

Software y estándares para la Web

PROYECTO CONVOCATORIA EXTRAORDINARIA 2024/2025

VERSIÓN 1.0 (07/02/2025) – REVISIÓN (03/04/2025)

Contenido

Temática	2
Estructura de archivos del proyecto	2
Menú de navegación del proyecto.....	3
HTML	4
CSS.....	5
Advertencias CSS	5
XML.....	6
Computación en el cliente usando ECMAScript.....	8
Computación en el servidor usando PHP	9
Pruebas obligatorias.....	10
Pruebas de funcionamiento y funcionalidad.	10
Cumplimiento de los estándares tanto del código estático como dinámico.....	10
Pruebas de Usabilidad.....	10
Pruebas de Adaptabilidad.	11
Pruebas de Accesibilidad.....	11
Pruebas de despliegue en la nube.	11
Documentación del Proyecto	12
Entrega del proyecto.....	12
Importante	13
Anexo 1. Concejos del Principado de Asturias	14

VERSIONES

1.0 (07/02/2025) – Versión inicial del documento

1.1 (03/04/2025) – Corrección del recuadro resumen del apartado XML (página 6)

El proyecto de la convocatoria extraordinaria será un proyecto web completo de la temática propuesta (no se admiten ejercicios/módulos independientes o de otra temática) y con la estructura especificada.

El proyecto es el mismo para la convocatoria extraordinaria adelantada (mayo) y para la convocatoria extraordinaria (junio)

El proyecto se compone de código, pruebas y documentación, la entrega parcial del proyecto tendrá la consideración de No presentado.

Temática

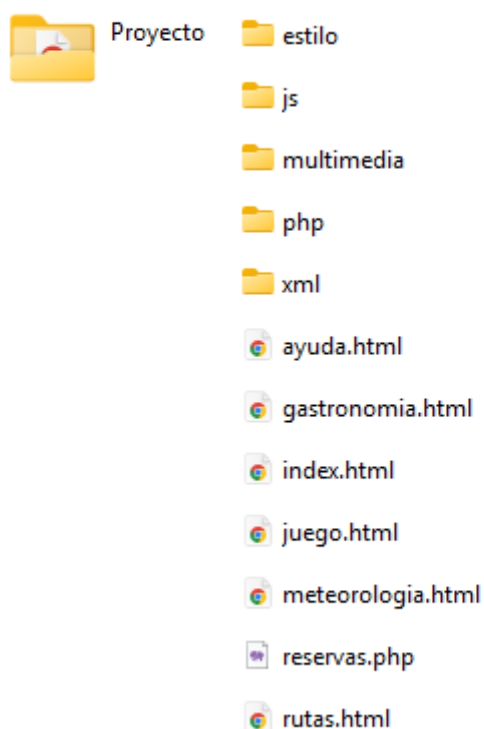
Se realizará un sitio Web sobre los recursos turísticos de un concejo del Principado de Asturias.

El estudiante deberá realizar el proyecto en función de todos los dígitos de su UO con la fórmula siguiente:

$$\text{CÓDIGO} = \text{X MÓDULO } 78 + 1$$

siendo **X** son todos los dígitos del UO y el **CÓDIGO** de la clasificación del Anexo 1.

Estructura de archivos del proyecto



Donde:

- Carpeta “estilo” contendrá las hojas de estilo “estilo.css” y “layout.css”
- La carpeta “js” contendrá los archivos JavaScript
- La carpeta “multimedia” contendrá todos los archivos multimedia del proyecto (imágenes, videos y audios)
 - o No se permiten enlaces a imágenes, vídeos o audios de sitios web externos o almacenados en OneDrive u otros sistemas en la nube
- La carpeta “php” contendrá los archivos php, el archivo .sql con las sentencias de creación de la base de datos, y archivo .csv con datos de ejemplo para las tablas.
- La carpeta “xml” contendrá los archivos de Tecnologías XML

Menú de navegación del proyecto

Todos los documentos del sitio web del proyecto debe tener un menú de navegación estático que muestre las opciones para el acceso a los recursos turísticos del concejo.

El menú de navegación debe aparecer en todas las páginas del proyecto, y en cada documento, se debe indicar el elemento activo del menú, para lo que está permitido la utilización del atributo class en HTML y el selector de clase en CSS.

Las opciones del menú de navegación deben ser:

- Página principal enlaza el documento index.html
- Gastronomía enlaza el documento gastronomia.html
- Rutas enlaza el documento rutas.html
- Meteorología enlaza el documento meteorologia.html
- Juego enlaza el documento juego.html
- Reservas enlaza el archivo reservas.php
- Ayuda enlaza el documento ayuda.html

HTML

Todos los documentos (estáticos y generados) que componen el proyecto deben cumplir el estándar HTML5, con una correcta estructuración de los contenidos y el marcado semántico correcto.

Se van a crear 2 documentos estáticos en HTML: gastronomía y ayuda.

El documento denominado “**gastronomia.html**” que será enlazado desde la opción del menú de navegación “Gastronomía”.

Este documento debe mostrar información relativa a la gastronomía típica del concejo asturiano. Debe tener **al menos** los siguientes elementos HTML:

- Lista ordenada con al menos 3 elementos
- Lista no ordenada con al menos 3 elementos
 - Uno de los elementos de la lista debe tener una lista anidada, pudiendo ser ordenada o no ordenada
- Lista de definición con al menos 3 elementos
- Tabla accesible y adaptable de 2 columnas y 4 filas de datos. Se permite el uso del atributo “id” para hacer la tabla accesible.
- Recursos multimedia:
 - Al menos una imagen adaptable y accesible
 - Al menos un vídeo adaptable y accesible servido con al menos 2 formatos diferentes. No es obligatorio poner subtítulos
 - Opcionalmente un audio adaptable y accesible servido con al menos 2 formatos diferentes. No es obligatorio poner subtítulos

El documento denominado “**ayuda.html**” será enlazado desde la opción del menú de navegación “Ayuda”.

Este documento contendrá la información para ayudar al usuario a manejar el sitio web con información sobre cada una de las opciones del menú y la funcionalidad creada. Este documento es necesario para las pruebas de usabilidad.

CSS

Las hojas de estilo deben ser aplicadas a todos los documentos que componen el sitio web.

La definición de selectores debe ser correcta y óptima, está prohibido el uso de ID y class como selectores de estilo salvo en las excepciones especificadas.

Todos los selectores de las hojas de estilo deben ir precedidos por su especificidad como un comentario.

Se deben añadir comentarios en las reglas de las hojas de estilo, justificando las advertencias del validador de CSS del W3C, en concreto, la herencia de colores (indicando de donde se hereda) y las redefiniciones de las propiedades, véase apartado “Advertencias CSS”

Se debe garantizar la adaptabilidad del sitio web.

Se deben crear únicamente 2 hojas de estilo:

- **estilo.css**: debe contener las características generales de estilo que se aplican a los elementos HTML de los documentos del sitio web.

Se debe tener en cuenta la herencia de las propiedades para realizar una correcta definición de la hoja de estilo.

- **layout.css** con las propiedades necesarias para la implementación del posicionamiento y la maquetación de los elementos HTML de todos los documentos que componen el sitio web, garantizando la adaptabilidad de los contenidos a través de la utilización de media queries.

Para la maquetación y posicionamiento solo se permite el uso de los módulos de CSS: [CSS-Flexible Box Layout], [CSS-Grid layout] y [CSS- CSS Multi-column Layout].

Advertencias CSS

Las hojas de estilo deben cumplir el estándar CSS3 sin errores ni advertencias. Hay 2 tipos de advertencias que están permitidas si y solo si están justificadas y documentadas en el código de las hojas de estilo, mediante comentarios, que son:

- Advertencias de la herencia de colores: Para garantizar que se cumple el correcto contraste de los colores para el nivel de accesibilidad AAA, en el archivo CSS, en la línea donde se produce la advertencia debe indicarse mediante un comentario de donde se hereda el color (de fondo o de primer plano) que provoca la advertencia.
- Advertencias por redefinición de propiedades: solo se permiten las derivadas del uso de media queries; en la línea donde se produce la advertencia debe indicarse mediante un comentario que esta provocada por la redefinición en la media query de la propiedad afectada.

El resto de las advertencias que puedan producirse en las hojas de estilo deben ser corregidas siempre, indican posibles deficiencias en la aplicación de las propiedades de la hoja de estilo que pueden provocar problemas en la visualización del sitio web.

XML

Debe diseñarse un archivo XML, de complejidad similar a las prácticas de la convocatoria ordinaria.

- Deberá contener validadores DTD y Schema. El Schema debe validar tipos de datos y será generado a partir del DTD.
- Se utilizará el lenguaje Python exclusivamente, para generar los archivos KML (planimetría) y SVG (altimetría) a partir de un archivo XML.
- Deberá traducirse a HTML usando ECMAScript y jQuery, para crear el archivo “**rutas.html**” que será enlazado desde la opción del menú de navegación “Rutas”. El KML y el SVG de cada ruta se deben visualizar en el HTML como un mapa y un elemento gráfico
- No está permitido la utilización de transformaciones XSLT por problemas de seguridad.
- No está permitido la utilización de lenguajes diferentes a los especificados

Se debe construir un archivo XML (**rutas.xml**) con **3 o más rutas** por el concejo asturiano a pie, en bicicleta u otro medio de transporte con paradas en los principales atractivos turísticos.

El archivo XML deberá tener los siguientes elementos o atributos:

- Nombre de la ruta turística (por ejemplo “Ruta de los molinos”)
- Tipo de ruta (por ejemplo “Arquitectura y monumentos”, “Gastronómica”, “Paisajística”, “Mixta tapas y monumentos”, “Escalada”, “Senderismo”, etc.)
- Medio de transporte (por ejemplo “A pie”, “Bicicleta”, etc)
- Fecha de inicio de la ruta (elemento o atributo marcado como opcional)
- Hora de inicio de la ruta (elemento o atributo marcado como opcional)
- Tiempo de duración de la ruta (por ejemplo “2 horas”, “3 días”, “2 semanas”)
- Agencia que gestiona la ruta (por ejemplo “Sin agencia”, “NaturaTur”)
- Descripción de la ruta (elemento que contiene texto)
- Personas adecuadas para la ruta (por ejemplo “Se puede ir con niños”, “Personas en buena forma física”, “tercera edad”, etc.)
- Lugar de inicio de la ruta (por ejemplo “Plaza del ayuntamiento”)
- Dirección de inicio de la ruta (por ejemplo “calle mayor”)
- Coordenadas geográficas de inicio de la ruta: longitud, latitud y altitud
- Referencias y bibliografía con información de la ruta (mínimo 3)
 - Referencia 1: por ejemplo <https://es.wikipedia.org/wiki/Foncalada>
 - Referencia 2: <http://prerrománicoasturiano.es/>
 - Referencia 3. etc.
- Recomendación de la ruta de 0 a 10 (por ejemplo “7”)
- Hitos de la ruta con algún tipo de atractivo turístico (**mínimo 5 hitos**):

- Nombre del hito
- Descripción del hito
- Coordenadas geográficas del hito: longitud, latitud, altitud
- Distancia desde el hito anterior (las unidades se expresarán como atributos)
- Galería de fotografías del hito (mínimo 1, máximo 5)
 - Fotografía 1: Por ejemplo Monumento.jpg
 - Fotografía 2: Por ejemplo Panorama.jpg
 - Fotografía 3: etc...
- Galería de vídeos del hito (opcional). Mínimo 0 y máximo 3.
 - Video 1: Por ejemplo Paisaje360.mpeg
 - Video 2: Por ejemplo Modelo3D.mpeg
 - Video 3: etc.
- Planimetría de la ruta usando el archivo KML generado con Python.
- Altimetría de la ruta usando el archivo SVG generado con Python.

El documento denominado “[rutas.html](#)” que será enlazado desde la opción del menú de navegación “Rutas”, cargará el archivo rutas.xml desde la máquina local (véase el apartado de Computación en el Cliente usando ECMAScript)

Computación en el cliente usando ECMAScript

- Deberá usarse obligatoriamente el paradigma de orientación a objetos en ECMAScript. No se admitirá el paradigma procedimental ni otros paradigmas soportados por ECMAScript.
- No se pueden utilizar bibliotecas externas (a excepción de jQuery)
- Deberá usarse jQuery obligatoriamente encapsulado dentro clases, objetos y métodos para mantener el paradigma de orientación a objetos.

En la página principal del proyecto, [index.html](#), deberá utilizar ECMAScript y la biblioteca jQuery para:

- Carrusel de fotos: un mínimo de 5 fotos con los principales recursos turísticos del concejo asturiano
 - Una de las imágenes del carrusel debe ser el mapa de situación del concejo
 - Las imágenes son archivos locales ubicados en la carpeta “multimedia”
- Sección de noticias sobre el concejo (o lugares cercanos al concejo): consumiendo Servicios Web

En la opción de “**Meteorología**” del menú de navegación, [meteorología.html](#), deberán consumirse Servicios Web utilizando la biblioteca jQuery para:

- Mostrar la información meteorológica en tiempo real relativa a la capital del concejo, o al lugar más cercano a dicha capital.
- Mostrar las previsiones meteorológicas en el concejo para los próximos 7 días, o al lugar más cercano al concejo.

En la opción de “**Juego**” del menú de navegación, [juego.html](#), deberá desarrollarse en ECMAScript puro un juego de 10 preguntas de tipo test con el objetivo de conocer la experiencia de usuario con el proyecto desarrollado:

- Cada pregunta con 5 opciones de respuesta y una sola respuesta correcta.
- Las preguntas versarán únicamente sobre información contenida en el sitio web del proyecto.
- Se obtendrá una calificación de 0 a 10 en función del número de respuestas acertadas (1 punto) o falladas (0 puntos).
- Es obligatorio que el jugador responda todas las preguntas.
- Al finalizar el juego, se debe mostrar la puntuación obtenida al jugador

En la opción de “**Rutas**” del menú de navegación, [rutas.html](#), deberá desarrollarse en ECMAScript puro y con la biblioteca jQuery, para cada una de las tres (o más) rutas:

- Carga y visualización de la información completa de la ruta mediante la carga de la información contenida en el archivo “rutas.xml”
- Carga y visualización de la planimetría de la ruta (archivo kml) sobre una aplicación de cartografía (por ejemplo, Google Maps, OpenLayers, OpenStreetMap,...)
- Carga y visualización de la altimetría de la ruta (archivo svg) señalando los principales hitos de la ruta con cadenas de texto y una línea que indique la cota de cero metros sobre el nivel del mar.

Computación en el servidor usando PHP

La computación en el servidor se utilizará para la creación y gestión de una Base de Datos usando obligatoriamente MySQL o MariaDB.

La base de datos debe tener un mínimo de 5 tablas relacionadas y normalizadas, y deberá entregarse el diagrama E-R en la **Documentación del proyecto**.

- Deberá usarse obligatoriamente el paradigma de orientación a objetos en PHP. No se admitirá el paradigma procedimental ni otros paradigmas soportados por PHP.
- Deberá usarse el sistema de gestión de bases de datos en el servidor
- No se pueden usar bibliotecas externas

El usuario y contraseña de la Base de Datos MySQL o MariaDB deben ser:

--- Usuario: DBUSER2025

--- Password: DBPWD2025

Además de los archivos .php, se deben crear en la carpeta “php” los archivos:

- Archivo .sql con las sentencias de creación de la base de datos.
- Archivo .csv con datos de las diferentes tablas que son necesarios para inicializar la base de datos

El objetivo es simular en PHP una central de reservas de los recursos turísticos. Se deben desarrollar los siguientes apartados de forma obligatoria:

- Registro de usuarios: mediante un formulario se debe permitir al usuario registrarse en la base de datos.
- Recursos turísticos: información sobre los diferentes recursos turísticos que se pueden reservar indicando para cada uno:
 - Límites de ocupación. Número de plazas del recurso turístico.
 - Fecha y hora de inicio y finalización del recurso turístico.
 - Precio del recurso turístico.
 - Breve descripción del recurso turístico.
- Reserva de recurso turístico:
 - El usuario debe estar registrado para realizar una reserva.
 - El usuario puede elegir el recurso turístico que desea reservar.
 - Se generará un presupuesto del recurso turístico.
 - Se confirmará la realización de la reserva.
- Consulta de los recursos turísticos reservados
 - Se generará un listado de las reservas de los recursos turísticos del usuario.
- Anulación de los recursos turísticos reservados:
 - Se permitirá anular reservas de los recursos turísticos del usuario.

Los recursos turísticos pueden ser de diferentes tipos: museos, rutas, restaurantes, hoteles, etc. Es necesario tener un mínimo de 5 recursos turísticos.

Pruebas obligatorias

Las pruebas que deben realizarse sobre el proyecto son:

- Pruebas de funcionamiento y funcionalidad.
- Cumplimiento de los estándares tanto del código estático como dinámico.
- Pruebas de Usabilidad.
- Pruebas de Adaptabilidad.
- Pruebas de Accesibilidad.
- Pruebas de despliegue en la nube.

Pruebas de funcionamiento y funcionalidad.

Se debe comprobar que se cumplen todos los requisitos de funcionamiento y funcionalidad del Proyecto.

Cumplimiento de los estándares tanto del código estático como dinámico.

Validación del cumplimiento del estándar HTML5 **sin errores ni advertencias**, mediante el validador del W3C

- Código estático (archivos html) comprobados mediante la opción en el validador de carga de archivos locales.
- El código html generado en computación web y tecnologías XML comprobados mediante la opción en el validador de pegar el código generado obtenido del inspector del agente de usuario añadiendo en el validador la sentencia Doctype

Validación del cumplimiento del estándar CSS3 **sin errores ni advertencias**, mediante el validador del W3C

- Se debe activar siempre la opción “Todas las advertencias”
- Las advertencias permitidas (herencia de colores y redefiniciones provocadas por el uso de media queries) deben controlarse y documentarse según se especifica en el apartado de CSS “Advertencias CSS”)

Pruebas de Usabilidad.

Se deben realizar las pruebas de Usabilidad del Proyecto, serán entregadas en el Documentación del Proyecto.

Las características de las pruebas de usabilidad a desarrollar son:

- Al menos 12 personas diferentes, indicando su edad, género y nivel de destreza en el uso de ordenador (de 0 a 10), en 3 tandas de 4 personas.
 - No deben identificarse con nombre y apellidos para respetar la protección de datos de carácter personal.
- Se deben guardar los tiempos (en segundos) que tarda cada usuario en realizar cada tarea.

- Se deben documentar las reacciones y comentarios de cada usuario en el uso del proyecto.
- Entre las tandas se debe especificar en el apartado de “Pruebas de usabilidad” de la documentación del proyecto, los cambios realizados para mejorar la usabilidad.
- Con cada usuario se deben repetir las pruebas en ordenador, tableta y móvil, siguiendo siempre el mismo orden, para que se tengan en cuenta las experiencias aprendidas en el uso del Proyecto.
- Los resultados de las pruebas de usabilidad se guardarán en una hoja de cálculo, una fila por usuario y en columnas cada una de las pruebas.
 - La hoja de cálculo debe ser entregada en el campus virtual
- Con los datos de la hoja de cálculo se harán estadísticas y gráficas con el objetivo de observar la mejora de la usabilidad según avanzan las tandas que deben ser documentadas en el apartado de “Pruebas de usabilidad” de la documentación del proyecto.
- Las tareas que debe realizar cada persona son:
 - TAREA1: Jugar al juego desarrollado.
 - TAREA2: El usuario debe encontrar la información sobre el tiempo de duración de la ruta turística de su interés.
 - TAREA3: Hacer una reserva de actividades turísticas durante una semana y obtener el presupuesto.

Pruebas de Adaptabilidad.

Se debe garantizar la adaptabilidad y realizar su verificación para todos los documentos que componen el proyecto mediante pruebas de visualización en ordenador, tableta y móvil.

Se pueden utilizar diferentes herramientas de visualización del Proyecto en diferentes dispositivos y resoluciones.

Pruebas de Accesibilidad.

Se deben realizar pruebas de accesibilidad utilizando dos herramientas, Wave y aChecker, garantizando no tener errores de accesibilidad de “**verificación automática**” para el nivel AAA.

- No pueden existir advertencias de contraste de colores
- Los warnings/advertencias generales de las herramientas están permitidos, dado que proporcionan información para la realización de una auditoría de accesibilidad manual.

La revisión de la accesibilidad debe realizarse sobre todos los documentos HTML estáticos y generados.

Pruebas de despliegue en la nube.

Se debe realizar el despliegue del proyecto completo en la nube y debe documentarse en el apartado “Despliegue en la nube” de la Documentación del Proyecto todos los pasos seguidos:

- Creación y configuración de la máquina virtual
- Creación y configuración de la BBDD
- Despliegue del Proyecto en la nube
- Comprobación del funcionamiento del despliegue del proyecto en la nube

Documentación del Proyecto

La documentación del proyecto debe incluir los siguientes apartados:

- **Portada** que contengan información del estudiante: nombre, apellidos, UO.
- **Tabla de contenidos** del documento
- Apartado “**Información del desarrollo del Proyecto**” en el que, de forma somera, se indique información relativa al desarrollo del proyecto realizado:
 - Concejo asignado
 - Comentarios relativos al desarrollo del proyecto: decisiones adoptadas, dificultades encontradas, etc.
- Apartado “**Diagrama Entidad – Relación**” de la base de datos desarrollada en el apartado de Computación web usando PHP.
- Apartado “**Pruebas de usabilidad**” con la documentación detallada de las pruebas de usabilidad desarrolladas.
- Apartado “**Despliegue en la nube**” con la documentación detallada del proceso de las pruebas de despliegue en la nube realizado.

Entrega del proyecto

El proyecto se compone de código, pruebas y documentación. La entrega del proyecto se ha dividido en tres tareas en el campus virtual por la limitación de 90MB establecido en el mismo:

- “**UOXXXXX-Codigo.zip**” con el contenido completo del código del proyecto desarrollado en la carpeta “Proyecto” con la **estructura especificada** en el apartado “Estructura de archivos del proyecto”:
 - Todos los documentos HTML
 - Las dos hojas de estilo CSS
 - Todos los recursos multimedia (imágenes, videos y audios)
 - Todos los archivos de tecnologías xml
 - Todos los archivos js
 - Todos los archivos php, csv y sql
- “**UOXXXXX-Documentación.pdf**” con el contenido en formato PDF especificado en el apartado “Documentación del Proyecto”.
- “**UOXXXXX-Usabilidad.xlsx**” con la hoja de cálculo creada en el desarrollo de las pruebas de usabilidad.

Importante

Se deben tener en cuenta las siguientes consideraciones:

- Se deben cumplir las pautas y normas de la asignatura, el no cumplimiento de estas invalidará el proyecto presentado.
- Tamaño máximo de cada uno de los archivos en el campus virtual es de **90MB**. Es responsabilidad del estudiante que el contenido de las entregas se adapte a ese tamaño.
- Todos los recursos multimedia utilizados deben estar en la carpeta Proyecto, no se admiten enlaces a OneDrive u otros sistemas en la nube.

Se considerarán como “No Entregados” los trabajos que:

- no cumplan el formato de entrega y el nombrado de los archivos y el contenido.
- realicen la entrega parcial del proyecto
- los archivos que estén protegidos con contraseña o con archivos corruptos
- el uso de usuario y password diferente a los especificados

Anexo 1. Concejos del Principado de Asturias

CODIGO	NOMBRE
1	ALLANDE
2	ALLER
3	AMIEVA
4	AVILÉS
5	BELMONTE DE MIRANDA
6	BIMENES
7	BOAL
8	CABRALES
9	CABRANES
10	CANDAMO
11	CANGAS DE ONÍS
12	CANGAS DE NARCEA
13	CARAVIA
14	CARREÑO
15	CASO
16	CASTRILLÓN
17	CASTROPOL
18	COAÑA
19	COLUNGA
20	CORVERA DE ASTURIAS
21	CUDILLERO
22	DEGAÑA
23	EL FRANCO
24	GIJÓN
25	GOZÓN
26	GRADO
27	GRANDAS DE SALIME
28	IBIAS
29	ILLANO
30	ILLAS
31	LANGREO
32	LAVIANA
33	LENA
34	LLANERA
35	LLANES
36	MIERES
37	MORCÍN
38	MUROS DE NALÓN
39	NAVA
40	NAVIA
41	NOREÑA
42	ONÍS
43	OVIEDO
44	PARRES
45	PEÑAMELLERA ALTA
46	PEÑAMELLERA BAJA

47	PESÓZ
48	PILÓN
49	PONGA
50	PRAVIA
51	PROAZA
52	QUIRÓS
53	LAS REGUERAS
54	RIBADEDEVA
55	RIBADESELLA
56	RIBERA DE ARRIBA
57	RIOS
58	SALAS
59	SAN MARTÍN DE OSCOS
60	SAN MARTÍN DEL REY AURELIO
61	SAN TIRSO DE ABRES
62	SANTA EULALIA DE OSCOS
63	SANTO ADRIANO
64	SARIEGO
65	SIERO
66	SOBRESOBIO
67	SOMIEDO
68	SOTO DEL BARCO
69	TAPIA DE CASARIEGO
70	TARAMUNDI
71	TEVERGA
72	TINEO
73	VALDÉS
74	VEGADEO
75	VILLANUEVA DE OSCOS
76	VILLAVICIOSA
77	VILLAYÓN
78	YERNES Y TAMEZA