la titulación y, en su caso, la intensificación, el título del TFG, los nombres del autor, del tutor(es) y, en su caso, del ponente, el Área de Conocimiento, el Departamento y el año de ejecución del proyecto.

### Resumen

Texto que muestra de forma abreviada y precisa el contenido del trabajo. **Se podrá también incorporar una versión en inglés que se colocará a continuación precedida del término Abstract.**

### Prefacio o Introducción

Debe incluir una breve explicación de las razones que han llevado a la realización del Trabajo, el propósito y los objetivos que se pretenden, el ámbito, alcance y límites de la investigación, así como la metodología empleada y, si se considera oportuno, un avance de las conclusiones alcanzadas.

### Índice y/o Índice general

Tabla de contenidos donde se reflejan todas las partes del Trabajo y sus anexos, si los hubiera. Deben aparecer los títulos, en su orden y con indicación de la página en la que se pueden encontrar.

Si el Trabajo consta de varios volúmenes, cada uno deberá llevar su propio índice, pero se debe incluir también un índice general.

### Lista de ilustraciones y tablas

Si el Trabajo incluye ilustraciones y tablas se puede añadir un listado que incluya el número identificativo que figura dentro del texto, la leyenda y el número de la página en la que se encuentra.

También es conveniente mencionar los datos sobre las fuentes de donde se han obtenido dichas ilustraciones si no se han incluido en el propio texto de la memoria.

### Lista de abreviaturas y símbolos

El Trabajo debe contener las abreviaturas y símbolos internacionalmente reconocidos. Si se incorporan unidades, abreviaturas o acrónimos que puedan ser poco conocidos se deberán explicar brevemente en estas listas.

### Glosario

Los términos que requieran definición o explicación se deberán incorporar en un glosario.

## Texto principal

Para su elaboración es importante tener en cuenta su distribución en capítulos y secciones numeradas, siguiendo la plantilla elaborada por….

El texto debe comenzar con una introducción que muestre las investigaciones previas existentes sobre el tema y destacar los objetivos y métodos seguidos para llevar a cabo la investigación o análisis del tema tratado.

Para finalizar se deben escribir las conclusiones que deben estar en relación con los objetivos marcados previamente.

## Bibliografía

Debe contener las referencias bibliográficas de los documentos consultados para demostrar las bases del trabajo realizado y avalar los datos incorporados y citados en el texto.

Se elaborará de forma normalizada, para lo que se aconseja utilizar la norma UNE vigente (actualmente la “UNE ISO 690:2013. Información y documentación. Directrices para la redacción de referencias bibliográficas y de citas de recursos de información”).

Para la elaboración de esta parte del Trabajo se recomienda consultar la Web de la Biblioteca de Ingeniería que contiene recursos, guías y ayudas para la elaboración de las referencias bibliográficas.

## Anexos

Se puede incluir de esta forma material extenso utilizado en el trabajo, importante para justificar los resultados y las conclusiones obtenidas, pero que no es esencial para la comprensión del texto principal. Pueden ser datos estadísticos, legislación, etc.

**La paginación debe ser correlativa y continuar la del texto principal. Cada uno de los anexos debe identificarse con una letra mayúscula del alfabeto, comenzando por la letra A, precedida de la palabra Anexo.**

# Uso de Estilos

Esto es una cita al principio de un capítulo.

- El autor de la cita -

L

os capítulos nuevos pueden comenzarse de una forma cómoda copiando y pegando esta página (desde el principio hasta el final) en una página en blanco. Hay cuatro “profundidades” de Títulos que representan los capítulos, las secciones, las subsecciones y los apartados. Para asegurarse de que el capítulo comienza en página impar puede introducir un salto de sección impar.

## Secciones

Texto de sección. Esto está escrito con el estilo Normal *(Inicio > Estilos)*.

### Subsección

Texto de subsección.

#### Apartado

Texto del apartado dentro de la subsección.

Una cita a otra obra se inserta desde el menu *Referencias > Insertar Cita > Agregar Nueva Fuente*. Hay que rellenar los campos que nos interesen. Existen muchos Estilos para poner la cita. Por ejemplo, con el estilo IEEE 2006 quedan así (Autor, 2012), pero se pueden probar otros hasta ver el que más nos convence (Autor O. , 2001).

Una vez tengamos insertada esa cita, ya podemos encontrarla en el menú *Referencias > Insertar Cita* sin tener que agregar una nueva fuente.

### Otra subsección

Texto de la otra subsección. Aprovecharemos para insertar una nota[[2]](#footnote-2) al pie.

## Otra sección

Cada vez que escribamos uno de estos títulos y pulsemos Enter, volverá al estilo Normal.

Gracias a los Elementos Rápidos, podemos insertar *Teoremas, Ejemplos, Demostraciones, Definiciones, Imágenes y Tablas.*

|  |
| --- |
|  |

Para insertar ecuaciones, deberemos tener instalado MathType y que se integre correctamente en Word (Aparecerá una pestaña en el menú superior). Se puede usar el editor de ecuaciones integrado en Word, pero aunque tiene alguna ventaja no se recomienda. Aquí, como puede ser que el lector no tenga MathType, se ha optado por usar el editor de ecuaciones de Word.

Dentro de la pestaña MathType, se puede insertar una ecuación inline como esta, que no modifica la separación entre líneas gracias a una propiedad del estilo Normal. Hay que tener cuidado con esto, pues ecuaciones inline con mucha altura pueden superponerse con el texto. En el raro caso de que ocurra, lo mejor es reducir el tamaño de la ecuación o poner una ecuación a parte.

Las ecuaciones a parte se ponen pulsando sobre *MathType > grupo “Insert Equations” > Right-numbered*, y se numeran automáticamente según el número del capítulo. Otra opción, la que se adopta a continuación es crear una tabla con el número generado con un título “(“ incluyendo el número de capítulo, si no lo ve en sus títulos tendrá que crearlo,

|  |  |
| --- | --- |
|  | (4–1) |

Las ecuaciones numeradas tienen su propio estilo, MTDisplayEquation, que puede editarse para cambiar alineación y otros parámetros (aunque no es necesario).

Para referenciar una ecuación, usamos *MathType > grupo Equation Numbers > Insert Reference*, y luego hacemos doble click sobre el número de la ecuación que queremos, en este caso (4–1).

Vamos a ver algunos elementos rápidos, empezando con el más sencillo de ellos, la Definición.

**Definición** (Concepto):Texto de la definición. Esto es otro de los Elementos Rápidos, aunque no tenga numeración.

Las tablas también son Elementos Rápidos. Todos los elementos rápidos se acceden desde el menú de Insertar. No obstante, también podemos copiar de estas mismas líneas el elemento que nos interese, y pegarlo donde queramos editarlo.

Tabla 4–1. Tipos de transmisión y frecuencia central

|  |  |
| --- | --- |
| Tipo de Transmisión | Frecuencia central de transmisión |
| Modem | 100-1800 Hz |
| Radio AM | 530-1600 kHz |
| Radio FM | 88-108 MHz |
| Televisión | 178-216 MHz |
| Telefonía móvil | 850 MHz-1,8 GHz |
| Redes inalámbricas | 2*,*4 GHz |
| Fibra óptica | 2·1014 Hz |

De hecho, cualquiera puede crear un Elemento Rápido a su gusto desde el propio menu de Elementos Rápidos, pulsando en *“Guardar selección …”* tras haber seleccionado la parte del documento que queremos guardar.

Otro de los Elementos Rápidos es el Ejemplo.

**Ejemplo 4–1.** Al insertar ecuaciones en un ejemplo, puede haber problemas con las barras grises superior e inferior.

Los ejemplos que dan más problemas son los que llevan ecuaciones o los que continúan en la página siguiente. En ese caso hay que hacerle un retoque final al mismo.

**Ejemplo 4–2.** Al insertar ecuaciones en un ejemplo, puede haber problemas con las barras grises superior e inferior.

|  |  |
| --- | --- |
|  | (4–2) |

En cualquier caso, estas barras se configuran una vez está acabado el ejemplo en el menú Inicio > grupo “Párrafo” > pulsando sobre el icono inferior derecho de ese grupo. Por ejemplo, si nos aparecen barras grises delimitando la ecuación recién insertada, basta con situarse en el párrafo superior e inferior a la ecuación y eliminar esas barras en el menú que hemos dicho.

Si queremos referenciar cualquier elemento numerado a lo largo del texto (como por ejemplo “0

Secciones” o “Tabla 4–1”, se hace desde el menú *Insertar > grupo “Vínculos” > Referencia cruzada*. Luego hay que buscarlo en la caterogría adecuada (*Tipo*) y especificar a qué queremos que haga referencia (*Referencia a*). Si quedan con un formato indeseado, siempre se puede marcar y aplicar el estilo Normal.

Debemos saber que en cualquier momento, si una numeración no nos aparece correctamente, puede ser debido a que Word todavía no la ha actualizado. Para forzar esta actualización, hacemos click sobre la última cifra de esa numeración y pulsamos *Actualizar Campos*.

|  |
| --- |
| **Teorema 4.1 (Nombre del teorema)**  Aunque algunas ecuaciones aparezcan cortadas contra el fondo gris, al imprimirlas en PDF salen bien. No obstante, al insertar una ecuación puede ser necesario incluir alguna línea en blanco adicional, si parece que van a solapar con el texto…  Y esta es la línea final del Teorema. |

**Demostración:** Las demostraciones siguen al teorema, tienen cierta sangria y acaban con un cuadrado negro antes de volver al estilo Normal. Las demostraciones demasiado engorrosas pueden ponerse en un apéndice al final del documento, pero eso ya es cuestión personal.

∎

Algunos conceptos son muy importantes para la Tesis y conviene añadirlos al glosario al menos la primera vez que aparecen. Para ello, los marcamos y pulsamos *Referencias > Marcar entrada.* Luego pulsaremos “Marcar” y “Cerrar”. Si Word nos ha pasado al modo “Mostrar Todo”, donde se muestran los símbolos de formato, podemos salir de él pulsando *Control + Shift + (.*

# Otro Capítulo

The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.

Claude Shannon, 1948

E

n este capítulo vamos a describir las partes de las que consta un documento tipo, cómo deben interpretarse los diferentes comandos que se han definido para su confección.

Los capítulos nuevos pueden comenzarse de una forma cómoda copiando y pegando esta página (desde el principio hasta el final) en una página en blanco. Hay cuatro “profundidades” de Títulos que representan los capítulos, las secciones, las subsecciones y los apartados. (título1 a título5)

Hay mucho de elección personal en lo que sigue y únicamente se justifica desde el gusto personal de quienes escribimos esto. No pretendemos por ello sentar precedentes, obligaciones ni restricciones a quien desee utilizar este documento. En cualquier caso, esperamos que su lectura sea provechosa para la confección y edición de libros, apuntes de clase, proyectos, etc.

## Estilos de un documento

Aquí definiremos los principales estilos que se han definido en este documento. Estos pueden accederse desde la pestaña *Inicio* y clicando en la sección *Estilos*, los verá en la parte media de la barra de herramientas. Pruebe a consultar todos los estilos que posee el documento pinchando en la flecha que apunta hacia abajo con una raya encima, debajo de los estilos que aparecen.

El dominio de los estilos es fundamental para una buena edición de un texto con Word. Se recomienda invertir un tiempo en “jugar” con las opciones en el menú Formato>Estilo… Una de las mayores ventajas del dominio de los estilos es asignar una numeración automática a cada capítulo, sección y apartados. De esta forma al introducir o eliminar estos elementos el editor renumera automáticamente todo el documento.

Los principales estilos que se han diseñado para este texto son los siguientes:

* *Autor\_cita\_celebre\_capitulo* -> Se aplica al autor de la cita con la que se suele iniciar un capítulo.
* *Cita\_celebre\_capitulo* -> El contenido de la cita con la que se inicia un capítulo.
* *Código\_cuerpo ->* El contenido (sin incluir el título) de un bloque de código.
* *Definición ->* El texto que forma parte de una definición.
* *Demostración ->* El texto que forma parte de una demostración.
* *Encabezado ->* El estilo con el que se muestran los encabezados de las distintas páginas.
* *Normal ->* El texto “estándar” del documento,
* *Otro\_titulo ->* Para aquellos títulos de primer nivel que no deben numerar, por ejemplo: Resumen, Agradecimientos, etc.
* *Otro\_titulo2 ->* Para aquellos títulos de segundo nivel que no se deben numerar, por ejemplo los títulos de los problemas.
* *Portada\_autores->* Para mostrar los nombres de los autores en la portada del documento.
* *Portada\_departamento ->* Para mostrar el nombre del departamento en la portada del documento.
* *Portada\_fecha ->* Para mostrar la fecha en la portada del documento.
* *Portada\_proyecto ->* Para mostrar el nombre del proyecto en la portada.
* *Subportada\_proyecto ->* Para mostrar los datos en la página interior que sigue a la portada.
* *Teorema ->* Para usar cuando se exprese un teorema.
* *Titulo\_previo ->* Muy similar a otro\_titulo.
* *Titulo\_codigo ->* Para los títulos en los bloques de código
* *Título 1 ->* Primer nivel de títulos (capítulos), tiene un subrayado en gris.
* *Título 2 ->* Segundo nivel de títulos (secciones).
* *Título 3 ->* Tercer nivel de títulos (subsecciones).
* *Título 4 ->* Cuarto nivel de título (apartados).
* *Título 5 ->* Usado para algunos encabezados menores.
* *Descripción ->* Se usa en los títulos de figuras, tablas, etc

## Títulos y Referencias Cruzadas

El otro elemento fundamental, además de los estilos, que el usuario debe dominar es el par Títulos/Referencia Cruzada. Si va al menú Insertar>Título verá que puede incluir un título de la lista. En el presente documento hay títulos definidos para ecuaciones del tipo (X-Y), por si no usa la numeración de Mathtype, para Figura X-Y, Tabla X-Y, Ejemplo X-Y y Teorema X-Y, donde X es el número del capítulo e Y es el número dentro del capítulo. Puede probar a crear uno, como por ejemplo Lema. Una vez que se ha introducido un título, para la leyenda de una Figura o Tabla, o para numerar una ecuación, se puede hacer referencia a él insertando Referencia Cruzada, en el mismo menú Insertar.

Este elemento es fundamental, de nuevo, para que al introducir o eliminar una nueva figura o ecuación el sistema pueda renumerar automáticamente (opción actualizar campos) todos los títulos y actualizar las referencias cruzadas. Para esto a veces es preciso seleccionar el texto y en el menú que aparece con el botón derecho del ratón pulsar Actualizar Campos. Además, esto permite insertar una lista de elementos al final o principio del documentos.

## Versiones y Sistemas Operativos

El programa Word® dispone de diversas versiones, este documento ha sido realizado usando la más reciente de ellas Office 2013, se ha comprobado que funciona correctamente, al menos, con las versiones 2010 y 2007 de la suite ofimática. Se proporciona un fichero con extensión .docx, que no es totalmente compatible con versiones anteriores. En todo caso el Centro de Cálculo y la Biblioteca de la Escuela cuentan en sus instalaciones con equipos con este software instalado y de uso público. Veamos ahora cómo trabajar con este documento en sistemas distintos de Windows.

### **Macintosh**

Para Macintosh está también disponible la suite Microsoft Office, por lo que no debe haber problemas al trabajar con este documento. Se ha probado en un Mac con la versión 2010 de Word® y funciona correctamente.

### Linux

Para Linux no existen versiones nativos de Office. Se nos ofrecen varias alternativas para poder usar este documento:

* Ejecutar un emulador para correr sobre él la suite de Office original, ejemplos son Wine y Crossover Office (específicamente diseñado para correr Office). La ventaja de esta opción es que trabajamos con el mismo sistema y el documento no sufre ninguna alteración y el consumo de recursos es moderado, la desventaja es que su configuración puede llegar a ser tediosa.
* Instalar una máquina virtual y sobre ella un sistema operativo Windows con su correspondiente copia de Office. Como desventaja más evidente comentar que no todos los equipos cuentan con potencia suficiente como para ejecutar esta opción.
* Usar una suite ofimática alternativa, como LibreOffice. El consumo de recursos será moderado y su uso es inmediato, por el contrario puede que el documento no se vea exactamente igual y que algunas operaciones deban ser realizadas de una forma distinta.

## Texto en inglés

El idioma por defecto de este documento es el Español, si usted quiere que algunas opciones como la autocorrección, el formato de moneda y alguna otras le aparezcan en Inglés o en algún otro idioma, simplemente empiece a escribir en ese idioma y el programa automáticamente lo detectará si el texto es lo suficientemente largo, es posible que le pida que instale el soporte extra para ese idioma si usted no lo tenía previamente instalado

## Elementos básicos de un libro

En este capítulo describimos los puntos que pueden incluirse con el formato propuesto. En primer lugar, la longitud de un libro, en general, justifica su separación en partes. Una posibilidad es que un libro esté dividido en Partes y esta a su vez en Capítulos. Y por último, a veces existen Apéndices que se incorporan cuando han acabado los capítulos. En nuestro caso sólo hemos considerado la posibilidad de dividir el libro en capítulos. Además, existen un conjunto de elementos como dedicatoria, prefacio, agradecimientos, cubierta, etc, que también son partes que se han tenido en cuenta.

En un nivel de descripción diferente, podríamos considerar que un libro se encuentra dividido en cubierta, página de título y trasera de la página de título, elementos antes del cuerpo del libro, tales como agradecimientos, prefacio, índices, etc, el cuerpo del libro en sí, dividido en capítulos y esto a su vez en secciones, subsecciones, subsubsecciones, subcapítulos, apéndices y, por último, la parte del libro después del cuerpo, que agruparía elementos tales como la lista de figuras del libro, la bibliografía, el índice, etc.

## Símbolos y fórmulas

Aunque Word® no es un editor científico ni la herramienta más utilizada para estos menesteres, hay que reconocer que el editor de fórmulas que trae integrado ha mejorado notablemente con las sucesivas versiones. Así mismo, actualmente se pueden insertar casi cualquier símbolo de un uso medianamente cotidiano. Para insertar un símbolo o una fórmula simple se debe ir a la pestaña *Insertar* y después pulsar en *Símbolo,* en la parte derecha de la barra de herramientas. Justo al lado nos encontramos con la opción de introducir un *Cuadro de texto*, el cual puede servirnos para resaltar ciertos tipos de contenido. En MacOS es muy común tener problemas con los símbolos, al estar duplicada la fuente “símbolo” y tener que desactivar una de ellas.

## Ecuaciones y MathType®

Como hemos comentados en el apartado previo el editor de ecuaciones integrado de Word® ha mejorado mucho versión tras versión, aún así si nuestro trabajo tiene un número importante de expresiones matemáticas no la mejor opción. Nos encontramos en el mercado con la herramienta Mathtype® que, tras instalarla, se integra perfectamente con Word®, apareciéndonos una nueva pestaña en la parte derecha donde tendremos acceso a todas las funciones de Mathtype®. Esta aplicación no es gratuita, si bien cuenta con una versión de evaluación plenamente funcional operativa durante 30 días ya que puede descargarse desde la url: [http://www.dessci.com/en/products/Mathtype®/](http://www.dessci.com/en/products/mathtype/).

Para insertar ecuaciones, deberemos tener instalado MathType y que se integre correctamente en Word (Aparecerá una pestaña en el menú superior[[3]](#footnote-3)). Se puede usar el editor de ecuaciones integrado en Word, pero aunque tiene alguna ventaja no se recomienda. Aquí, como puede ser que el lector no tenga MathType, se ha optado por usar el editor de ecuaciones de Word. Las ecuaciones s incluyen pulsando sobre *Mathtype® > grupo “Insert Equations” > Right-numbered*, y se numeran automáticamente según el número del capítulo. Las ecuaciones numeradas tienen su propio estilo, MTDisplayEquation, que puede editarse para cambiar alineación y otros parámetros (aunque no es necesario). Para referenciar una ecuación, usamos *Mathtype® > grupo Equation Numbers > Insert Reference*, y luego hacemos doble click sobre el número de la ecuación que queremos, en este caso **¡Error! No se encuentra el origen de la referencia.**.

Dentro de la pestaña MathType, se puede insertar una ecuación inline como esta, que no modifica la separación entre líneas gracias a una propiedad del estilo Normal. Hay que tener cuidado con esto, pues ecuaciones inline con mucha altura pueden superponerse con el texto. En el raro caso de que ocurra, lo mejor es reducir el tamaño de la ecuación o poner una ecuación a parte. También puede jugar con el interlineado, en el menú Formato>Párrafo.

Las ecuaciones a parte se ponen pulsando sobre *MathType > grupo “Insert Equations” > Right-numbered*, y se numeran automáticamente según el número del capítulo. Otra opción, la que se adopta a continuación es crear una tabla con el número generado con un título “(“ incluyendo el número de capítulo, si no lo ve en sus títulos tendrá que crearlo,

|  |  |
| --- | --- |
|  | (5–1) |

Las ecuaciones numeradas tienen su propio estilo, MTDisplayEquation, que puede editarse para cambiar alineación y otros parámetros (aunque no es necesario).

Para referenciar una ecuación, usamos *MathType > grupo Equation Numbers > Insert Reference*, y luego hacemos doble click sobre el número de la ecuación que queremos, en este caso (4–1).

Por otro lado, en el menú de MathType se permite definir un estilo de ecuación, con tipos de letras y tamaños de las mismas, para nuevas ecuaciones y para aplicarlo a todas las ecuaciones del texto. Esto último es muy útil a veces.

### Fuentes

La selección de las fuentes para la edición de cualquier texto no es una tarea trivial. En realidad, el diseño tipográfico es todo un arte. Un convenio bastante aceptado es utilizar fuentes con serif para el texto y sin serif para titulares y cabeceras de páginas. Sin embargo, la elección de cualquiera de estas familias de fuentes es prácticamente cuestión de gusto personal y, por qué no decirlo, de la moda del momento.

En los primeros tiempos de Word® las posibilidades de elección estaban bastante limitadas. Sin embargo, con el advenimiento de nuevos métodos y programas, es posible elegir prácticamente cualquier fuente existente para su uso, y asociarla a un estilo.

### Epígrafes o citas célebres

En muchos libros, después del título de un capítulo o antes del resumen, o en el lugar que apetezca, se coloca una frase con diversos significados. Para ello nosotros simplemente debemos escribir el texto que deseemos y después aplicarle el estilo *Cita\_celebre\_capitulo*, para indicar el autor de la cita debemos escribirlo y después aplicarle el estilo *Autor\_cita\_celebre.*

### Figuras y tablas

Una parte importante de cualquier texto son las figuras y tablas que lo acompañan. En Word® estos elementos se consideran elementos flotantes.

La inclusión de las figuras se realiza simplemente yendo a la pestaña *Insertar -> Imágenes* y seleccionando la imagen de la ubicación de nuestro equipo donde se encuentre. Si bien Word® incluye algunas herramientas para la generación de figuras y diagramas, nuestra recomendación es usar alguna herramienta externa, guardar el resultado como una imagen y después insertarla. Por ejemplo para la creación de diagramas podemos recomendar Visio (propietario), Dia (libre y gratuito) y Lucidchart (una opción online muy potente). Para la edición de imágenes cualquier programa de edición de las mismas nos servirá.

|  |
| --- |
|  |
| Figura 5‑1. Pie de figura |

Para insertar una tabla actuaremos de una forma similar en la pestaña *Insertar -> Tabla,* una vez hallamos hecho esto nos aparecerán a la derecha dos nuevas pestañas con la etiqueta “Herramientas de tabla” por encima. Estas dos nuevas pestañas son *Diseño* y *Presentación,*  donde podremos designar varias características visuales y de formato de las tablas. Nuestra elección se muestra en la tabla inferior, nuevamente solo hay que copiar y pegar para poder reusar este formato. Si quiere mantener una tabla sin que se rompa en dos hojas contínuas, tendrá que seleccionar todas las filas menos la última y en el menú Formato>Párrafo seleccionar la pestaña Líneas y Saltos de Páginas y allí pulsar la opción Conservar con el siguiente.

Tabla 5–1 Tipos de transmisión y frecuencia central

|  |  |
| --- | --- |
| Tipo de Transmisión | Frecuencia central de transmisión |
| Modem | 100-1800 Hz |
| Radio AM | 530-1600 kHz |
| Radio FM | 88-108 MHz |
| Televisión | 178-216 MHz |
| Telefonía móvil | 850 MHz-1,8 GHz |
| Redes inalámbricas | 2*,*4 GHz |
| Fibra óptica | 2·1014 Hz |

Observemos que en la parte inferior de las figuras y en la superior de las tablas (esta ha sido nuestra elección), se colocan textos explicativos sobre las mismas. Para introducir este título pulsamos con el botón derecho sobre el elemento (figura o tabla) y pulsamos insertar título, no se nos debe olvidar indicar la categoría correcta en cada caso, figura o tabla para que la autonumeración funcione correctamente.

### Hiperenlaces

Un primer paso a la hora de crear un documento es generar una versión en formato electrónico del mismo. Hemos decidido que ese formato sea pdf. En un formato pdf existe la posibilidad de crear hiperenlaces que facilitan la navegación a lo largo del mismo. Por ejemplo, el índice en un libro en formato pdf se generará, con la propuesta que hemos realizado, creando enlaces a las diversas partes del mismo. O bien, cuando nos referimos a una figura o tabla, es muy útil la existencia de esos enlaces al lugar exacto en el que se encuentra la figura o tabla. Esto lo realizaremos y veremos con más detalle en la siguiente sección. Si queremos incluir un enlace a un elemento externo (una página web, por ejemplo) deberemos ir a la pestaña *Insertar -> Hipervínculo*, nos aparecen dos campos para rellenar: en “Texto” pondremos el texto que apecerá en nuestro documento, por ejemplo Google, y en “Dirección” la url a la que apuntará, por ejemplo www.google.es, quedando el resultado final así: [Google](http://www.google.es).

### Tabla de contenido

La generación de la tabla (o tablas) de contenido de un texto suficientemente largo suele ser una tarea sumamente laboriosa. Word® facilita enormemente este trabajo. En otra sección de este capítulo explicaremos cómo y dónde se incorporará esta tabla de contenidos. En este apartado nos centramos en explicar algunos aspectos de cómo se construye la principal tabla de contenidos, que denominamos *Índice*.

Nuestra primera decisión fue establecer que en el índice deben aparecer hasta los apartados que hemos denominados subsecciones, lo que se logra mediante la secuencia que vamos a ver a continuación. También hemos propuesto que no aparezcan los habituales puntos que existen entre el texto y el número de página correspondiente de muchos índices.

Para usar el que índice incluido simplemente deberemos escribir nuestro texto usando los correspondientes estilos de títulos, a la hora de actualizarlo deberemos pulsar el botón derecho sobre el mismo y pulsar en *Actualizar Campos* y elegir *Actualizar todo.* También podemos ir al menú Insertas>Indice y Tablas y allí tabla de contenido, usando el estilo sencillo.

### Índice de figuras, tablas y otros elementos

Es habitual, y aquí lo hemos hecho también, incluir un índice de figuras y/o tablas al final del documento cuando éste posee muchos de estos elementos. También se pueden incluir índices de Código, Teoremas o cualquier otro elemento del estilo. Para crear un índice de este tipo debemos hacer lo siguiente: Ir al la pestaña *Referencias-> Insertar tabla de ilustraciones;* ahí debemos pinchar en el desplegable *etiqueta de título* y seleccionaremos el elemento para el cual queramos crear el índice. Otra posible opción es pulsar en el botón *Opciones*, marcar la pestaña estilo *estilo*  bajo Generar tabla de ilustraciones a partir de: y seleccionar el estilo del elemento del que queramos generar el índice.

### Formatos de páginas

El aspecto de un libro está básicamente definido por el formato que se ha elegido para los diferentes títulos de las partes que lo constituyen, el formato de las páginas y qué queremos que aparezca en las cabeceras y pies de páginas del mismo. En las páginas pares el nombre del capítulo en el que estamos y en las impares el nombre del documento. Estos elementos se colocan encima de una raya horizontal que se ha definido previamente, tanto en su grosor como en su longitud.

En todo caso, estos parámetros no se deberían de tocar, salvo en contadas ocasiones.

### Teoremas y otros elementos similares

En la escritura de cualquier texto científico los Teoremas, propiedades y demás elementos constituyen una parte muy significativa. Existen, de nuevo, múltiples posibilidades de tratar estos elementos, pero hemos considerado que el formato propuesto se adapta perfectamente a nuestros gustos y decisiones.

A modo de ejemplo hemos definido el Título Teorema X-Y, puede probar a crear los estilos Lemas, Propiedades y Corolarios si les hiciera falta. Centrándonos en los teoremas, el resultado sería el siguiente, donde se ha definido una etiqueta Teorema:

**Teorema 5–1** (Teorema de Pitágoras) *En un triángulo rectángulo...*

Demostración. Sea el triángulo ABC...

|  |  |
| --- | --- |
|  | (5–2) |

∎

Podemos observar que al finalizar la demostración hemos incluido el símbolo ∎.

### Ejemplos

En este apartado se muestra cómo meter un ejemplo. Esta forma propuesta es el resultado de jugar con Títulos y Estilos. Para citarlo, puede recurrir a Referencia Cruzada, siendo el resultado ***Ejemplo 5–1***.

**Ejemplo 5–1.** Al insertar ecuaciones en un ejemplo, puede haber problemas con las barras grises superior e inferior.

### Índices de palabras y glosarios

Con los paquetes index y glossaries podemos incluir índices de palabras y listas con definiciones, ya sea de acrónimos u de otro tipo. Por ejemplo, se podría usar también para definir magnitudes o la notación utilizada.

#### Índices de palabras

Algunos conceptos son muy importantes para la Tesis y conviene añadirlos al glosario al menos la primera vez que aparecen. Para ello, los marcamos y pulsamos *Referencias > Marcar entrada.* Luego pulsaremos “Marcar” y “Cerrar”. Si Word® nos ha pasado al modo “Mostrar Todo”, donde se muestran los símbolos de formato, podemos salir de él pulsando *Control + Shift + (.*

#### Glosario

Para realizar un glosario deberá realizarse a mano respetando (si así se quiere), el formato propuesto en la sección correspondiente.

Tambíen se puede generar en Word de una forma más dinámica usando tabla de autoridades. Hacemos click en la parte del documento donde se quiere añadir el glosario y, posteriormente, en el apartado *referencias* de barra superior, en la sección *Tabla de autoridades*, seleccionamos la opción *Insertar* *Tabla de autoridades,* donde podremos configurar el formato que tendrá nuestro glosario y, al aceptar, declarar dónde se situará.

Para añadir entradas de forma dinámica basta con seleccionar la entrada que queremos añadir y pulsar en *Marcar cita*, en el mismo apartado donde hemos pulsado para crear la tabla. En la ventana que aparecerá añadimos la descripción de la entrada en el cuadro *Texto seleccionado* y pulsamos *Marcar.* Se puede ir actualizando el glosario pulsando F9.

En el caso de que al general el glosario aparezca un título, pulsando click derecho sobre el mismo y eligiendo la opción *Activar o desactivar códigos de campo* veremos una línea de código similar a {TOA \h \c "1"}.   
Quitando el \h eliminaremos ese título.

Si aparecieran símbolos en los saltos de linea o se sustituyeran los espacios por puntos, en el apartado *Inicio*, sección *Párrafo* pulsamos en Button imagey volveríamos a tener el documento como antes. Este botón hace que muestra/desaparecan esas marcas de formato.

## Antes del documento

Antes de empezar la edición del documento, además de cargar el fichero de estilo anterior, hemos creído necesario realizar una serie de operaciones que faciliten nuestro trabajo o lo configuren de una determinada manera. Además, hay que incluir la portada.

## Fuente del texto

Las instrucciones % establecen que el texto estará escrito en una fuente Times New Roman. El cambio de fuente se debe hacer modificando los estilos predefinidos en el documento.

## Cubierta

La primera hoja incluye una propuesta de cubierta. Puede modificarla, introduciendo una imagen central distinta, o introduciendo o sustituyendo algún logo y cambiando autores, títulos, departamento, etc.

# Referencias

A continuación detallaremos todas las referencias consultadas para el aprendizaje y conocimiento de las tecnologías que se van a usar para el desarrollo de este proyecto, y para el conocimiento teórico de muchos de lo

|  |  |
| --- | --- |
| [1] | Autor, «Este es el ejemplo de una cita,» *Tesis Doctoral,* vol. 2, nº 13, 2012. |
| [2] | O. Autor, «Otra cita distinta,» *revista,* p. 12, 2001. |

s conceptos que se aplican:

# **Índice de Conceptos**

conceptos 9

# **Glosario**

ISO: International Organization for Standardization 4

UNE: Una Norma Española 4

1. La portada es la primera hoja no en blanco que aparece después de la cubierta. Esta hoja también es obligatoria incluirla en todos los proyectos defendidos en la ETSI. [↑](#footnote-ref-1)
2. Esto se hace desde el menu *Referencias > Insertar nota al pie*. [↑](#footnote-ref-2)
3. En MacOS no suele aparecer y hay que recurrir a algún foro para ver cómo incluirla, por ejemplo http://ask.brothersoft.com/how-to-get-the-missing-toolbar-and-menu-back-331.html [↑](#footnote-ref-3)