

# Sistema de cronometrado y seguimiento de eventos deportivos basado en tecnologías RFID

Francisco Vicente Parra

Fecha: Junio de 2018

Tutor: Federico Simmross Wattenberg

Grado en Ingeniería de Tecnologías de Telecomunicación



---

**Universidad de Valladolid**

## Resumen

En los eventos deportivos basados en recorrer un circuito se emplean sistemas de cronometraje automatizados que registran los tiempos de los participantes. Estos sistemas permiten obtener los tiempos de cada participante y generar una clasificación en base a los mismos. Además, en algunos casos proporcionan herramientas para seguir el avance del evento en directo.

Nos encontramos con que muchos de estos sistemas de cronometraje deportivo requieren adquirir dispositivos y software que resultan bastante costosos. Por otro lado, existen alternativas más económicas, pero estas alternativas tienen debilidades que se traducen en errores de precisión, falta de fiabilidad y carencia de funcionalidades importantes.

En este Trabajo Fin de Grado se va a desarrollar un sistema de cronometraje que permita cronometrar de forma automática un evento deportivo, además de facilitar el seguimiento del mismo en directo. Una vez tengamos desarrollado el sistema, se pondrá a prueba mediante la simulación de varios eventos deportivos con participantes ficticios, tratando de representar en la medida de lo posible casos reales.

## Palabras clave

Cronometraje deportivo, RFID, Django, Raspberry Pi, MAX3232

## Abstract

In sports events based on a circuit, automated timing systems are used to record the times of the participants. These systems allow you to obtain the times of each participant and generate a classification based on them. In addition, in some cases they provide tools to follow the progress of the event live.

We find that many of these sports timing systems require the purchase of devices and software that are quite expensive. On the other hand, there are cheaper alternatives, but these alternatives have weaknesses that result in precision errors, lack of reliability and lack of important functionalities.

In this Final Degree Project, a timing system will be developed for automatic timing of sports events, in addition to provide live tracking. Once we have developed the system, it will be tested by simulating several sporting events with fictitious participants, trying to represent real cases as much as possible.

## Keywords

Sports timing, RFID, Django, Raspberry Pi, MAX3232

# Índice general

|  |           |
|--|-----------|
| <b>1. Introducción</b>   | <b>5</b>  |
| 1.1. Introducción . . . . .  | 5         |
| 1.2. Motivación . . . . .  | 6         |
| 1.3. Objetivos y metodología . . . . .   | 8         |
| 1.4. Método . . . . .  | 8         |
| 1.5. Medios necesarios . . . . .   | 9         |
| 1.5.1. Etiquetas y lector/antena RFID . . . . .  | 9         |
| 1.5.2. Miniordenador . . . . .   | 9         |
| 1.5.3. Conectores y adaptadores . . . . .  | 10        |
| 1.5.4. Servidor <i>web</i> . . . . .   | 10        |
| 1.6. Desarrollo de un sistema de cronometraje deportivo automático . . . . .   | 10        |
| 1.6.1. Funcionalidades del sistema . . . . .   | 10        |
| 1.6.2. Requisitos no funcionales del sistema . . . . .   | 11        |
| 1.6.3. Componentes del sistema . . . . .   | 12        |
| 1.7. Estructura de la memoria . . . . .  | 14        |
| <b>2. Técnicas y sistemas de cronometraje deportivo existentes</b>   | <b>17</b> |
| 2.1. Técnicas manuales de cronometraje . . . . .   | 17        |
| 2.1.1. Discusión . . . . .   | 18        |
| 2.2. Sistemas semiautomáticos de cronometraje . . . . .  | 18        |
| 2.2.1. RaceClocker . . . . .   | 19        |
| 2.2.2. Webscorer . . . . .   | 21        |
| 2.2.3. Discusión . . . . .   | 23        |
| 2.3. Sistemas automáticos de cronometraje . . . . .  | 23        |
| 2.3.1. Sistemas automáticos de cronometraje basados en la tecnología NFC . . . . .   | 23        |
| 2.3.2. Sistemas automáticos de cronometraje basados en la tecnología RFID en la banda UHF . . . . .                        | 25        |
| <b>3. Medios y tecnologías para desarrollar un sistema de cronometraje deportivo automático</b>                            | <b>35</b> |
| 3.1. Equipos de cronometraje . . . . .   | 35        |
| 3.1.1. Dispositivos para la identificación de participantes . . . . .  | 35        |
| 3.1.2. Dispositivos para la asignación de tiempos y comunicación con la aplicación web . . . . .                           | 40        |
| 3.1.3. Lenguaje de programación y bibliotecas para el desarrollo de la aplicación en los equipos de cronometraje . . . . . | 43        |
| 3.2. Aplicación <i>web</i> . . . . .   | 43        |
| 3.2.1. Software para el desarrollo de la aplicación <i>web</i> . . . . .   | 43        |
| <b>4. Desarrollo de los equipos de cronometraje</b>  | <b>49</b> |
| 4.1. Descripción . . . . .   | 49        |
| 4.2. Investigaciones previas . . . . .   | 50        |
| 4.2.1. Características de la <i>Raspberry Pi</i> para la comunicación . . . . .  | 50        |
| 4.2.2. Características del lector UHF RFID para la comunicación . . . . .  | 51        |

|   |            |
|---|------------|
| 4.2.3. Pruebas realizadas . . . . .   | 51         |
| 4.3. Manipulaciones Hardware . . . . .  | 55         |
| 4.3.1. Conexión <i>Raspberry Pi</i> - Módulo <i>MAX3232</i> . . . . .                 | 55         |
| 4.3.2. Conexión lector RFID - Módulo <i>MAX3232</i> . . . . .                         | 56         |
| 4.4. Desarrollo Software . . . . .  | 56         |
| 4.4.1. Configuraciones e instalaciones previas . . . . .                              | 56         |
| 4.4.2. Desarrollo de aplicación para <i>Raspberry Pi</i> . . . . .                    | 57         |
| 4.4.3. Instalación de aplicación en <i>Raspberry Pi</i> . . . . .                     | 60         |
| 4.5. Componentes y precios . . . . .  | 60         |
| <b>5. Desarrollo de la aplicación web</b>   | <b>61</b>  |
| 5.1. Descripción de la aplicación web . . . . .                                       | 61         |
| 5.1.1. Panel de administración . . . . .  | 61         |
| 5.1.2. Seguimiento de eventos deportivos . . . . .                                    | 63         |
| 5.2. Usuarios y acceso al sistema . . . . .   | 63         |
| 5.3. Casos de uso . . . . .   | 64         |
| 5.3.1. Diagrama de casos de uso . . . . .   | 64         |
| 5.3.2. Casos de uso detallados . . . . .  | 65         |
| 5.4. Diagrama de clases . . . . .   | 83         |
| 5.5. Desarrollo de la lógica y control del sistema de cronometraje . . . . .          | 84         |
| 5.5.1. Operaciones para determinar la validez de los envíos de cronometraje . . . . . | 84         |
| 5.5.2. Operaciones para generar clasificación con la tabla de tiempos . . . . .       | 85         |
| <b>6. Pruebas y simulaciones</b>  | <b>87</b>  |
| 6.1. Pruebas de software . . . . .  | 87         |
| 6.2. Simulaciones de eventos deportivos . . . . .                                     | 98         |
| <b>7. Conclusiones y líneas futuras</b>   | <b>101</b> |
| 7.1. Conclusión sobre el sistema desarrollado . . . . .                               | 101        |
| 7.2. Líneas futuras: mejoras a implementar en el sistema . . . . .                    | 102        |
| 7.2.1. Mejoras en el panel de administración de la aplicación web . . . . .           | 102        |
| 7.2.2. Mejoras en la pantalla de seguimiento público . . . . .                        | 102        |
| 7.2.3. Mejoras en los equipos de cronometraje . . . . .                               | 103        |



# Capítulo 1

## Introducción

### 1.1. Introducción

Los eventos deportivos son actividades que permiten practicar un deporte de forma colectiva. Estos eventos cuentan con participantes que realizan la actividad deportiva, y público que sigue el avance del evento deportivo ya sea de forma presencial o telemática, además de otras personas necesarias para la organización del evento.

En algunos eventos deportivos se practican deportes que consisten en recorrer circuitos, como por ejemplo ciclismo de montaña o *running*. En este tipo de deportes, el objetivo de los participantes es completar un circuito en el menor tiempo posible. Para conocer el tiempo que tarda cada participante en completar el circuito se realiza el cronometraje, que puede hacerse de forma manual apoyándose en un cronómetro o reloj, o de forma automática mediante soluciones tecnológicas, siendo el método automático más preciso y fiable.

Los eventos deportivos suelen estar divididos en categorías, que pueden basarse en rangos de edad, sexo, o simplemente la dificultad o longitud del recorrido a realizar. Por tanto, en algunos eventos deportivos los participantes realizan recorridos diferentes en función de la categoría a la que pertenecen.

En aquellos casos en los que un evento deportivo cuenta con varias categorías, y en cada categoría los participantes realizan un recorrido diferente, cabe la posibilidad de que un participante de una categoría del recorrido de mayor dificultad realice el recorrido de la categoría de menor dificultad, pudiendo conseguir un tiempo total inferior al que obtendría si hubiera realizado el recorrido que le corresponde, es decir, un participante podría hacer trampas.

La forma de evitar que los participantes puedan realizar esta clase de trampas consiste en controlar el paso de los mismos en lugares clave del recorrido. Para cumplir con este objetivo se suele realizar el cronometraje deportivo no solo en la meta, sino también en algunos puntos de control de paso intermedios.

Es conocido por aquellas personas aficionadas al deporte que en España se realizan cada vez con más frecuencia eventos deportivos destinados a participantes aficionados, conocidas habitualmente como carreras o marchas populares. Podemos encontrar carreras populares a lo largo de toda la geografía española, de diferente envergadura y dificultad, incluso en algunos casos podemos encontrar tanto participantes aficionados como profesionales.

Algunos de estos eventos deportivos son muy conocidos, como *la San Silvestre Vallecana* o *los 10.000 del Soplao*, que cuentan con un gran número de participantes, lo cual conlleva grandes contratos de patrocinadores y ayudas públicas. Por otro lado, existen eventos deportivos en pequeños pueblos de zonas rurales alejadas de las ciudades, que cuentan con menor número de participantes y por tanto menos ingresos por patrocinadores, y en muchos casos sin ayudas públicas. Entre estos dos extremos

de eventos deportivos, se realizan otros muchos de diferente envergadura, cantidad de participantes y presupuestos, tanto en ciudades como en zonas rurales.

Los participantes se vuelven cada vez más exigentes de cara a los organizadores de los eventos deportivos, con el propósito de disponer de servicios similares a los eventos deportivos profesionales. Sin embargo, los organizadores de eventos modestos, que no cuentan con un gran número de participantes, se encuentran en dificultades para proporcionar estos servicios, debido principalmente a sus limitaciones presupuestarias.

Entre los participantes que acuden a estos eventos deportivos se comenta que uno de los servicios cada vez más demandados es un sistema de cronometraje fiable que permita el seguimiento en directo de forma telemática.

## 1.2. Motivación

Actualmente, en la mayoría de eventos deportivos que se celebran en España se realiza el cronometraje deportivo. Para ello, habitualmente las organizaciones recurren a empresas especializadas en el cronometraje de este tipo de eventos. Aunque también encontramos algunos casos en los que las propias organizaciones realizan el cronometraje, ya sea de forma manual o de forma semiautomática apoyándose en tecnologías de bajo coste.

Las técnicas manuales de cronometraje son muy simples. La persona o personas encargadas del cronometraje se ubican en aquellos lugares en los que se toman los tiempos y, apoyándose en un cronómetro o reloj, van anotando en una lista los dorsales junto con las marcas de tiempo de cada participante. Una vez finalizado el evento, se revisan las listas de tiempos en la meta y en los diferentes puntos de control de paso en busca de posibles errores. Tras realizar las comprobaciones se publican los tiempos y las clasificaciones en papel o en Internet.

Estas técnicas manuales tienen el inconveniente del posible error humano a la hora de tomar los tiempos, y son imprecisas. Además, no es posible realizar un seguimiento de la prueba deportiva por medios telemáticos.

Las técnicas de cronometraje semiautomáticas se apoyan en tecnologías bastante económicas y su funcionamiento también es simple. La persona o personas encargadas del cronometraje se ubican en aquellos lugares en los que se toman los tiempos y, a través de una aplicación *web* o una aplicación móvil, van seleccionando los dorsales de los participantes en el momento en el pasan por el punto de control. La propia aplicación se encarga de elaborar la tabla de tiempos y la clasificación. Además, este tipo de aplicaciones suelen permitir el seguimiento en directo a través de Internet. Una vez finalizado el evento, se puede consultar el resultado en Internet, y también se pueden imprimir en papel.

Estas técnicas semiautomáticas tienen la ventaja de permitir un seguimiento en directo de la prueba deportiva. Sin embargo, al igual que las técnicas manuales, son propensas al error humano e imprecisas.

Para eventos deportivos de media y gran participación, tanto las técnicas de cronometraje manuales como las semiautomáticas se vuelven inservibles. Esto es debido a que en ambos casos el cronometraje depende de la habilidad de una persona a la hora de registrar el paso de cada participante, y en los casos en los que se deben registrar a muchos participantes al mismo tiempo es altamente probable cometer errores o dejar algún participante sin registrar.

Los sistemas de cronometraje automático suelen requerir medios tecnológicos de un coste elevado, por lo tanto las organizaciones no suelen adquirir estos medios sino que recurren a empresas externas para realizar el cronometraje. En España existen multitud de empresas que se encargan del cronometraje deportivo en carreras populares. Además, en la mayoría de los casos estas empresas proporcionan servicios adicionales como la gestión de las inscripciones, animación del evento deportivo, etc. Al ofrecer esta variedad de servicios, las empresas realizan presupuestos para cada evento, cuyo valor puede oscilar dependiendo de los requerimientos de cada caso.

Para llevar a cabo el cronometraje, estas empresas disponen de medios tecnológicos de cronometraje que han adquirido previamente a socios y proveedores, y que pueden emplear en diferentes eventos deportivos para amortizar la inversión. Normalmente, estos sistemas de cronometraje se basan en tecnologías RFID (*Radio Frequency Identification*). El funcionamiento básico en estos casos es el siguiente: cada participante lleva consigo una etiqueta o chip RFID adherido al dorsal, o a modo de pulsera en el tobillo o la muñeca, y esta etiqueta es leída por un lector RFID al llegar a la meta o al pasar por un punto de control intermedio.

Los lectores RFID habitualmente van conectados a un ordenador en el que se dispone de una aplicación desarrollada por el propio fabricante del lector o por terceros, que permite gestionar los participantes, registrar los tiempos de cada participante y elaborar una tabla de tiempos en el ordenador. Una vez que el evento finaliza, estas empresas pueden imprimir la tabla de tiempos en papel, y a su vez exportarla a PDF para su posterior publicación en Internet. Las aplicaciones desarrolladas para la gestión del evento y el cronometraje deportivo también pueden encontrarse en la nube, en forma de aplicación *web*. En estos casos, se puede seguir el evento deportivo en directo a través de aplicaciones móviles o un navegador *web*.

Al igual que ocurre en todos los negocios, existen empresas que proporcionan un servicio de cronometraje excelente y empresas que ofrecen un servicio de cronometraje de peor calidad. Obviamente, las empresas que proporcionan el mejor servicio también ofrecen los precios más elevados, mientras que las empresas que ofrecen el servicio de peor calidad resultan ser más económicas.

Las diferencias en cuanto a la calidad del servicio de cronometraje automático radican principalmente en los medios tecnológicos empleados por cada empresa para llevar a cabo el cronometraje. Aunque todas emplean sistemas de cronometraje basados en RFID, existen diferencias de calidad entre unos sistemas y otros tanto en los dispositivos *hardware*, como en el *software*. Las diferencias entre unos sistemas y otros radican principalmente en:

**Número de etiquetas RFID leídas por segundo.** Unos lectores RFID permiten realizar un gran número de lecturas de etiquetas por segundo, mientras que los lectores más económicos permiten leer un número de etiquetas menor por segundo. Esto implica que un lector mejor se podrá emplear para realizar el cronometraje en eventos deportivos con un gran número de participantes, mientras que un lector peor se podrá emplear únicamente en eventos deportivos de baja participación. Los eventos de alta participación no tienen problemas en contratar a empresas que disponen de medios capaces de detectar un número elevado de etiquetas por segundo, mientras que en los eventos de baja participación es posible realizar el cronometraje con lectores con menores tasas de etiquetas por segundo. De modo que esta diferencia técnica no supone una diferencia real entre las empresas más económicas y las empresas más caras a la hora de realizar el cronometraje en eventos deportivos de baja participación.

**Seguimiento del evento en directo.** Los medios técnicos que emplean las empresas más económicas no permiten el seguimiento del evento en directo de forma telemática, sino que únicamente se puede seguir en directo encontrándose físicamente en la meta o en el lugar donde esté ubicado el equipo de cronometraje. En estos casos, para poder conocer la tabla de tiempos es necesario que finalice el evento y posteriormente los tiempos sean publicados en Internet.

**Disponer de un único lugar en el que tomar tiempos.** Las empresas que ofrecen servicios de cronometraje más económicos disponen de un único equipo de cronometraje ubicado en un lugar, típicamente en la meta. De esta forma, sólo se puede conocer el tiempo total obtenido por los participantes, sin la posibilidad de conocer los tiempos obtenidos en puntos intermedios del recorrido.

Los organizadores de grandes eventos deportivos disponen de presupuestos lo suficientemente elevados como para poder permitirse contratar a empresas que proporcionan un cronometraje deportivo de la mejor calidad, con sistemas basados en la tecnología RFID que permiten tomar tiempos en meta y puntos de control intermedios, además de permitir el seguimiento a través de Internet. Sin embargo, los organizadores de los eventos deportivos más modestos, con bajo número de participantes, se encuentran con el problema de no contar con presupuesto suficiente para contratar a una empresa de cronometraje automático, y en caso de hacerlo recurren a las empresas más económicas que proporcionan estos servicios. Como vemos, las empresas que ofrecen servicios más económicos tienen carencias importantes a la hora de realizar el cronometraje, siendo las más importantes la imposibilidad de tomar tiempos en más de un lugar y la imposibilidad de poder seguir el evento deportivo en directo. Por otro lado, las organizaciones de eventos deportivos modestos podrían recurrir a técnicas de cronometraje manuales o a sistemas de cronometraje semiautomático, que también tienen carencias importantes.

Dadas las dificultades existentes para conseguir un cronometraje adecuado en eventos deportivos modestos, de baja participación, se propone desarrollar un sistema que sea capaz de realizar de forma automática el cronometraje en estos eventos deportivos. Este sistema debe ser válido para los eventos deportivos cuyas modalidades son más habituales entre eventos deportivos de este nivel: ciclismo de montaña y *running* sobre cualquier superficie.

### 1.3. Objetivos y metodología

El objetivo de este TFG es desarrollar un sistema de cronometraje automático, que pueda ser empleado en eventos deportivos de baja participación cuyas modalidades deportivas sean ciclismo de montaña y *running* sobre cualquier superficie. Para que el sistema de cronometraje sea válido en este tipo de eventos, debe cumplir con una serie de características que se analizan en capítulos posteriores.

Para proceder con el desarrollo de un sistema de cronometraje automático, se comenzará analizando los sistemas de cronometraje deportivo existentes que se vienen utilizando en eventos deportivos destinados a deportistas aficionados. Este análisis permitirá conocer las fortalezas y debilidades de cada sistema, además de ofrecer una idea sobre cómo funcionan y se estructuran estos sistemas.

Partiendo de la base de los sistemas analizados se podrá determinar qué tecnologías pueden ser aptas para desarrollar un sistema de cronometraje deportivo automático, así como elegir las adecuadas para cumplir los objetivos.

Finalmente, se realizará el desarrollo de un sistema de cronometraje. Para ello se seguirá el *proceso unificado* (UP), que emplea UML (*Unified Modeling Language*) como lenguaje de modelado. Se comenzará recopilando los requisitos del sistema, que permitirán definir los casos de uso, para posteriormente obtener un diagrama de clases y realizar el desarrollo *software*.

### 1.4. Método

Para llevar a cabo el proyecto se plantea emplear la tecnología RFID o identificación por radio frecuencia. Esta tecnología consiste en unas etiquetas RFID en combinación con unos lectores y antenas RFID. El propósito de esta tecnología es permitir la identificación de objetos mediante el uso de ondas de radio.

Los lectores RFID envían continuamente señales de radio para detectar si existe alguna etiqueta RFID en su área de cobertura. Una vez efectuada la detección de etiquetas, envían la información a través de una interfaz a otros equipos conectados para su procesamiento.

Las etiquetas RFID son elementos de tamaño pequeño que contienen una antena, y pueden responder a las peticiones emitidas por un lector RFID con su identificador. En el caso de las etiquetas pasivas pueden hacerlo sin alimentación interna.

En la implementación del sistema de cronometraje, los participantes llevarán una etiqueta RFID pasiva que permitirá su identificación, esta etiqueta irá pegada o incorporada al dorsal.

Estas etiquetas RFID serán detectadas por los lectores RFID que se encuentran en aquellos puntos del recorrido donde se desea realizar la medición de tiempos, también llamados puntos de control. Con esta implementación se obtiene el tiempo de cada participante a su paso por cada punto de control.

Cada lector RFID se conectará con un miniordenador, encargado de gestionar el funcionamiento del lector, además de obtener y procesar los datos de las etiquetas RFID detectadas. El miniordenador envía la información obtenida a una aplicación web, donde tanto los participantes como el público en general pueden seguir el avance del evento con los tiempos de cada participante en cada punto de control.

El sistema será utilizado por los administradores para gestionar los eventos deportivos y todos los aspectos relacionados con los mismos. Los participantes y el público en general emplearán el sistema para realizar el seguimiento de los eventos deportivos y consultar los tiempos obtenidos.

Para comprobar el funcionamiento del sistema de cronometraje, se realizarán varias pruebas que consistirán en simulación de eventos de diferentes características y varios participantes ficticios.

## **1.5. Medios necesarios**

Para llevar a cabo el desarrollo del sistema de cronometraje es necesario disponer de un conjunto de dispositivos, además de algunas herramientas y conectores.

### **1.5.1. Etiquetas y lector/antena RFID**

El sistema de cronometraje se fundamenta en la identificación automática de los participantes por medio de unas etiquetas o chips que llevan adheridos a los dorsales. Por tanto, es necesario disponer de etiquetas o chips RFID que permitan la identificación por radiofrecuencia, además de un lector/antena RFID que se encargue de la detección de las etiquetas.

Estos dispositivos deben permitir la detección de los participantes a una distancia de varios metros para no causar molestias ni retrasos a la hora de realizar la identificación, por tanto, deben operar en la banda UHF que es la que permite la identificación a la distancia requerida. Además, los dispositivos deben poder operar en exteriores, bajo condiciones meteorológicas adversas, por lo que deben estar suficientemente protegidos de las inclemencias climatológicas.

### **1.5.2. Miniordenador**

El lector RFID se encarga simplemente de la identificación de las etiquetas que se encuentran en su área de actuación. Sin embargo, para el sistema de cronometraje que pretendemos desarrollar, es necesario que el código identificativo de cada etiqueta que llevará asociado cada participante, esté asociado a una marca de tiempo que corresponda al instante de su lectura.

Además, la información identificativa de cada participante y la marca de tiempo, deben ser enviados a un módulo central del sistema de cronometraje para el seguimiento y clasificación de los eventos deportivos.

Estas tareas deben llevarse a cabo por un dispositivo con la potencia computacional suficiente para ello, y que además cuente con los puertos adecuados para el lector RFID. Además, debe tener un coste reducido para cumplir con el objetivo de ser un sistema de cronometraje económico. Por tanto, se propone emplear un miniordenador que cumpla con las condiciones mencionadas, y este es el caso de una Raspberry Pi 2 ó Raspberry Pi 3.

### 1.5.3. Conectores y adaptadores

Para la comunicación entre el miniordenador y el lector RFID será necesario emplear un conjunto de cables y adaptadores, que en todo caso, deberán tener un coste reducido.

### 1.5.4. Servidor *web*

El módulo central del sistema de cronometraje consistirá en una aplicación *web* encargada tanto de permitir la gestión todos los aspectos de los eventos deportivos, así como generar las clasificaciones y tablas de tiempos a partir de la información que envíen los miniordenadores. Por tanto, es necesario disponer de un servidor que soporte la aplicación *web*.

## 1.6. Desarrollo de un sistema de cronometraje deportivo automático

El sistema de cronometraje deportivo automático a desarrollar para este TFG tiene por objetivo ofrecer una solución económica y de calidad para el cronometraje en eventos deportivos de baja participación cuyas modalidades deportivas sean el ciclismo de montaña y *running*, aunque también podrá ser aplicado a otras modalidades deportivas similares.

Se incorpora en esta sección una introducción extensa al desarrollo del sistema de cronometraje ya que el desarrollo se divide en dos capítulos, un capítulo correspondiente al desarrollo de la parte encargada de la identificación de participantes, y un capítulo correspondiente a la parte encargada de generar la clasificación y tabla de tiempos.

### 1.6.1. Funcionalidades del sistema

#### 1.6.1.1. Permitir el seguimiento en directo

Tanto las técnicas de cronometraje manuales como los sistemas de cronometraje automático más económicos existentes, no permiten realizar un seguimiento en directo de forma telemática. En estos casos, para poder seguir el avance de la prueba es necesario estar físicamente en la meta o en los puntos intermedios del recorrido que sean de interés.

El sistema a desarrollar debe permitir el seguimiento de los eventos deportivos en directo desde dispositivos con conexión a Internet y un navegador *web*. Esta característica hará posible que cualquier persona pueda conocer desde cualquier lugar los tiempos que van obteniendo los participantes a su paso por los puntos de control intermedios y en el momento de su llegada a meta.

#### 1.6.1.2. Permitir varios puntos de control

La mayor limitación de los sistemas de cronometraje automáticos más económicos existentes es la imposibilidad de cronometrar en más lugares que en la línea de meta. La necesidad de permitir varios puntos de control viene motivada por las siguientes razones:

- En el caso de que en un evento deportivo se realicen recorridos diferentes. En algunos eventos deportivos existen diferentes categorías, y en cada categoría se realiza un recorrido diferente. Cuando este es el caso, un participante podría estar inscrito en una categoría que tiene el recorrido más largo y complejo, pero después realizar el recorrido más corto y fácil. De esta forma, y al tomar únicamente tiempos en la meta, el participante obtendría un tiempo que no se corresponde con la realidad. Esto se podría solventar empleando un punto de control de paso intermedio en el recorrido de mayor distancia.
- En el caso de un evento deportivo en el que no hay recorridos diferentes, pero sí la posibilidad de tomar atajos, existe la posibilidad de que un participante acorte el recorrido saliendo en un punto del mismo y volviendo a incorporarse más adelante. De esta forma, un participante podría atajar y por tanto realizar menos distancia que el resto, obteniendo por tanto un tiempo incorrecto en meta. Esto se podría solventar empleando puntos de control de paso intermedios en el recorrido.
- Además, en algunos casos los participantes esperan conocer en qué parte del recorrido han sido mejores, para su superación personal. Esta opción es interesante sobre todo cuando hay partes del recorrido bien diferenciadas.

### 1.6.1.3. Funcionalidades para la gestión de eventos deportivos

El sistema de cronometraje a desarrollar ha de proporcionar funcionalidades para realizar el cronometraje de eventos deportivos de varios deportes, así como proporcionar las herramientas necesarias para la configuración y gestión de los eventos deportivos:

- El sistema debe permitir ser empleado para múltiples eventos deportivos, por lo tanto, debe contener un panel de administración desde el cual se gestionen los mismos.
- Cada evento deportivo, a su vez, puede contener una o varias categorías. Por tanto, el sistema debe proporcionar funcionalidades para gestionar las categorías de cada evento.
- En cada categoría se puede realizar un recorrido diferente o el mismo recorrido. Por tanto el sistema debe proporcionar funcionalidades para gestionar los recorridos de un evento deportivo, y además, permitir asociar a cada categoría el recorrido correspondiente.
- En los recorridos de un evento deportivo se pueden tomar tiempos en un lugar o en varios, lo que se conoce como puntos de control. Por tanto, el sistema ha de permitir configurar para cada recorrido los puntos de control necesarios.
- La parte más importante de un sistema de cronometraje son los participantes de la prueba a cronometrar. El sistema debe proporcionar funcionalidades para gestionar los participantes de un evento deportivo, así como asignar la categoría correspondiente y el identificador que permita tomar los tiempos.

## 1.6.2. Requisitos no funcionales del sistema

### 1.6.2.1. Sistema económico

La mayor dificultad con la que se encuentran las organizaciones de eventos deportivos de baja participación a la hora de realizar el cronometraje deportivo es el coste del mismo. Por tanto, el sistema a desarrollar debe estar basado en componentes y tecnologías que resulten económicas y accesibles para su adquisición.

De este modo, los eventos deportivos que no cuenten con una gran financiación o, en su caso, las empresas que se encargan de cronometrar estos eventos deportivos, podrán emplear el sistema.

#### 1.6.2.2. Sistema modular

El sistema debe tener la capacidad de trabajar con uno o varios puntos de control en un mismo evento deportivo. Para conseguir este objetivo, es necesario que el sistema a desarrollar esté dividido en módulos o subsistemas que se comunican entre sí, y puedan operar de forma independiente. El objetivo es disponer de dos tipos de subsistemas:

- **Subsistemas encargados de tomar tiempos.** Estos subsistemas son los que se encargan de realizar la medición de los tiempos de cada participante a su paso por un punto de control, y enviar la información a otro subsistema. La complejidad de los mismos debe ser la mínima posible para facilitar su configuración en día del evento. Es necesario disponer de varios subsistemas encargados de tomar tiempos ya que es necesario cronometrar en varios puntos de control de un recorrido.
- **Subsistema principal.** El objetivo es que este subsistema se base en una aplicación *web* accesible desde cualquier lugar, y que sea el núcleo de todo el sistema de cronometraje.

#### 1.6.2.3. Fácil de configurar e instalar

El sistema debe permitir ejecutar todas las configuraciones necesarias para la realización de un evento deportivo de forma fácil. Este es un aspecto clave, ya que una complejidad elevada a la hora de instalar y configurar el sistema provocaría que no fuera viable su uso.

El sistema a emplear debe permitir colocar los chips o etiquetas que identifiquen a los participantes en el dorsal. Esto simplificará la labor del organizador y se podrán evitar errores del participante a la hora de su colocación.

#### 1.6.2.4. Operabilidad en exteriores

El sistema debe poder ser ubicado en exteriores, soportando las inclemencias del tiempo. Por tanto, los componentes físicos del sistema deben contar con un nivel mínimo de protección ante polvo y líquidos.

#### 1.6.2.5. Precisión y fiabilidad

El sistema debe ser capaz de proporcionar tiempos con una precisión lo suficientemente buena para un evento deportivo de aficionados, lo cual se significa precisión de segundos.

El sistema debe ser fiable a la hora de detectar a los participantes, con una probabilidad alta de lectura.

#### 1.6.2.6. Sin contacto

El sistema debe ser capaz de tomar los tiempos de los participantes sin necesidad de que éstos se detengan o deban variar su rumbo para ser detectados.

### 1.6.3. Componentes del sistema

Como se ha mencionado anteriormente, el sistema de cronometraje permite tomar tiempos en uno o varios puntos de control de un mismo recorrido. Para lograr esta característica, es necesario que el sistema o una parte del mismo, se encuentre ubicado en cada punto de control del recorrido. Esto nos lleva a la necesidad de dividir el sistema en varios subsistemas o módulos.

En el sistema de cronometraje que se desarrolla para este TFG, cada módulo del sistema que se encuentra ubicado en cada punto de control se denomina equipo de cronometraje.



De cara a que todos los equipos de cronometraje puedan operar conjuntamente, de forma que en cada evento deportivo todos los equipos de cronometraje estén claramente asignados a los puntos de control correspondientes, y todos los tiempos obtenidos se puedan visualizar desde un mismo lugar, se emplea un subsistema central o principal encargado de comunicarse con cada uno de los equipos de cronometraje. Además, este subsistema central permite proporcionar adecuadamente funcionalidades del sistema como gestionar los eventos deportivos y el seguimiento en directo.

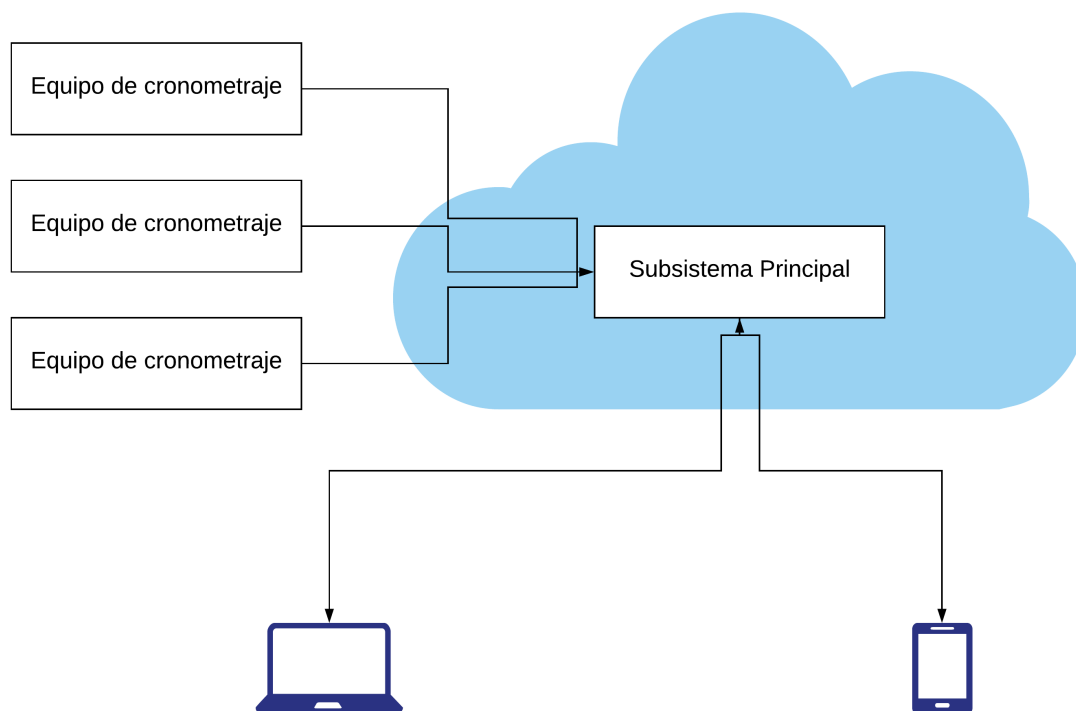


Figura 1.1: Esquema de los subsistemas en el sistema de cronometraje automático.

#### 1.6.3.1. Equipos de cronometraje

Los equipos de cronometraje son los módulos encargados de identificar a los participantes a su paso por cada punto de control, así como asignar sus marcas de tiempo correspondientes. Por tanto, en un evento deportivo cada equipo de cronometraje podrá ser asignado a un punto de control.

Cada vez que un equipo de cronometraje identifica a un participante, le asigna una marca de tiempo, y posteriormente transmite esta información al subsistema central.

En la figura 1.2 se muestra un ejemplo de recorrido que sirve para explicar el uso de los equipos de cronometraje. El recorrido se realiza en el sentido de las agujas del reloj, comenzando en la salida, pasando por los dos puntos de control intermedios y finalizando en meta. En este caso, se deben tomar tiempos en en punto de control 1, punto de control 2 y meta.

En cada uno de los puntos en los que se toman tiempos, se ubica un equipo de cronometraje, que se encarga de tomar tiempos y comunicarlos al subsistema central. Dentro del subsistema central, cada equipo de cronometraje deberá estar asignado a un punto de control del recorrido (punto de control 1, punto de control 2 y meta), de esta forma el subsistema central podrá asignar las marcas de tiempo enviadas por cada equipo de cronometraje al punto de control correspondiente dentro del recorrido.

Cada equipo de cronometraje se compone de un lector/antena RFID *INNOD IND8001* conectado a una *Raspberry Pi*, además de los cables y conectores necesarios que se han analizan en el capítulo 3.

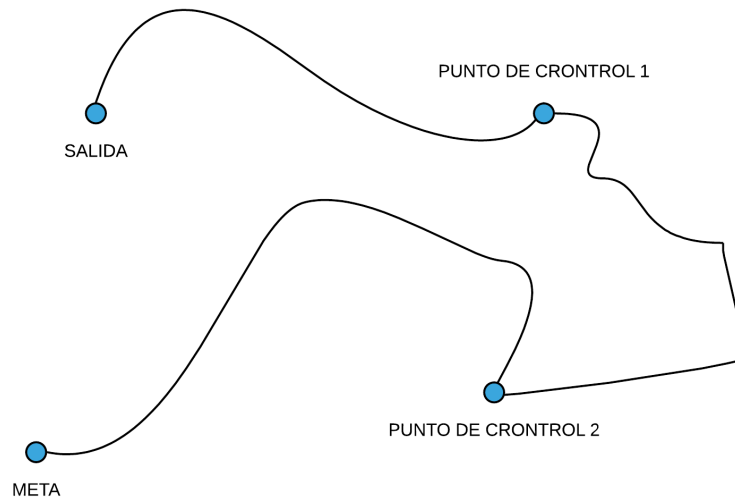


Figura 1.2: Ejemplo de recorrido con dos puntos de control intermedios.

Los equipos de cronometraje de este sistema permiten ser empleados en diferentes eventos deportivos, para diferentes puntos de control, sin necesidad de realizar ninguna configuración en los equipos. Su instalación y configuración el día del evento deportivo se reduce a conectar la alimentación eléctrica y el cable *Ethernet*.

Todas las instalaciones y configuraciones que se deben realizar en un *Equipo de Cronometraje* para que pueda formar parte del sistema se reducen a las necesarias para ser usado por primera vez. El desarrollo e implementación de los equipos de cronometraje se analiza en el capítulo 4.

#### 1.6.3.2. Subsistema central

En el subsistema central reside la mayor parte de la lógica y control del sistema de cronometraje. Este subsistema permite al administrador gestionar y configurar todos los detalles necesarios para llevar a cabo eventos deportivos, además de permitir el seguimiento en directo de los mismos.

El subsistema central se encarga de recibir la información procedente de los equipos de cronometraje, determinar si la información es válida, y en caso de que así sea asignar los tiempos recibidos a los participantes adecuados, en los puntos de control adecuados.

Para lograr satisfacer los requerimientos de este subsistema, se ha optado por realizar un desarrollo basado en una aplicación *web*. En el capítulo 5 se analiza como se ha llevado a cabo este desarrollo.

### 1.7. Estructura de la memoria

En esta memoria se lleva a cabo un estudio del estado del cronometraje deportivo en la actualidad, además de describir y analizar el desarrollo de un sistema de cronometraje deportivo que solventa algunos problemas encontrados en los sistemas actuales.

En el capítulo 2 se analizan las técnicas y sistemas de cronometraje deportivo que existen en la actualidad. Este capítulo se divide en tres secciones que agrupan técnicas y sistemas de cronometraje de complejidad técnica incremental: Técnicas manuales, sistemas semiautomáticos y, sistemas automáticos. La sección de sistemas automáticos, a su vez, se divide en dos grandes subsecciones ya que es necesario diferenciar entre sistemas que se apoyan en dos tecnologías diferentes.

En el capítulo 3 se analizan los medios y tecnologías que podrían servir para desarrollar un sistema de cronometraje deportivo automático. Este capítulo se divide en dos grandes secciones que reflejan los dos componentes fundamentales del sistema de cronometraje propuesto: equipo de cronometraje y aplicación *web*.

El desarrollo del sistema de cronometraje propuesto se analiza en los capítulos 4 y 5. Se ha optado por dividir el desarrollo en dos capítulos debido a la división del sistema de cronometraje propuesto en dos partes y la gran independencia entre las partes a la hora de realizar el desarrollo: equipo de cronometraje y aplicación *web*.

En el capítulo 6 se documentan varias pruebas y simulaciones realizadas a fin de determinar la validez del sistema de cronometraje propuesto.

En el capítulo 7 se extraen conclusiones y se exponen posibles líneas futuras a fin de mejorar el sistema de cronometraje desarrollado.



## Capítulo 2

# Técnicas y sistemas de cronometraje deportivo existentes

Hoy en día, en la mayoría de eventos deportivos que se celebran en España se lleva a cabo el cronometraje deportivo. Para realizar el cronometraje, los organizadores de estos eventos pueden optar por múltiples opciones, que a su vez implican diferentes costes tanto en términos económicos como en términos de personas necesarias para llevarlo a cabo.

El método más simple y antiguo es realizar el cronometraje de forma manual. Si bien este TFG trata sobre un sistema de cronometraje automático, es necesario analizar el funcionamiento de las técnicas de cronometraje manuales porque constituyen una base sólida para el funcionamiento de los sistemas de cronometraje automáticos y semiautomáticos.

También analizamos algunos sistemas semiautomáticos, ya que suponen un salto de calidad entre los sistemas manuales y los automáticos, además para algunos eventos deportivos pueden resultar una opción muy interesante.

### 2.1. Técnicas manuales de cronometraje

Las técnicas manuales de cronometraje tienen un funcionamiento muy simple, y apenas necesitan emplear tecnologías para llevar a cabo el cronometraje de un evento deportivo.

Para realizar el cronometraje de forma manual es suficiente con disponer de varias personas encargadas de llevarlo a cabo, y disponer de relojes sincronizados u otros elementos que puedan proporcionar la hora de forma precisa, como pueden ser *smartphones* o *tablets*. Es necesario contar con tantas personas como puntos de control en los que se vaya a tomar tiempos en una prueba deportiva, además de una persona en la línea de salida.

Cuando se da la salida del evento deportivo, la persona encargada del cronometraje en la salida anota el tiempo. Este es el caso más habitual cuando se realiza salida en masa. En el caso de que los participantes no salgan todos a la vez sino que lo hagan de forma escalonada, la persona encargada en la salida debe ir anotando en una lista el dorsal y la hora de salida de cada participante.

En cada punto de control intermedio y en la meta, la persona encargada del cronometraje va anotando en una lista el dorsal y la hora de paso de cada participante.

Una vez que el evento deportivo ha finalizado, las personas encargadas del cronometraje se reúnen y analizan todas las listas que han ido realizando durante la realización del evento. Se comprueba participante a participante para verificar que éstos han pasado por todos los puntos de control necesarios en su categoría, y una vez hechas las comprobaciones se procede a realizar las clasificaciones y tablas de tiempos.

Para realizar las clasificaciones y tablas de tiempos, se dividen los participantes por categorías, se ordenan por hora de llegada a meta, y se calculan los tiempos obtenidos tanto en la meta como en los puntos de control intermedios restando a la hora anotada en las listas la hora de salida.

Una vez que están realizadas las clasificaciones y tablas de tiempos, se pueden publicar en papel, o transcribir a formato electrónico para su publicación en Internet.

### **2.1.1. Discusión**

Estas técnicas manuales tienen la ventaja de no necesitar inversión económica, aunque sí necesitan disponer de personal suficiente. En los eventos deportivos de baja participación el personal necesario no suele ser un problema ya que cuentan con bastantes personas voluntarias dispuestas a ayudar.

Por otro lado, tienen el inconveniente del posible error humano a la hora de tomar los tiempos, y son imprecisas. Además, no es posible realizar un seguimiento de la prueba deportiva por medios telemáticos.

## **2.2. Sistemas semiautomáticos de cronometraje**

Los sistemas semiautomáticos de cronometraje automatizan una parte importante del cronometraje, mientras que se siguen apoyando en las personas encargadas para llevarlo a cabo. Algunas empresas consideran estos sistemas como manuales [Rac18] [Inc18]. Sin embargo, se ha considerado tratarlos como semiautomáticos para diferenciarlos de los sistemas puramente manuales, ya que permiten automatizar una parte importante del cronometraje, mejorando la precisión y fiabilidad.

Al igual que las técnicas de cronometraje manuales, los sistemas de cronometraje automáticos requieren disponer de varias personas encargadas de llevarlo a cabo, tantas como puntos de control en los que se vaya a tomar tiempos en una prueba deportiva, además de una persona en la línea de salida. Por otro lado, es necesario disponer de una aplicación *web* o una aplicación móvil instalada, lo que conlleva que cada persona encargada disponga de un dispositivo con conexión a Internet, y en su caso, la aplicación instalada.

El coste económico de estos sistemas suele ser bajo, ya que requiere un pago único o por suscripción de la aplicación, y en ambos casos resulta económico. Al igual que las técnicas de cronometraje manuales, se requiere bastante personal voluntario para llevarlo a cabo, lo cual no suele ser un problema en eventos deportivos de baja participación.

Para emplear sistemas de cronometraje semiautomático, es necesario registrar y configurar los detalles del evento deportivo a cronometrar en la aplicación, con sus categorías, recorridos, participantes, etc. Una vez configurado el evento deportivo en la aplicación, este es el proceso que se sigue el día del evento:

1. Para el caso de la salida en masa, la persona encargada pulsa el botón de comienzo en la aplicación web o en la aplicación móvil en el instante en el que se da a salida. Para el caso de la salida escalonada, la persona encargada va pulsando en la aplicación el botón asignado a cada dorsal de cada participante en el momento en el que toma la salida.
2. En cada punto de control intermedio y en la meta, la persona encargada del cronometraje va pulsando en la aplicación el botón asignado a cada dorsal de cada participante en el momento en el pasa.
3. La propia aplicación se encarga de elaborar la tabla de tiempos y la clasificación. Además, este tipo de aplicaciones suelen permitir el seguimiento en directo a través de Internet. Una vez finalizado el evento, se puede consultar el resultado en Internet, y también se pueden imprimir en papel.

A continuación analizamos algunos ejemplos de sistemas de cronometraje semiautomáticos.

### 2.2.1. RaceClocker

*RaceClocker* [Rac18] es un sistema de cronometraje deportivo semiautomático de pago, que se basa en una aplicación en la nube accesible por medio de un navegador web. Este sistema permite tanto gestionar el cronometraje de eventos deportivos, como seguir el evento en directo desde un navegador web.

El funcionamiento de este sistema sigue la metodología explicada anteriormente en los sistemas de cronometraje semiautomáticos.

La gestión y configuración necesarias para cronometrar un evento deportivo empleando este sistema se realizan a través de un panel de administración. El panel de administración proporciona las siguientes funcionalidades:

- Permite crear nuevos eventos deportivos, modificar o eliminar eventos deportivos existentes. Los eventos deportivos pueden tener los siguientes parámetros: Nombre de la prueba, fecha y hora, deporte, distancia, número de puntos de control intermedios, tipo de salida, sitio web y ubicación.
- Permite añadir nuevos participantes, modificar o eliminar participantes existentes. Los participantes pueden tener los siguientes parámetros: Nombre, número, categoría y club.

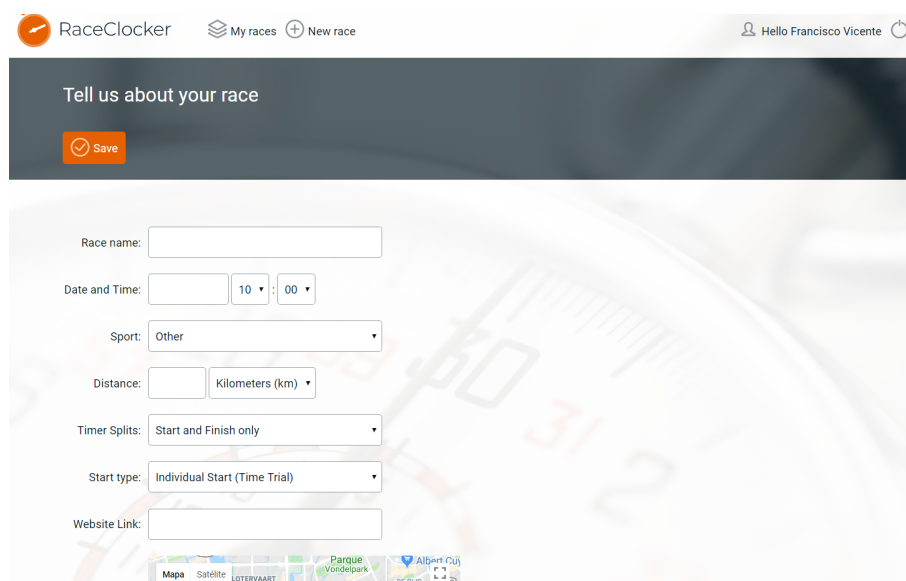
The image shows a web browser displaying the RaceClocker application. At the top, there's a navigation bar with the RaceClocker logo, 'My races', a '+ New race' button, and a user profile section saying 'Hello Francisco Vicente'. Below this is a dark grey banner with the text 'Tell us about your race' and an orange 'Save' button with a checkmark. The main content area is a form for creating a race. It includes fields for 'Race name', 'Date and Time' (with a date picker and a time selector set to 10:00), 'Sport' (a dropdown menu currently showing 'Other'), 'Distance' (a text input followed by a unit dropdown set to 'Kilometers (km)'), 'Timer Splits' (a dropdown menu set to 'Start and Finish only'), 'Start type' (a dropdown menu set to 'Individual Start (Time Trial)'), and 'Website Link'. At the bottom of the form, there is a map showing the location, with labels for 'Mapa', 'Satellite', 'LOTTERVAART', 'Parque Vondelpark', and 'Alberici'. The map also shows a location pin and some coordinates.

Figura 2.1: Aplicación web de RaceClocker vista desde un ordenador. Registro de un evento deportivo

Este sistema de cronometraje cuenta con algunas características de gran interés:

- Seguimiento en tiempo real. Los resultados de los eventos deportivos pueden ser seguidos en tiempo real a través de la aplicación web.
- Importación de participantes. La aplicación web permite importar los participantes del evento deportivo en formato CSV o Excel.
- Corrección de errores. Se permite la modificación manual las marcas de tiempos de los participantes, así como añadir o eliminar marcas de tiempo.
- Tomar tiempos en varios puntos de control. Permite configurar varios lugares o puntos de control donde tomar tiempos en un evento deportivo. Permite configurar salida, meta, y hasta 3 puntos de control intermedios.

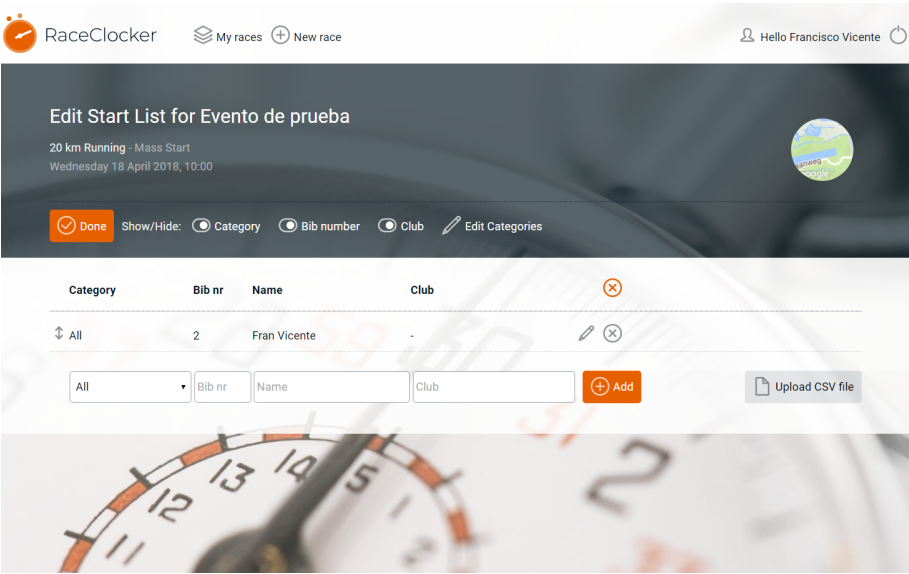


Figura 2.2: Aplicación web de RaceClocker vista desde un ordenador. Registro de participantes

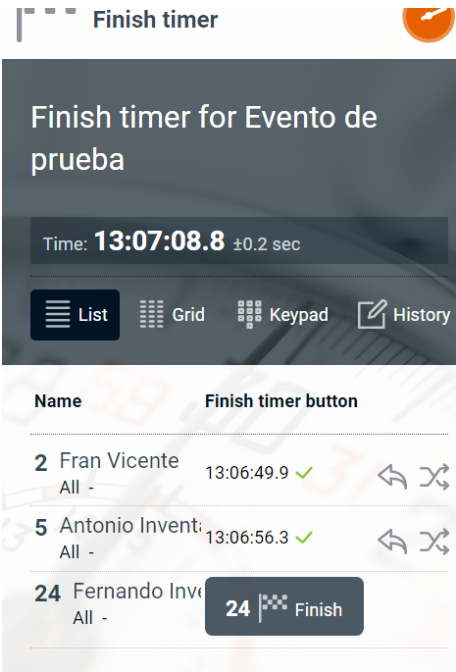


Figura 2.3: Aplicación web de RaceClocker vista desde un Smartphone. Panel de cronometraje en el punto de meta.



- Salidas escalonadas. Permite configurar tanto salidas en masa en las que todos los participantes toman la salida al mismo tiempo, como salidas escalonadas, en las que los participantes toman la salida en diferentes instantes de tiempo.
- Exportar resultados. Los resultados de un evento deportivo pueden ser publicados *online*, y también pueden ser exportados en formato Excel.

Una carencia importante de este sistema de cronometraje es la imposibilidad de configurar varios recorridos en un evento deportivo, lo cual conlleva a que los participantes de todas las categorías deben realizar un mismo recorrido.

RaceClocker dispone de una versión gratuita que permite cronometrar eventos deportivos de hasta 10 participantes, y una versión de pago económica con el formato de pago por suscripción (alrededor de 50 € al año) que permite cronometrar eventos de hasta 500 participantes.

### 2.2.2. Webscorer

*Webscorer* [Inc18] es una empresa que desarrolla sistemas de cronometraje deportivo. Dispone de un sistema de cronometraje semiautomático que se analiza en este apartado, y dos sistemas de cronometraje automático que se analizarán más adelante.

El sistema de cronometraje semiautomático de Webscorer funciona de forma similar al sistema RaceClocker, pero en lugar de disponer de aplicación web, proporciona una aplicación para dispositivos móviles que puede ser instalada en dispositivos con sistema Android o iOS. A través de la aplicación se pueden configurar y gestionar los eventos deportivos, así como registrar los tiempos de los participantes al pasar por los puntos de control.

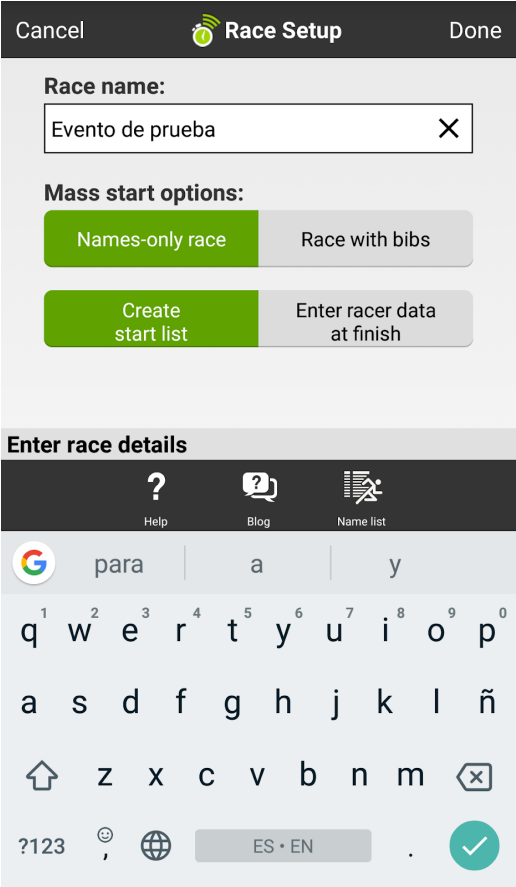
Las aplicaciones de los dispositivos móviles empleados para el cronometraje se encuentran en constante comunicación con un servidor en la nube al que le envían los registros de cronometraje, lo cual permite generar la clasificación y la tabla de tiempos.

El sistema de cronometraje semiautomático de webscorer incluye todas las funcionalidades que hemos analizado en el sistema RaceClocker, además, dispone de algunas funciones que hacen que Webscorer sea un sistema de cronometraje más avanzado:

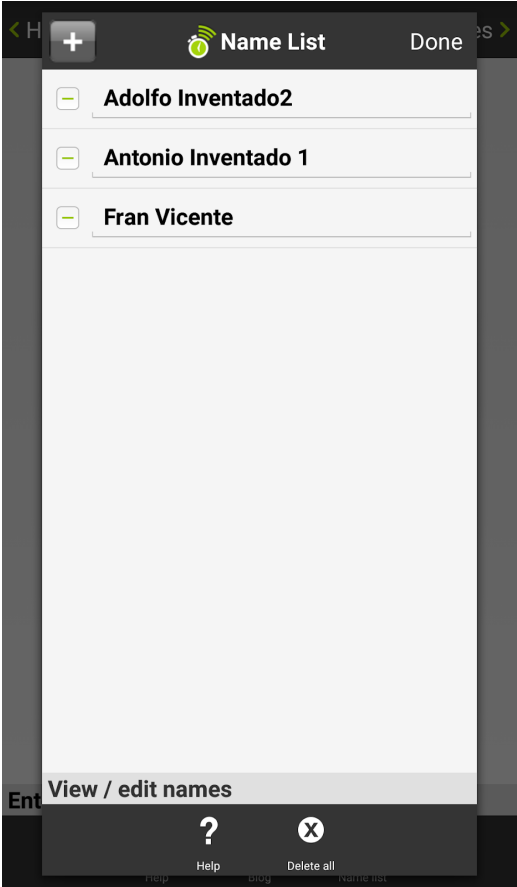
- Permite salidas en masa, salidas individuales, salidas a intervalos, etc.
- Permite configurar múltiples categorías y recorridos para cada evento.
- Ofrece la posibilidad de penalizar y bonificar a participantes.
- Permite la posibilidad de generar clasificaciones por equipos.

Webscorer también cuenta con un sistema de inscripciones *online*, donde los participantes pueden inscribirse por sí mismos a través de una web. Este sistema de inscripciones se puede combinar con el sistema de cronometraje para no tener que introducir a todos los participantes manualmente.

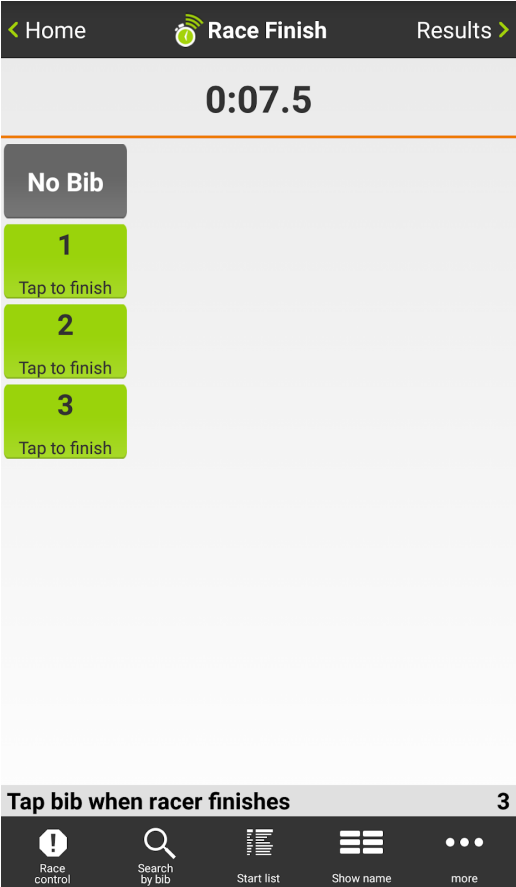
Webscorer es una aplicación de pago por suscripción anual, con un coste de 50 € al año para una licencia que permite configurar hasta 5 dispositivos. También cuenta con una opción gratuita con numerosas limitaciones, que a efectos prácticos no permite el cronometraje de casi ningún evento deportivo.



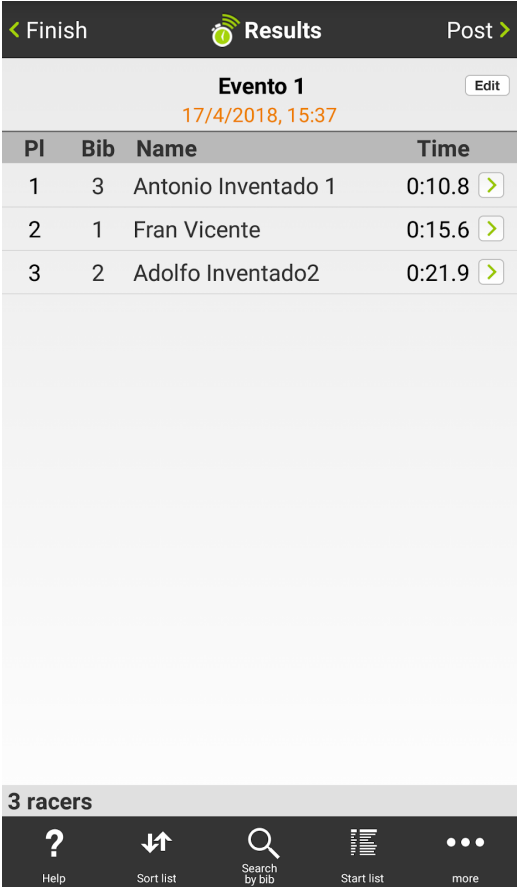
(a) Creación de evento deportivo.



(b) Gestión de participantes de un evento.



(c) Registro de los tiempos de los participantes en línea de meta.



(d) Resultados de un evento deportivo finalizado.

Figura 2.4: Aplicación Webscorer.

### 2.2.3. Discusión

Estos sistemas pueden ser de utilidad para algunos eventos deportivos de baja participación en los que los participantes se encuentran bastante dispersos, y además la organización del evento deportivo cuenta con suficiente personal para encargarse del cronometraje en cada punto de control. Sin embargo, los tiempos tomados no son precisos, ya que la precisión depende de la habilidad de las personas encargadas del cronometraje. Además, en caso de que varios participantes se encuentren muy próximos entre sí en el paso por la meta o por un punto de control intermedio, es de esperar que sus tiempos tomados sufran variaciones de algunos segundos respecto de sus tiempos reales debido a los retardos causados por la habilidad de las personas encargadas del cronometraje.

Como punto a favor de estos sistemas semiautomáticos, pueden permitir un seguimiento en directo de la prueba deportiva.

## 2.3. Sistemas automáticos de cronometraje

Los sistemas de cronometraje deportivo automático permiten automatizar por completo la realización del cronometraje deportivo, aportando con ello un alto grado de precisión y fiabilidad. Además, estos sistemas pueden permitir el seguimiento en directo del evento deportivo.

En comparación con los sistemas de cronometraje considerados semiautomáticos, los sistemas de cronometraje automáticos aportan la automatización del proceso de identificación y asignación de tiempos de los participantes a su paso por los puntos de control intermedios y en su llegada a meta. En los sistemas semiautomáticos, esta parte del cronometraje depende de las personas encargadas y su habilidad. Por esta razón los sistemas automáticos pueden aportar mayor fiabilidad y precisión.

La mejora real de los sistemas de cronometraje automáticos en cuanto a fiabilidad y precisión dependerá en primer lugar de las tecnologías empleadas para la identificación de los participantes y, en segundo lugar, de la calidad de los dispositivos que implementan dichas tecnologías.

Del mismo modo, tanto el coste económico del sistema de cronometraje, como las necesidades en cuanto a personal necesario para llevarlo a cabo, dependen de las tecnologías empleadas para la identificación de los participantes y de la calidad de los dispositivos que implementan dichas tecnologías. Los sistemas de cronometraje deportivo que se emplean actualmente en eventos deportivos, emplean tecnologías de identificación por radiofrecuencia. Estas tecnologías permiten la identificación automática de personas, animales u objetos que llevan aparejados un chip o etiqueta mediante el uso de un lector. Los sistemas de cronometraje deportivo existente pueden emplear dos tecnologías: NFC o RFID.

### 2.3.1. Sistemas automáticos de cronometraje basados en la tecnología NFC

Los sistemas de cronometraje deportivo basados en la tecnología NFC consisten en un *software*, además de unos dispositivos *hardware*. El software proporciona todas las funcionalidades necesarias para la gestión del evento, así como generar la clasificación y tabla de tiempos, de forma similar al software de los sistemas semiautomáticos. El hardware se encarga de la identificación automática de los participantes a su paso por un punto de control intermedio o en la meta, por medio de la tecnología NFC.

NFC (*Near Field Communication*) es una tecnología que permite la identificación automática, y de corto alcance. Para llevar a cabo la detección de los participantes en un evento deportivo, cada participante lleva un chip o pulsera NFC que es leído en cada punto de control por un dispositivo móvil que cuenta con la tecnología NFC, además de la aplicación del sistema de cronometraje empleado instalada.

Al ser una tecnología de corto alcance, los participantes deben detener la marcha y acercar el chip al dispositivo móvil que se encarga de la lectura en cada punto de control intermedio y a su llegada a meta.

Estos sistemas de cronometraje no requieren una gran inversión económica, ya que el *hardware* consiste en unos dispositivos móviles con NFC, lo cual es algo que cualquier persona puede tener hoy en día. A nivel del personal necesario para llevar a cabo el cronometraje, es necesario disponer de una persona por cada punto en el que se toman tiempos.

Estos sistemas ofrecen un mayor nivel de fiabilidad y precisión que los sistemas semiautomáticos. Además, permiten seguir el evento en directo de forma telemática.

NFC permite la identificación por radiofrecuencia de objetos o personas que se encuentran cerca del dispositivo con el que se lee. Este aspecto aplicado al campo del cronometraje deportivo implica que los participantes deban detenerse en los puntos de control donde se toman tiempos para poder ser identificados. Además, el hecho de ser leídos por una app móvil implica que debe haber una persona en cada punto de control encargada de manejar la app y asegurarse de que cada participante es identificado correctamente.

Este sistema de cronometraje puede ser de gran utilidad en eventos deportivos de muy baja participación en los que además los participantes se encuentran bastante distanciados unos de otros, de forma que el hecho de perder tiempo para la identificación no influya en los tiempos de cada participante.

Si este sistema se empleara en eventos deportivos donde hubiera un número mayor de participantes, o donde pudieran pasar varios participantes seguidos en un punto de control, unos participantes tendrían que esperar a que los participantes que van inmediatamente delante terminaran su identificación para poder ser identificados.

A continuación analizamos algunos ejemplos de sistemas de cronometraje automáticos que emplean la tecnología NFC.

#### **2.3.1.1. Webscorer NFC**

Webscorer es una empresa que desarrolla sistemas de cronometraje de diferentes tipos, como ya se ha analizado en los sistemas semiautomáticos.

El sistema de cronometraje automático basado en NFC consiste en la aplicación para dispositivos móviles que se ha analizado en el sistema semiautomático de Webscorer, que permite en sus ajustes emplear el lector NFC del dispositivo. Esta característica únicamente está disponible para dispositivos con sistema operativo Android.

Cuando un chip NFC es leído, la aplicación obtiene automáticamente el identificador y una marca de tiempo, lo cual permite realizar de forma automática la operación que en sistema semiautomático realizaba la persona encargada por medio de pulsar un botón en la pantalla.

El coste de este sistema es una suscripción anual de 250 €.

#### **2.3.1.2. Evorama**

*Evorama* [SL18] es una *startup* de Barcelona que se dedica al cronometraje deportivo. Disponen de un sistema completo de cronometraje formado por un *software* para la gestión de los eventos deportivos basado en una aplicación web, y unos sistemas de detección basados en la tecnología NFC.

**Software Evorama** Se trata de un *software* de seguimiento y gestión de eventos deportivos cronometrados que permite:

- Gestión del evento deportivo para el organizador: gestión de participantes, creación de recorridos, configuración de puntos de control, etc.
- Visualización de tiempos en directo multidispositivo (web y móvil): paso de los participantes en los distintos puntos de control establecidos, clasificación, evolución de la posición en la clasificación respecto a la última detección, posibilidad de seleccionar los corredores favoritos, posibilidad de compartir resultados en redes sociales, etc.

**Sistema de detección Evorama** Para llevar a cabo la detección de los participantes en un evento deportivo, cada participante lleva un chip-pulsera NFC que es leído en cada punto de control por un dispositivo móvil que lleva instalada la app de Evorama.

- Aplicación móvil fácil e intuitiva para el usuario.
- Envío de datos a tiempo real a través de Internet. La transmisión de datos se puede realizar con muy poca señal y en caso que no haya cobertura, el dispositivo almacena los datos y cuando puede los envía.

Esta empresa proporciona servicios de cronometraje directamente a organizaciones de eventos deportivos bajo un presupuesto, por lo que no podemos conocer su coste. No ofrece su sistema ni de forma gratuita, ni de pago a otras empresas.

### 2.3.2. Sistemas automáticos de cronometraje basados en la tecnología RFID en la banda UHF

Los sistemas de cronometraje deportivo basados en la tecnología RFID en la banda UHF consisten en un software y unos dispositivos hardware. El software proporciona todas las funcionalidades necesarias para la gestión del evento, así como generar la clasificación y tabla de tiempos, de forma similar al software de los sistemas semiautomáticos. El hardware se encarga de la identificación automática de los participantes a su paso por un punto de control intermedio o en la meta, por medio de la tecnología RFID.

RFID (*Radio Frequency Identification*) [Cur+07] en la banda UHF (*Ultra High Frequency*) es una tecnología que permite la identificación automática, y de largo alcance. Para llevar a cabo la detección de los participantes en un evento deportivo, cada participante lleva un chip o etiqueta RFID que es leído en cada punto de control por un dispositivo formado por una antena y un lector RFID, además de algún dispositivo con la aplicación del sistema de cronometraje empleado instalada.

A diferencia de la tecnología NFC que requería a los participantes detener la marcha y acercarse al lector para ser detectados, la tecnología RFID UHF permite detectar a los participantes a varios metros de distancia, sin necesidad de detenerse para ello.

Estos sistemas de cronometraje requieren una inversión económica mucho mayor que todas las alternativas anteriores, ya que es necesario adquirir antenas y lectores RFID que suelen resultar bastante costosos, además de necesitar ir acompañados de otros dispositivos para poder operar dentro de un sistema de cronometraje.

Estos sistemas ofrecen mayor grado de fiabilidad y precisión que los sistemas analizados anteriormente. Es por ello que los sistemas de cronometraje basados en tecnologías RFID UHF son los empleados en la mayoría de eventos deportivos en los que se practican deportes activos, incluidos los de mayor envergadura.

Actualmente en España existen multitud de eventos deportivos populares en los que se realiza un cronometraje automático con tecnologías RFID UHF. Normalmente cada evento se realiza una vez al año, por lo que no son los organizadores de los eventos deportivos quienes disponen de los medios necesarios para el cronometraje, sino que contratan a empresas estos servicios.

Se pueden encontrar numerosas empresas que se dedican al cronometraje deportivo en eventos deportivos populares y semiprofesionales. Estas empresas pueden permitirse disponer de los medios necesarios para realizar el cronometraje, ya que amortizan su coste acudiendo a cronometrar múltiples eventos deportivos. Además, en la mayoría de los casos estas empresas proporcionan servicios adicionales como la gestión de las inscripciones, animación del evento deportivo, etc.

Al igual que ocurre en todos los negocios, existen empresas que proporcionan servicios de diferentes calidades y presupuestos, la calidad del sistema de cronometraje viene dada por los medios técnicos empleados para ello. Por supuesto, las empresas que ofrecen un cronometraje de mejor calidad ofertan presupuestos más altos que las empresas que ofrecen un cronometraje peor.

En el mercado del cronometraje deportivo basado en la tecnología UHF RFID existen diversos fabricantes y soluciones de diferentes tipos que compiten entre sí.

Algunas empresas ofrecen unos sistemas de cronometraje completos y cerrados, con soluciones tanto de software como de hardware que cubren necesidades de todo tipo. Estas suelen ser las empresas con más experiencia en el sector, ofrecen los mejores sistemas, y a su vez, los más caros, ya que sus opciones más económicas parten de varios miles de euros. Las empresas más representativas de este tipo son MyLaps y ChronoTrack (grupo Athlinks). Ambas empresas proporcionan productos y servicios similares y son competencia directa. En este TFG analizaremos únicamente los sistemas de MyLaps por ser la empresa más representativa en su categoría.

Por otra parte, se encuentran soluciones tanto a nivel Software (instalables en dispositivos móviles u ordenadores), como componentes Hardware, que permiten interconectarse por medio del protocolo LLRP (*Low Level Reader Protocol*) para formar un sistema de cronometraje basado en tecnologías RFID. Analizaremos algunas opciones para comprobar su funcionamiento y el coste real de formar un sistema de cronometraje basado en este tipo de soluciones.

### 2.3.2.1. MYLAPS

*MYLAPS* [Tim18c] es una empresa holandesa creada en 1982 que se atribuye el mérito de inventar el cronometraje deportivo automático. Esta empresa desarrolla productos y servicios para medir, publicar y analizar los resultados de las competiciones. Es una empresa de gran reconocimiento, que se encarga del cronometraje de grandes eventos deportivos de nivel internacional como las 20 Horas de Le Mans, el Giro de Italia y las Olimpiadas.

Ofrecen todo tipo de sistemas y servicios de cronometraje. Disponen de soluciones adaptadas a las necesidades de los cronometradores, desde soluciones para deportistas profesionales hasta el nivel amateur. Proporcionan sistemas que permiten registrar, identificar a cada corredor o atleta, determinar sus tiempos de salida, llegada y parciales, recopilar datos de rendimiento, calcular clasificaciones y publicar los resultados.

Esta empresa dispone de una red de socios en todo el mundo que emplean su tecnología para realizar el cronometraje deportivo, es decir, empresas asociadas que se dedican al cronometraje deportivo en eventos de un nivel no tan alto, y que cuentan con la tecnología de MYLAPS para llevarlo a cabo.



(a) Decodificador.



(b) SideAntennas.

Figura 2.5: Componentes básicos del sistema BibTag.

MYLAPS dispone de tecnologías que permiten el cronometraje de todo tipo de eventos deportivos, sin embargo únicamente vamos a analizar la tecnología destinada a deportes activos como el running o el ciclismo, ya que este es el área de interés para este TFG.

### Sistema BibTag

El sistema *BibTag* [Tim18a] de *MYLAPS* es un sistema de cronometraje diseñado para minimizar la logística y es extremadamente fácil de ser utilizado por participantes. Características del sistema:

- Sirve para carreras a pie, competiciones ciclistas y eventos multidisciplinarios.
- Número ilimitado de participantes y puntos de control.
- Las alfombras modulares pueden cubrir pistas y carreteras de hasta 16 metros de anchura.
- Puede operar con aplicaciones de teléfono y servicios de MYLAPS.
- Altas tasas de lectura.

Componentes del sistema BibTag:

**Alfombras de detección** Las alfombras de detección se colocan a la salida, meta y en los puntos de control intermedios. Las alfombras hacen la función de antenas del sistema detectando las señales que envían los chips BibTag. Ofrecen alfombras modulares que permiten una anchura de entre 1 y 8 metros.

**SideAntennas** Las SideAntennas tienen la misma funcionalidad que las Alfombras de detección. Se diferencian en la forma y ubicación de las mismas. Las SideAntennas son antenas de tipo panel que se ubican en un lado del circuito a la salida, meta y en los puntos de control intermedios. La estructura de la SideAntenna se mantiene en una posición baja en el suelo y permite colocar el decodificador en la parte posterior de la estructura para crear una configuración compacta y fácil de manipular.

**Decodificador** El decodificador va conectado a las alfombras y se encarga de decodificar las señales de los chips BibTag que permiten identificar los participantes y sus respectivos tiempos. Los decodificadores portátiles tienen una batería integrada y pueden medir hasta 50 pasadas por segundo.

**Chips BibTag** El chip se adhiere al reverso del dorsal de participante y emite una señal única. Cada chip emite una señal diferente en el espectro UHF que permite a las alfombras de detección identificar al participante y registrar el momento exacto en que cruza la alfombra. Los chips tienen un diseño que los hace resistentes a la lluvia y el sudor, ya que su parte mecánica está cubierta de un separador de espuma. Estos chips son de usar y tirar, para minimizar la manipulación antes y después de la carrera. Sin embargo, si fuera necesario, los chips se pueden recoger y reutilizar en un evento posterior. Características principales:

- Solución de cronometraje fiable para eventos masivos.
- No se requiere manipulación después de la carrera.
- Manipulación mínima antes de la carrera.
- Alta fiabilidad.
- Muy ligeros.
- Tecnología RFID UHF.

**ThinTags** Estos componentes se emplean con el mismo propósito que los chips BibTag, y son muy similares. El ThinTag utiliza una espuma aislante compuesto por un material esponjoso que solo se expande cuando absorbe fluidos. Esta tecnología está patentada por MYLAPS. Los ThinTags se pueden imprimir con una impresora de la marca MYLAPS creada específicamente para ello.

**MultiSports Tags** Estos componentes se emplean con el mismo propósito que los chips BibTag y los ThinTags, sin embargo presentan grandes diferencias ya que han sido diseñados para triatlones y carreras de barro. Los MultiSports Tags pueden ser reutilizables o desechables, en ambos casos tienen forma de pulsera y van ubicados en la muñeca del participante.

**Software** MYLAPS dispone de un software que se encarga del cronometraje y clasificación. Cronometrar una carrera y generar resultados fiables es todo un desafío. Ofrece un paquete de software completo y fácil de usar para configurar y cronometrar una carrera y, después, generar y publicar resultados fiables.

El software se ha diseñado especialmente para carreras de atletismo y triatlón. Permite configurar múltiples salidas, tramos, carreras y recorridos en un mismo evento y ofrece diagnósticos muy detallados. Es fácil de usar y permite tanto a cronometradores profesionales como a aficionados configurar y cronometrar una carrera y crear y publicar resultados fiables en muy pocos «clicks».

**Accesorios** MYLAPS cuenta con varios accesorios que pueden ser empleados para facilitar el cronometraje de un evento deportivo. Estos son:

- **Botón de salida.** Este botón genera un indicador de salida al software marcando así el comienzo de la carrera. Este dispositivo se conecta por cable en el decodificador BibTag y está listo para comenzar.
- **Lector USB.** El lector se puede usar para comprobar los datos del participante, para verificar fácilmente si un participante se ha registrado para la carrera o cualquier otra comprobación que sea necesaria.
- **Impresora ThinTags.** Esta impresora permite ahorrar dinero al ofrecer la posibilidad de imprimir los propios tags, puede ser rentable si se utiliza mucho.



### Sistema ProChip

El sistema de cronometraje *ProChip* [Tim18d] de *MYLAPS* es rápido, fiable y fácil de usar. El sistema sirve para una gran variedad de deportes activos, como ciclismo, carreras a pie, triatlón, natación, etc.

Este sistema cuenta con características que le hacen ser más preciso que el sistema BibTag, sobre todo en carreras con llegadas muy competidas. El sistema ProChip mide los tiempos de los participantes con una precisión de milisegundos y determina al instante quién tiene la victoria. Las posiciones de llegada de todos los participantes se pueden publicar fácilmente en papel, en marcadores y en Internet.

Los componentes del sistema ProChip son similares a los del sistema BibTag, pero con ligeras diferencias que hacen que este sistema sea más preciso:

**Decodificador** Se diferencia del decodificador BibTag en que puede enviar los tiempos a un ordenador con el software de cronometraje instalado o a un servidor de Internet. Existe una versión estándar y una versión portable que puede funcionar sin conexión a un ordenador.

**Detection Loop** Cumple la misma funcionalidad de los SideAntennas y las Alfombras de detección. Tiene la ventaja de permitir un ancho de circuito mayor, hasta 100 metros. También tiene otras características que le permiten ser empleada en deportes que se practican en la nieve.

**ProChip FLEX** Son los componentes que llevan los participantes y permiten su identificación. Se ubican en la tobillera, la muñequera o la bicicleta (en una posición del cuadro de la bicicleta) para poder identificar a cada participante y registrar el momento exacto en que atraviesa las líneas de salida, de meta y de control intermedio. Diseñado para deportes de élite a gran velocidad.

**Software** El sistema ProChip es compatible con el software que se emplea en el sistema BibTag. Además, existen otros dos paquetes de software que pueden operar con el sistema ProChip: Orbits Cycling y Orbits 4.

**Software Orbits Cycling** Es un paquete de software para cronometraje y clasificación de eventos de ciclismo. El software se puede usar para múltiples modalidades de ciclismo, como el de carretera, bicicletas de montaña, ciclismo en pista y otras disciplinas ciclistas. Características:

- Permite configurar los eventos deportivos.
- Permite crear trayectos e importar los participantes.
- Cronometraje de las carreras.
- Procesamiento de los resultados.
- Herramientas para publicar los resultados: Imprimir y publicar los resultados de los eventos y subirlos a mylaps.com.
- Gestión de campeonatos.
- Múltiples categorías en un mismo evento.
- Permite asignar tiempos de grupo, herramienta de utilidad para llegadas en pelotón.
- Permite asignar bonificaciones.

### Software Orbits 4

El software Orbits 4 se utiliza para deportes activos a gran velocidad. Es un paquete de software completo y fácil de usar. Incluye asistentes sencillos que ayudan a empezar rápido y a configurar la aplicación según sus preferencias. Características:

- Configuración de eventos.
- Registro de participantes.
- Cronometraje de eventos.
- Procesamiento de resultados.
- Distribución de resultados.
- Gestión de campeonatos.

### Sistema ChampionChip

El sistema *ChampionChip* [Tim18b] se diseñó para carreras de participación masiva. Mide tiempos netos y permite gestionar una gran densidad de competidores en maratones y otros eventos deportivos masivos.

Este sistema cuenta con una complejidad mayor en la parte de los decodificadores que los dos anteriores analizados, y está diseñado para eventos deportivos de participación masiva. Por tanto sus características detalladas no son de gran interés para este TFG, y no serán analizadas.

### Discusión

Los productos y servicios de MYLAPS proporcionan un sistema completo para realizar el cronometraje de eventos deportivos de diferentes tipos. Esta empresa ha conseguido desarrollar un ecosistema de componentes compatibles entre sí, y en su mayor parte incompatible o de difícil funcionamiento con componentes de otros desarrolladores y fabricantes.

Emplear cualquiera de los sistemas de MYLAPS es una opción interesante en casos donde el presupuesto del evento deportivo es lo suficientemente alto, ya que cuenta con un sistema completo y fiable. Sin embargo, los eventos deportivos que disponen de un presupuesto más ajustado no pueden permitirse estos servicios.

#### 2.3.2.2. Webscorer RFID

El sistema de cronometraje de Webscorer analizado anteriormente también puede ser configurado para realizar el cronometraje deportivo con tecnologías RFID. De nuevo, el coste de la suscripción anual es de 250 €. Este sistema de cronometraje puede funcionar mediante el uso de dispositivos Hardware compatibles.

El modo de funcionamiento de este sistema de cronometraje es el siguiente:

- Los participantes del evento deportivo llevan aparejada una etiqueta RFID. En la aplicación de Webscorer se les asigna manualmente un identificador basado en el EPC de la etiqueta RFID.
- Un lector RFID compatible con Webscorer se encuentra conectado a un router con WIFI, por medio de un cable Ethernet o WIFI.
- Un dispositivo móvil (Android o iOS) que tiene instalada la aplicación de Webscorer está conectado al router por WIFI.
- Los participantes son detectados por el lector RFID a su paso.

- El lector RFID envía automáticamente el identificador del participante a la aplicación de Webscorer.
- La aplicación de Webscorer puede comunicarse a través de Internet con los servidores de Webscorer, para permitir configurar varios puntos de control y el seguimiento del evento online.

Este sistema es compatible con lectores RFID de diversos fabricantes, que operan con el protocolo LLRP:

- Alien Technology.
- Honewell / Intermec.
- Impinj.
- Motorola / Zebra.
- ThingMagic.
- Radiant Sensors.

En conclusión, este sistema de cronometraje es una opción atractiva para realizar el cronometraje deportivo automático, ya que permite configurar varios puntos de control, además de permitir el seguimiento online. Por otro lado, tiene el problema de que los identificadores de los participantes deben ser introducidos manualmente en la aplicación, lo cual puede resultar un trabajo tedioso. Además, el coste total del sistema dependerá del coste de los dispositivos Hardware empleados.

### 2.3.2.3. Hardware de cronometraje compatible con LLRP

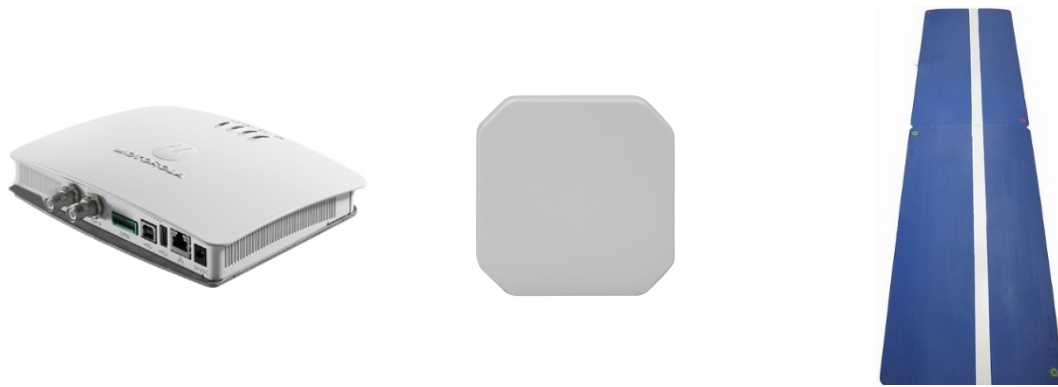
Gracias al protocolo LLRP es posible que un sistema de cronometraje deportivo se componga de unos dispositivos Hardware de un fabricante y un sistema Software de un desarrollador independiente, todo ello sin necesidad de ser adaptados para que funcionen conjuntamente.

El Hardware de cronometraje necesario en cada punto de control, compatible con LLRP, se compone de un lector RFID y una o varias antenas que son conectadas al lector. El lector cuenta con conexión WIFI o Ethernet, mediante el cual se conecta con un dispositivo en el que se encuentra instalado el Software de cronometraje, ya sea un dispositivo móvil o un ordenador. Existen varios fabricantes que proporcionan este tipo de lectores RFID y antenas, de entre los cuales se analizarán algunos de cara a conocer sus características y precios.

**Zebra FX7500 RFID Reader - 2 Port** El *Zebra FX7500* [at118c] es un lector RFID desarrollado específicamente para el cronometraje deportivo. Dispone de 2 puertos en los que es posible conectar antenas RFID, y conexión Ethernet que le permite conectarse a un router para operar bajo el protocolo LLRP. Tiene un precio de aproximadamente 1.200 €.

Características principales:

- Permite leer las etiquetas EPCglobal UHF Class 1 Gen 2 (ISO 18000-6C).
- Frecuencia de funcionamiento europeas: FCC (902-928 MHz).
- Nivel de aislamiento: IP53.
- Lecturas por segundo: 1.200.



(a) Lector RFID Zebra FX7500. (b) Antena RFID Laird S9025PL. (c) Antena RFID Times-7.

Figura 2.6: Dispositivos compatibles con LLRP.

**Laird S9025PL/S8655PL** La antena *Laird S9025PL/S8655PL* [atl18a] es una antena RFID de tipo panel que puede operar conectada a los lectores RFID analizados y puede ser empleada en exteriores. Tiene un precio aproximado de 130 €. Características principales:

- Frecuencias de operación: FCC (902-928 MHz), ETSI (865-868 MHz).
- Ganancia: 5 dBi.
- Polarización: Circular.
- Nivel de aislamiento: IP67.
- Dimensiones: 132 x 132 x 18 mm.
- Peso: 0.36 kg.

**Antena RFID Times-7** La antena RFID *Times-7* [atl18b] es una antena de tipo alfombra, que se ubica en el suelo de forma perpendicular a la dirección de avance de la carrera. Esta antena puede operar conectada a los lectores RFID analizados. Tiene un precio aproximado de 1.200 €. Características principales:

- Frecuencias de operación: FCC (902-928 MHz), ETSI (865-868 MHz).
- Ganancia: 10 dBi.
- Polarización: Lineal.
- Nivel de aislamiento: IP66.
- Dimensiones: 1220 x 600 x 12 mm.
- Peso: 8.4 kg.

## Discusión sobre el empleo de sistemas compatibles con LLRP

Estos sistemas de cronometraje son una solución de gran fiabilidad para llevar a cabo el cronometraje deportivo de eventos deportivos de baja participación. Sin embargo requieren disponer de un dispositivo móvil o un ordenador junto al lector para ejecutar el software de cronometraje. Tanto el ordenador o dispositivo móvil, como el lector RFID deberían ir situados en una caseta o similar que los proteja de las inclemencias meteorológicas .

Además, estos sistemas también tienen el problema del coste económico, ya que incluso en el caso más económico, en el que un evento deportivo tendría un único punto de control, las necesidades para realizar el cronometraje se compondrían de un lector, una antena, y una licencia de software, lo cual implica alrededor de 1.500 €.



## Capítulo 3

# Medios y tecnologías para desarrollar un sistema de cronometraje deportivo automático

El sistema de cronometraje a desarrollar se compone de dos tipos de módulos o subsistemas: **aplicación web**, y **equipos de cronometraje**. A su vez, estos módulos pueden tener componentes a nivel de *hardware*, como a nivel de *software*. Analizaremos las opciones para cada uno de los componentes del sistema de cronometraje, justificando la elección en cada caso.

### 3.1. Equipos de cronometraje

Los equipos de cronometraje son los módulos encargados de identificar a cada participante a su paso por los puntos de control intermedios, y en la meta. Estos equipos de cronometraje deben identificar a los participantes, asignar unas marcas de tiempo, y transferir la información al módulo de la aplicación web.

Para llevar a cabo este cometido, cada equipo de cronometraje estará compuesto por un dispositivo encargado de identificar a los participantes, y un dispositivo encargado de asignar un instante de tiempo de las identificaciones, así como comunicar dicha información a la aplicación web.

#### 3.1.1. Dispositivos para la identificación de participantes

En el Capítulo 2 de esta memoria se han analizado diferentes sistemas de cronometraje automáticos, que se apoyan en diferentes tecnologías para la identificación de los participantes. Estas tecnologías son NFC, y RFID en la banda UHF.

La tecnología NFC tiene el gran inconveniente de ser una tecnología de corto alcance, lo que, llevado a la práctica en el cronometraje deportivo, implica que los participantes deben detener su marcha para poder ser identificados, además, en caso de que varios participantes pasen al mismo tiempo se producirán colas y retrasos en la identificación.

La tecnología RFID en la banda UHF permite la identificación de los participantes a varios metros de distancia, además de permitir la identificación de varios participantes al mismo tiempo. Por tanto, para el sistema de cronometraje a desarrollar se emplea esta tecnología.

##### 3.1.1.1. Tecnología RFID

RFID es un método de identificación automático basado en el almacenamiento y recuperación remota de datos empleando etiquetas RFID. Una etiqueta RFID es un objeto pequeño que puede ser unido o incorporado a un producto, un animal o una persona. Una etiqueta RFID contiene una antena que

le permite recibir y responder a consultas enviadas por un lector RFID. Las etiquetas pasivas no necesitan disponer de alimentación propia, mientras que las etiquetas activas sí la necesitan.

La tecnología RFID permite la identificación de objetos en tiempo real, con un gran nivel de precisión. Esta tecnología se emplea en múltiples aplicaciones hoy en día, como logística, control de acceso, transporte, pasaportes, etc.

Las etiquetas RFID pasivas están formadas por un circuito integrado que contiene datos, y una antena. El circuito integrado es alimentado por el lector RFID cuando el lector se comunica con la etiqueta para obtener sus datos.

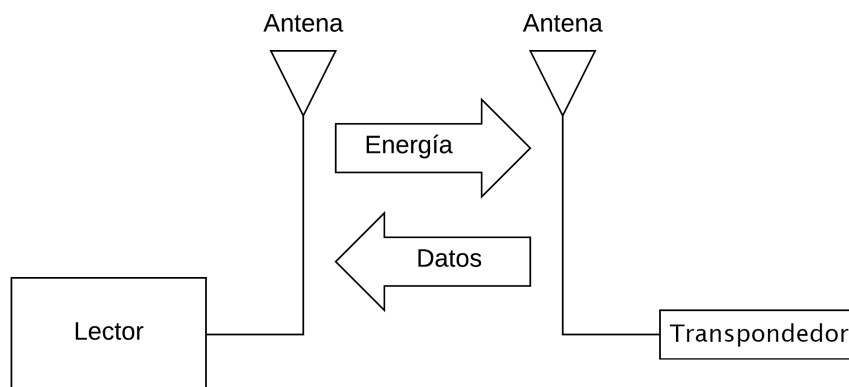


Figura 3.1: Esquema básico del funcionamiento de la tecnología RFID.

Las etiquetas que operan a bajas frecuencias (típicamente 135KHz o 13,56 MHz) funcionan únicamente a pequeña distancia, entre 0 y 1 metro. Sin embargo, las etiquetas que operan en frecuencias mayores, en la banda UHF (900 MHz o 2,45 GHz), pueden operar a mayores distancias. Esto es debido a que a bajas frecuencias la intensidad del campo magnético decae en mayor medida que a frecuencias altas.

**Etiquetas RFID** Las etiquetas RFID son dispositivos pequeños formados por una antena y un circuito integrado, pueden ser activas o pasivas.

La antena de la etiqueta puede tener formas diferentes dependiendo del tipo de etiqueta. Su trabajo es recibir las ondas de radiofrecuencia, alimentar el circuito integrado con su energía, y transmitir la información devuelta por el circuito integrado.

El circuito integrado puede ser de diferentes tipos. Se encarga de contener la memoria, procesar la información, enviar y recibir la información, y evitar la colisión con otras etiquetas.

Las etiquetas pasivas no tienen fuente de alimentación interna. Para alimentar el circuito integrado aprovechan la pequeña corriente eléctrica inducida en la antena por la señal de radiofrecuencia recibida. Esta pequeña corriente eléctrica es suficiente para que la etiqueta pueda enviar una respuesta. Debido a estas limitaciones de potencia disponible, las respuestas emitidas por las etiquetas pasivas son normalmente muy cortas, únicamente un número de identificación.

Las etiquetas activas en cambio, sí tienen fuente de alimentación interna. Al disponer de mayor potencia, estas etiquetas consiguen transmitir a mayores distancias, y además, transmitir mayor cantidad de información. Las etiquetas activas que encontramos en el mercado tienen un alcance de decenas de metros, y las baterías internas pueden durar varios años. El precio de las etiquetas activas



es mayor que el de las etiquetas pasivas. El tamaño de las etiquetas activas es mayor que el de las etiquetas pasivas, sin embargo, tampoco son demasiado grandes, aproximadamente el tamaño de una moneda.

En el sistema a desarrollar en este TFG necesitamos unas etiquetas que cumplan de la mejor manera las siguientes características:

- Permitir contener al menos un número de identificación, que identifique a cada participante.
- Alcance de al menos unos pocos metros (4-6 metros es el ancho típico de un circuito).
- Tener el menor precio posible. De este modo, si un participante no devuelve la etiqueta que le identifica en un evento deportivo, la pérdida económica no será grande.
- Ser lo más pequeña posible. En el caso de las etiquetas pasivas, tienen el grosor de un folio, con un alto y ancho de unos pocos centímetros. En el caso de las etiquetas activas, tienen aproximadamente el tamaño de una moneda.

El precio y el tamaño de las etiquetas pasivas las convierten en la mejor opción. Aunque tienen algunas debilidades respecto a las etiquetas activas, sus características son suficientes para cumplir con las necesidades del sistema.

**Estándares RFID** En la tecnología UHF RFID no hay un estándar aceptado universalmente, aunque hay algunas soluciones que son las más empleadas. Los estándares que más se están empleando son ISO18000 para lidiar con la comunicación, y EPC para la estructura de los identificadores de las etiquetas.

**Estándar EPC** El auto-ID center propone un estándar para el identificador de etiquetas RFID denominado EPC (*Electronic Product Code*), que permite identificar productos de forma similar a un código de barras. Su estructura está basada en GTIN (*Global Trade Item Number*). Existen cinco clases EPC según sus características:

| Clase EPC | Características                            | Tipo de transpondedor |
|-----------|--|-----------------------|
| Clase 0   | Solo lectura                               | Pasivo (solo 64 bits) |
| Clase 1   | Escritura una vez, lectura muchas          | Pasivo (mín. 96 bits) |
| Clase 2   | Lectura y escritura                        | Pasivo (mín. 96 bits) |
| Clase 3   | Lectura y escritura, con energía integrada | Pasivo (mín. 96 bits) |
| Clase 4   | Lectura y escritura, transmisor activo     | Pasivo (mín. 96 bits) |

Las clases EPC no son todas interoperables entre sí, aunque las etiquetas de las clases 0 y 1 sí pueden operar con lectores que soportan el estándar EPC de clase 2. Para solventar este problema de interoperabilidad, los lectores RFID suelen ser multiestándar.

**Estándar ISO18000** Este estándar está formado por un conjunto de especificaciones que definen los parámetros técnicos del enlace de comunicación, como la frecuencia de operación, ancho de banda, EIRP máximo, tipo de modulación, codificación de datos, tasa de transmisión, etc.

**Regulación UHF RFID para EPC Gen2** Actualmente, la tecnología RFID en la banda UHF se encuentra regulada, y podemos encontrar ligeras diferencias según el país en el que nos encontremos. Estas diferencias regulatorias afectan principalmente a la banda de frecuencias y a la potencia máxima de la antena.

España se encuentra dentro del área de frecuencias que se emplea en Europa, es decir, 865 - 868 MHz.

Si nos centramos en España, existe una resolución de la Secretaría de Estado de Telecomunicaciones y para la Sociedad de la Información [Est05], por la que se definen los requisitos técnicos de la interfaz radioeléctrica reglamentada IR-96 para los dispositivos de identificación por radiofrecuencia (RFID) en la banda de 865-868 MHz:

El artículo 10 del Reglamento que establece el procedimiento para la evaluación de conformidad de los aparatos de telecomunicaciones, aprobado por Real Decreto 1890/2000, de 20 de noviembre, en la redacción dada por el Real Decreto 424/2005, de 15 abril, establece que la Secretaría de Estado de Telecomunicaciones y para la Sociedad de la Información publicará como resolución en el «Boletín Oficial del Estado» las interfaces reglamentadas en España que hayan sido notificadas a la Comisión Europea.

La presente resolución tiene como objeto la publicación de los requisitos técnicos de la interfaz radioeléctrica para los dispositivos de identificación por radiofrecuencia en la banda de 865-868 MHz.

El cumplimiento de estos requisitos de interfaz es necesario para el uso de dispositivos de radiofrecuencia para aplicaciones de identificación en la banda de 865-868 MHz que operen en España y no exime del cumplimiento del resto de los requisitos establecidos en el Reglamento aprobado mediante el mencionado Real Decreto 1890/2000.

La concesión de títulos habilitantes para el uso del dominio público radioeléctrico, en su caso, está sujeta al cumplimiento de los requisitos mínimos especificados en las interfaces radioeléctricas correspondientes a los tipos de equipos utilizados.

Esta interfaz podrá ser revisada, de acuerdo con lo establecido en el Cuadro Nacional de Atribución de Frecuencias en vigor, como consecuencia de nuevas necesidades nacionales en cuanto al uso eficaz del espectro radioeléctrico o bien como consecuencia de tratados o acuerdos internacionales en los que España sea parte.

Esta disposición ha sido sometida al procedimiento de información en materia de normas y reglamentaciones técnicas y de reglamentos relativos a los servicios de la sociedad de la información, previsto en la Directiva 98/34/CE del Parlamento Europeo y del Consejo de 22 de junio, modificada por la Directiva 98/48/CE de 20 de julio, así como en el Real Decreto 1337/1999, de 31 de julio que incorpora estas Directivas al ordenamiento jurídico español.

En su virtud, resuelvo publicar los requisitos de la interfaz reglamentada que se relacionan en el anexo a esta Resolución.

Madrid, 25 de junio de 2008.-El Secretario de Estado de Telecomunicaciones y para la Sociedad de la Información, P. D. (Orden ITC/3187/2004, de 4 de octubre, modificada por la Orden ITC/3359/2005, de 20 de octubre), el Subdirector General de Infraestructuras y Normativa Técnica, Ricardo Alvarino Álvarez.

|   |  |
|---|--|
| <b>Frecuencia/Banda de frecuencias.</b>             | a) 865-865,6 MHz. b) 865,6-867,6 MHz. c) 867,6-868 MHz.                              |
| <b>Canalización/Anchura de banda.</b>               | 200 kHz.   |
| <b>Modulación.</b>                                  | No se especifica.  |
| <b>Separación dúplex.</b>                           | No se aplica.  |
| <b>Nivel de potencia.</b>                           | En banda a) 100 mW (p.r.a.). En banda b) 2 W (p.r.a.). En banda c) 500 mW (p.r.a.).  |
| <b>Servicio radioeléctrico/tipo de dispositivo.</b> | Servicio móvil terrestre. Dispositivos de identificación por radiofrecuencia (RFID). |
| <b>Licencia/uso.</b>                                | Sin licencia. Uso común.   |
| <b>Evaluación/notificación.</b>                     | Clase II.  |
| <b>Norma técnica de referencia.</b>                 | ETSI EN 302 208. CEPT ERC Rec.70-03. Decisión 2006/804/CE.                           |
| <b>Otras observaciones.</b>                         | Cuadro Nacional de Atribución de Frecuencia: UN-135.                                 |

Dispositivos de identificación por radiofrecuencia (RFID) en la banda de 865-868 MHz.

### 3.1.1.2. Dispositivos RFID elegidos para la identificación de participantes en este proyecto

A la hora de elegir el dispositivo o dispositivos RFID encargados de la identificación de los participantes, contamos con múltiples opciones. Dado que el cronometraje de eventos deportivos basado en RFID no es algo novedoso, podemos encontrar proveedores que ofrecen lectores RFID de todo tipo, características y precios.

Hay que tener en cuenta que podemos encontrar proveedores en todas las partes del mundo. En un primer análisis, se observa que productos de características técnicas similares tienen un precio mucho más bajo en proveedores chinos, que en proveedores europeos y americanos. Los precios en los proveedores chinos son entre un tercio y la mitad que en los proveedores occidentales. Por esta razón se opta por adquirir los dispositivos RFID a un proveedor chino.

La primera elección que se debe tomar a la hora de elegir el lector RFID, es el tipo de antena. En el mundo del cronometraje deportivo podemos encontrar dos tipos de antenas: Antenas de alfombra, y antenas de panel.

Las antenas de alfombra van ubicadas en el suelo, de forma perpendicular a la dirección del recorrido. Estas antenas suelen ir recubiertas de un protector que se caracteriza físicamente por su aspecto de badén. Estas antenas son muy habituales en eventos de gran número de participantes, y su precio es bastante elevado.

Las antenas de panel tienen forma cuadrangular, cuyo lado habitualmente tiene entre 20 y 50 cm. Estas antenas se suelen ubicar en los lados del recorrido, ancladas a un trípode o mástil, aunque también es posible encontrarlas ancladas en la parte superior de una estructura que esté ubicada sobre el recorrido. Estas antenas son más económicas que las antenas de alfombra.

La segunda elección es el tipo de dispositivo RFID. Hasta ahora estábamos dando por hecho que el lector RFID se compone tanto de la antena, como del lector. Sin embargo, se pueden encontrar lectores RFID que efectivamente llevan incorporado en una misma pieza la antena y el lector RFID, mientras que también existen lectores que no llevan antena incorporada, sino una o varias entradas para conectar antenas independientes.

Los lectores RFID con antena incorporada en la misma pieza son más económicos que comprar lectores y antenas de forma independiente. Sin embargo, tienen la capacidad de leer menor número de etiquetas por segundo.

Los lectores RFID sin antena incorporada en la misma pieza suelen ser más caros. Normalmente permiten conectar varias antenas, lo que permite mejores áreas de actuación para realizar el cronometraje. Estos lectores permiten mayor número de etiquetas por segundo y son más caros.

Dado que en este TFG se busca obtener un sistema de cronometraje destinado a eventos deportivos de baja participación y económico, se opta por emplear un lector con antena incorporada de tipo panel.

Dentro de los lectores RFID con antena de tipo panel, existen varias opciones con diferentes características. Las características más importantes que diferencian unos de otros son: Directividad, polarización, alcance, grado de protección para exteriores, y frecuencia de funcionamiento.

**Directividad** Existen antenas omnidireccionales, y antenas directivas. Para los equipos de cronometraje del sistema necesitamos antenas direccionales, lo más directivas posible.

**Polarización** Encontramos antenas RFID que tienen polarización lineal o circular. Con polarización circular se pueden leer las etiquetas que se encuentran en cualquier ángulo respecto a la antena, mientras que con polarización lineal únicamente se leerán las etiquetas que se encuentren en el ángulo requerido. En el sistema que vamos a desarrollar, las etiquetas irán acopladas a los dorsales de los participantes, y éstos irán en una posición fija, por lo que la lectura de la etiqueta está garantizada para cualquier polarización de la antena. Por otra parte, las antenas que utilizan polarización circular requieren mayor potencia que las antenas con polarización lineal para lograr el mismo alcance, por tanto, vamos a emplear una antena con polarización lineal.

**Alcance de la antena** El alcance es el rango de distancias en las que una etiqueta RFID puede ser detectada por el lector. Las antenas que tienen mayor alcance, tienen unas dimensiones mayores y son ligeramente más caras. Para los equipos de cronometraje necesitamos un alcance bastante grande, ya que podemos considerar que el recorrido tenga una anchura de unos 4-6 metros, y además el trípode con la antena se ubicará dejando 1 metro libre desde el borde del recorrido por motivos de seguridad. Por tanto, necesitamos un alcance de al menos 7 metros reales. Las dimensiones de la antena no suponen un problema ya que irá montada sobre un trípode y tendremos espacio suficiente para ella.

**Grado de protección del lector RFID** El grado de protección permite conocer si el dispositivo será apto para trabajar en exteriores. Existe la norma internacional IP que clasifica los diferentes grados de protección de equipos electrónicos, que vienen dados por los contenedores que envuelven y resguardan a los mismos. El lector a emplear debe poder trabajar en exteriores sin problemas, por tanto, evitar que el polvo entre en el dispositivo, y garantizar que no va a entrar agua. Este nivel de protección se especifica con la norma IP67.

**Frecuencia de funcionamiento** Debe poder operar en el rango de frecuencias establecidas por la unión europea para RFID UHF.

Para el equipo de cronometraje que vamos a emplear, se ha optado por el lector RFID **INNOD IND8001**. Este lector lleva antena incorporada, es muy económico, y tiene las siguientes características técnicas:

- 1-15 metros de alcance.
- Polarización lineal.
- Hasta 12 DBi de ganancia, configurable.
- Conexión RS232.
- Frecuencia de operación configurable (permite las frecuencias europeas).
- Estándar RFID: ISO/IEC 18000-6c EPC Class1 Gen2.
- Capacidad de lectura: 50 etiquetas/segundo.
- Dimensiones: 450x450x75 mm.
- Peso: 2.4 Kg.

### 3.1.2. Dispositivos para la asignación de tiempos y comunicación con la aplicación web

El lector RFID obtiene los identificadores de las etiquetas RFID que lleva cada participante, y los transmite por su interfaz de conexión, en el caso del lector elegido, RS232.



Figura 3.2: Lector/Antena RFID INNOD IND8001.

Para poder tratar esta información, es necesario disponer de un dispositivo conectado que se encargue de recibir y procesar la información obtenida por el lector RFID, asignar tiempos, y transmitir la información al módulo que se encarga de gestionar el sistema global. Este dispositivo podría ser un smartphone, un ordenador o un miniordenador.

El dispositivo debe cumplir las siguientes características:

- No requerir la presencia de una persona que deba configurar y portar el dispositivo.
- Tamaño pequeño, para que permita ser acoplado junto al lector RFID.
- Bajo coste.
- Conexión Ethernet, para poder comunicarse con el módulo central.
- Conexión RS232/UART, para la conexión con el lector RFID.
- Bajo consumo.
- Capacidad computacional suficiente para lidiar con las necesidades de la aplicación.

El dispositivo que mejor se adapta a estos requisitos es un miniordenador. De entre los miniordenadores disponibles en el mercado, se opta por una *Raspberry Pi*, debido a que cumple todos los requisitos necesarios, y entre los miniordenadores que cumplen los requisitos tiene el menor precio.

#### 3.1.2.1. Raspberry Pi

*Raspberry Pi* [Fou18d] es un miniordenador, un computador de placa única o computador de placa simple (SBC) de precio bajo desarrollado por la *Fundación Raspberry Pi*, que se lanzó en 2006 con el objetivo de acercar y fomentar la enseñanza de electrónica e informática en las escuelas.

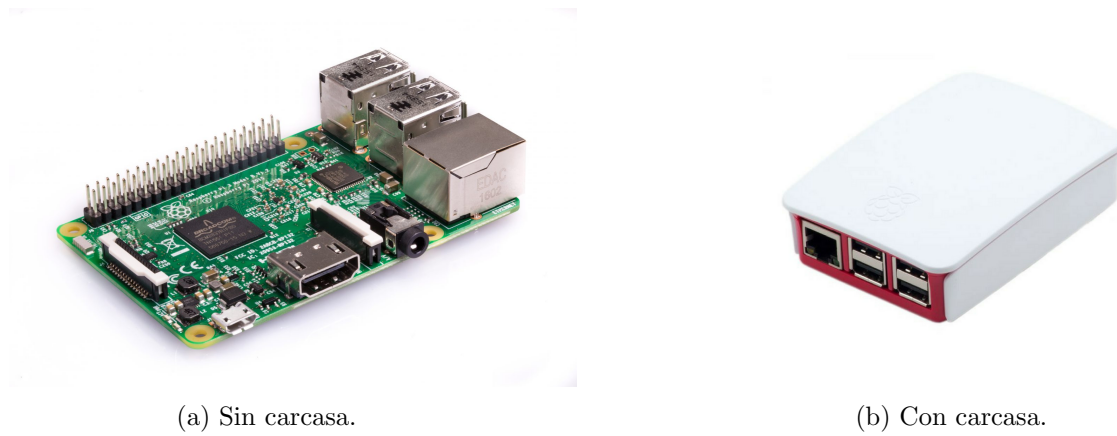


Figura 3.3: Raspberry Pi 3 Model B.

Emplea un sistema operativo denominado *Raspbian* [Fou18e] , que es una versión adaptada de *Debian*. También permite usar otros sistemas operativos, incluido una versión de *Windows 10* orientada al desarrollo de proyectos de Internet de las cosas. En todas sus versiones incluye un procesador Broadcom, una memoria RAM, una GPU, puertos USB, HDMI, Ethernet (excepto el primer modelo), 40 pines GPIO y un conector para cámara. No incluye memoria interna, si no una ranura para tarjetas SD.

La fundación de la Raspberry promueve el aprendizaje del lenguaje de programación Python, por ello proporciona un conjunto de bibliotecas en este lenguaje para operar con los pines y conexiones de la Raspberry. Aunque también es posible desarrollar aplicaciones en otros lenguajes de programación, el ecosistema de la Raspberry proporciona mayores facilidades para trabajar con Python.

Para los equipos de cronometraje que vamos a emplear en este proyecto, es válido tanto utilizar Una *Raspberry Pi 2*, como una *Raspberry Pi 3*.

### 3.1.2.2. Cableado y adaptadores

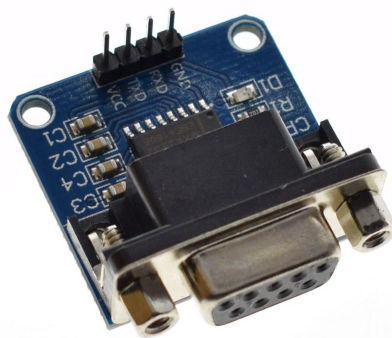


Figura 3.4: Max3232 RS232 Port To TTL Converter.

Para la conexión de la *Raspberry Pi* con el lector RFID, es necesario una serie de cables y adaptadores. Esto es debido a que la conexión serie del lector RFID emplea unos valores de voltaje diferentes a los empleados por la conexión *UART* de la *Raspberry Pi*.

El adaptador elegido es un módulo *Max3232 RS232 Port To TTL Converter*, basado en el circuito integrado *MAX3232*. Este adaptador consigue adaptar los niveles de voltaje del lector a los de la

Raspberry. Cuenta por una lado con un puerto RS232 serie donde conectar el lector, y por otro unos pines donde conectar la Raspberry.

### 3.1.3. Lenguaje de programación y bibliotecas para el desarrollo de la aplicación en los equipos de cronometraje

Para el desarrollo de la aplicación que se ejecutará en la *Raspberry Pi* se empleará *Python*. La principal razón para emplear python es la facilidad de trabajo de este lenguaje con los pines e interfaces de la *Raspberry Pi*.

*Python* es un lenguaje de programación interpretado cuya principal característica es la legibilidad de su código. Este lenguaje soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado multiplataforma. Es administrado por la *Python Software Foundation* y su licencia es de código abierto.

De cara a la programación de la aplicación con *Python* que se encarga de la comunicación entre la *Raspberry Pi* y el lector RFID, se emplea la biblioteca *pySerial*. *pySerial* [Lie17] es un módulo *Python* que encapsula el acceso a puertos serie. Este módulo puede funcionar con cualquier sistema operativo en el que trabaje *Python*. Proporciona métodos para la apertura de conexiones en serie permitiendo definir los parámetros necesarios, cierre de conexiones, escritura y lectura de datos. El empleo de esta librería permite el desarrollo de la aplicación que se ejecuta en la *Raspberry Pi* de forma rápida y efectiva.

## 3.2. Aplicación web

La aplicación a desarrollar debe ser realizada con tecnologías *web*, con una interfaz que permita ser accesible desde cualquier navegador. Esta aplicación *web* deberá proporcionar funcionalidades para la gestión de los eventos deportivos, además de permitir el seguimiento en directo. Por tanto, deberá disponer de una base de datos para la gestión de toda la información del sistema.

### 3.2.1. Software para el desarrollo de la aplicación web

A la hora de desarrollar la aplicación *web*, debemos elegir el conjunto de tecnologías a emplear basándonos en sus características. A la hora de tomar las decisiones, debemos diferenciar entre tecnologías que se encuentran del lado del cliente (navegador *web*) y que interactúan con el usuario, también llamadas *Frontend*. Por otro lado se encuentran las tecnologías que se encuentran en el lado del servidor, y que interactúan con la base de datos, denominadas *Backend*.

#### 3.2.1.1. Tecnologías Backend

El *Backend* contiene la lógica de la aplicación, y se encarga del acceso y gestión de los datos. El *Backend* no es accesible directamente por los usuarios, sino que se emplea el *Frontend* para ello.

Para el desarrollo del *Backend* de la aplicación *web* necesitamos emplear un lenguaje de programación, con la opción de utilizar también un *framework*. Además, se debe emplear una base de datos donde se almacenará la información de nuestro sistema.

El empleo de un *framework* proporciona un conjunto de módulos y funcionalidades que son comunes en el desarrollo de aplicaciones *web*, permitiendo el desarrollo ágil. Gracias a su uso, podemos centrarnos en desarrollar las características propias de nuestro sistema, sin reparar en algunos detalles que son comunes en la mayoría aplicación *web*. Por estas razones, se ha decidido emplear un *framework* para el desarrollo *Backend* de la aplicación web.

A la hora de escoger el lenguaje de programación y el *framework* adecuados para desarrollar el sistema, disponemos de múltiples opciones. Existen numerosos frameworks modernos, de diferentes lenguajes de programación, que permiten el desarrollo rápido de aplicaciones *web* y que pueden servir perfectamente para el desarrollo de la aplicación *web* del sistema de cronometraje. Algunos ejemplos de frameworks que podrían servir son *Django*, *Ruby on Rails*, *Symfony*, *Laravel*, etc.

Para el desarrollo de este TFG se empleará *Django* [Fou18a]. Muchas son las ventajas que aporta desarrollar una aplicación *web* con *Django*, como se analiza a continuación. Sin embargo, la razón principal de la elección de *Django* es el conocimiento previo del autor de este TFG en el desarrollo con framework, lo cual permite emplear un número mucho menor de horas para llevar a cabo el trabajo.

### 3.2.1.2. Django

*Django* es un framework para el desarrollo web de código abierto, escrito en el lenguaje de programación *Python*. Este framework está enfocado en el desarrollo rápido de aplicaciones, y pone mucho énfasis en la re-usabilidad de componentes.

Es un framework moderno, lanzado en el año 2003, y cuya última versión ha sido liberada en 2017. Junto con *Ruby on Rails*, es uno de los frameworks con mayor popularidad en los últimos años. Este framework ha sido utilizado para desarrollar sitios de gran fama mundial como *Instagram* o *Pinterest*.

Este es el framework elegido para el desarrollo de nuestra aplicación *web*, y por tanto vamos a analizar más en profundidad las características más importantes de este framework tan popular.

La característica que mejor define a *Django* es su modularidad. Las funciones principales de *Django* están desvinculadas unas de otras, lo que permite a los desarrolladores seleccionar y elegir qué piezas son adecuadas para su proyecto y cuáles no. Gracias a esta característica, se puede obtener un código más optimizado y con mayor rendimiento.

Un punto a favor de *Django* frente a otros frameworks, es la transparencia con la que se ejecutan sus funcionalidades. En algunos frameworks, como es el caso de *Ruby on Rails*, gran parte de sus funcionalidades ocurren de forma oculta al desarrollador, de forma que en ocasiones puede parecer que no se tiene el control a la hora de desarrollar una aplicación *web*. La transparencia de *Django* permite definir de forma explícita cada una de las funcionalidades y propiedades que se emplean en cada momento.

*Django* sigue el patrón Modelo-Vista-Plantilla. Este patrón es muy similar a el patrón clásico Modelo-Vista-Controlador, pero con unas pequeñas diferencias y adaptado al desarrollo web. Gracias a este patrón, *Django* permite estructurar una aplicación web de forma eficiente:

**Modelo** Los modelos en *Django*, siguen la misma filosofía que los modelos del patrón MVC clásico. Los modelos definen la estructura de los datos almacenados en la base de datos, incluidos los tipos de campo, tamaño máximo, valores predeterminados, opciones, etc. En *Django*, la definición del modelo es independiente de la base de datos empleada, y es el propio framework el encargado de comunicarse con la base de datos.

**Plantilla** Las plantillas se realizan con HTML, siguiendo los estándares web. Estas plantillas se encargan únicamente de presentar los datos y contenidos configurados en las Vistas. *Django* cuenta con un poderoso sistema integrado de plantillas basadas en el sistema de herencia, que permiten generar HTML dinámicamente y, al mismo tiempo eliminar redundancias. Gracias a el empleo de estas plantillas, personas que se encargan de la parte de diseño y sin conocimientos de programación en *Python*, pueden encargarse del diseño y mantenimiento de la apariencia de una aplicación web desarrollada con *Django*.



**Vista** Las vistas son funciones o clases que son llamadas cuando se accede a una URL. Las vistas se encargan de obtener y gestionar los contenidos que se van a mostrar accediendo a los modelos de datos, pero no se encargan de como se muestran esos datos. Las Vistas tienen algunas similitudes con los controladores del patrón clásico MVC, sin embargo, algunas de las funcionalidades que suelen recaer del lado de un controlador, en *Django* es el propio framework el que se encarga sin necesidad de programar nada.

Estas son algunas de las características que hacen a *Django* el framework adecuado para el desarrollo rápido de aplicaciones web:

- Una *API* potente y fácil de utilizar. *Django* tiene automatizadas las tareas necesarias para comenzar una aplicación web, todo ello ejecutando unos comandos simples sin necesidad de programación adicional.
- Abstracción de la base de datos. *Django* se encarga de lidiar con la base de datos, proporcionando una serie de utilidades que permiten realizar multitud de operaciones con la base de datos únicamente mediante código en *python*, sin necesidad de programar en el lenguaje del sistema gestor de la base de datos. Además, permite trabajar con diferentes sistemas gestores de bases de datos, e incluso cambiar la aplicación web de una base de datos sin necesidad de modificar el código de la aplicación.
- Un generador de panel de administración dinámico. *Django* proporciona un panel de administración de forma automática, con sólo unas pequeñas configuraciones. Este panel autogenerado permite gestionar los datos de los modelos que se han programado de una forma simple y efectiva. Aunque este panel no es siempre la mejor opción de cara a la aplicación final debido a que es bastante simple, siempre resulta de utilidad a la hora de desarrollar la aplicación web para introducir manualmente información que permita realizar pruebas.
- Manejador de URLs. *Django* cuenta con un manejador de URLs que permite definir y utilizar de forma clara y elegante las URLs de una aplicación web. En *django*, cada URL se define en un fichero de URLs, donde se especifica la ruta, la vista que será llamada al acceder a la URL, y un nombre. Gracias al nombre asignado a las URLs, éstas pueden ser usadas en las plantillas de forma clara y sencilla.
- *Django* tiene incorporado un servidor de desarrollo ligero y fácil de utilizar, que permite testear la aplicación mientras se realiza el desarrollo. Este servidor es muy útil a la hora de desarrollar la aplicación web, debido a que se lanza con sólo un comando, y se reinicia cada vez que se realizan cambios en el código. Por otra parte, no es un servidor que sirva para emplearse en producción.
- *Shell de Django*. El *Shell* es un intérprete interactivo de *python*, que nos permitirá probar los modelos de la aplicación, hacer consultas y analizar resultados. Es muy útil a la hora de desarrollar la aplicación, ya que te permite interactuar con los modelos de datos, sus atributos y sus métodos, desde una terminal o ventana de comandos.
- Características de seguridad. *Django* tiene herramientas para proteger contra ataques *XSS*, ataques *CSRF*, inyecciones *SQL*, *clickjacking* y otros ataques. Además, permite habilitar fácilmente *SSL/HTTPS*.
- *Middlewares*. Estos componentes de *Django* permiten ejecutar o realizar acciones específicas antes de que una petición llegue a una vista, o después de que una vista envíe una respuesta. Cada componente *middleware* se encarga de realizar una acción específica, aquella para la que ha sido diseñado. *Django* incorpora algunos *middlewares* en el propio *framework*, los cuales pueden ser usados, o no serlo, dependiendo de las necesidades de cada aplicación web. Además, es posible desarrollar *middlewares* adicionales e incorporarlos a la aplicación. El *middleware* más destacado es el encargado de la autenticación y autorización de usuarios.

### 3.2.1.3. Django REST framework

Para la transferencia de información entre los equipos de cronometraje y la aplicación web es necesario emplear REST (*Representational State Transfer*). Además, para el formato de la información se empleará JSON (*JavaScript Object Notation*).

Una de las características más importante de *Django* es su modularidad. Gracias a ello, es posible incorporar módulos externos que proporcionan características que no incorpora el propio framework.

Para el desarrollo del proyecto, es necesario disponer de una *API REST* que se encargue de recibir y suministrar datos a los Equipos de Cronometraje.

Por ello, se emplea esta herramienta, *Django REST framework* [Chr18], que permite construir una *API REST* sobre *Django*.

### 3.2.1.4. Base de datos

Para almacenar y acceder a la información de nuestra aplicación web, necesitamos emplear una base de datos. A la hora de decidir qué base de datos utilizar, debemos analizar el tipo de información con la que va a trabajar nuestro sistema.

En la aplicación web que vamos a desarrollar, se trabajarán con un conjunto de datos de Eventos Deportivos, Recorridos, Participantes, etc., todo ello son datos que debe ser almacenados de forma estructurada y organizada. Por ello, es necesario emplear una base de datos relacional.

Para determinar qué base de datos y sistema gestor de base de datos utilizar, es necesario tener en cuenta las características de *Django*. *Django* lleva incorporado una serie de funcionalidades para lidiar con las bases de datos de forma fácil, permitiendo al desarrollador trabajar con la información contenida en la base de datos, sin preocuparse por la complejidad del sistema utilizado. *Django* permite trabajar con estos sistemas gestores de bases de datos: *PostgreSQL*, *MySQL*, *SQLite*, *Oracle*. Por tanto, la elección debe estar entre estos 4 sistemas.

*Oracle* es un sistema de pago, por lo tanto queda descartado. Para el desarrollo de la aplicación web, nos sirve cualquiera de las otras tres opciones (*MySQL*, *PostgreSQL* y *SQLite*). Nos decantamos por *SQLite* por la simplicidad que supone emplear una base de datos de ficheros.

*SQLite* [Hip18] es un sistema de gestión de bases de datos relacional, de código abierto. Es un sistema rápido y eficiente para el manejo de información. Las bases de datos *SQLite* se almacenan en ficheros, permitiendo acceder a la información directamente accediendo al fichero de la base de datos, y evitando tener que realizar la comunicación a través de puertos o sockets. Esta característica también permite realizar exportaciones e importaciones con solo copiar y pegar el fichero.

### 3.2.1.5. Tecnologías Frontend

El *Frontend* es la parte de la aplicación web que se ejecuta en el navegador web del cliente. Lo conforman un conjunto de tecnologías que se encargan únicamente de la presentación, típicamente se emplean *HTML*, *CSS* y *Javascript*.

### 3.2.1.6. HTML

HTML (*HyperText Markup Language*) es un lenguaje de marcado para la elaboración de páginas web. HTML es el lenguaje estándar para la web, que es interpretado por el navegador para su visualización. HTML es un lenguaje formado por etiquetas que permiten definir la estructura y contenidos de un sitio web.

### 3.2.1.7. CSS

CSS (*Cascading Stylesheets*) es un lenguaje de diseño gráfico empleado para definir como se presentan las páginas web a los usuarios. CSS se emplea en combinación con HTML, para proporcionar mejores características a la hora de diseñar una web. Aunque HTML cuenta con algunos atributos que permiten definir el aspecto de los contenidos, estos atributos tienen grandes carencias y limitaciones.

Gracias al empleo de CSS en combinación con HTML, se puede definir la estructura de una web por un lado con HTML, mientras que por otro lado se puede definir cómo se mostrarán esos contenidos con CSS.

### 3.2.1.8. Javascript

*Javascript* es un lenguaje de programación interpretado que se ejecuta típicamente en un navegador web. El uso de *Javascript* permite mejorar la experiencia de usuario a la hora de navegar por la web. Todos los navegadores modernos permiten ejecutar *Javascript*, por lo que su uso está muy extendido.

Aunque el uso de *Javascript* inicialmente se limitaba a ser ejecutado para realizar operaciones únicamente del lado del cliente, actualmente se emplea también para realizar operaciones de envío y recepción de información al servidor con ayuda de otras tecnologías como *AJAX*.

### 3.2.1.9. AJAX

*AJAX* (*Asynchronous JavaScript And XML*) es una técnica empleada en la elaboración de sitios web para mantener una conexión asíncrona con el servidor. Consiste en ejecutar código *Javascript* que gestiona peticiones y respuestas al servidor, mejorando la interactividad y usabilidad de los sitios web.

### 3.2.1.10. jQuery

*jQuery* [Fou18b] es una biblioteca de *JavaScript* ligera y con una gran variedad de funciones disponibles. Proporciona herramientas para programar con facilidad cosas como la manipulación de elementos HTML, manejar eventos, animaciones, y *AJAX* y muchas otras funcionalidades. Para ello proporciona una API fácil de usar que además es compatible con la mayoría de navegadores modernos. Desde su aparición, *jQuery* ha cambiado la forma en que millones de personas desarrollan aplicaciones web con *JavaScript*.

### 3.2.1.11. Bootstrap

*Bootstrap* [OTc18] es un conjunto de herramientas de código abierto para desarrollar sitios web empleando HTML, CSS y *JavaScript*, además de *jQuery*. *Bootstrap* fue desarrollado y liberado por *Twitter*, y tiene por objetivo facilitar y acelerar el diseño web. Permite crear de forma sencilla webs de diseño que se adapta a cualquier dispositivo y tamaño de pantalla.



## Capítulo 4

# Desarrollo de los equipos de cronometraje

### 4.1. Descripción

Los equipos de cronometraje son los módulos del sistema encargados de identificar a los participantes a su paso por los puntos de control intermedios y en la meta. Cada uno de estos equipos está compuesto por un lector/antena RFID y una *Raspberry Pi*, además de conectores y adaptadores. Para poder comunicarse con el subsistema central es necesario que el equipo disponga de conexión a internet.

La configuración a realizar en un equipo de cronometraje para ser empleado en el cronometraje de un evento deportivo es prácticamente nula, simplemente es necesario conectar el equipo a internet y la alimentación eléctrica. Todas las configuraciones de estos equipos se reducen a la puesta en marcha, es decir, el primer uso de los mismos.

El método de funcionamiento para lograr identificar a cada participante es el siguiente: cada vez que un participante pasa por delante de un equipo de cronometraje, la etiqueta RFID que lleva acoplada al dorsal es detectada por el lector/antena RFID y el lector transmite esta información a la *Raspberry Pi*, la *Raspberry Pi* obtiene el identificador de la etiqueta RFID (denominado EPC) y le asigna una marca de tiempo. Una vez que la *Raspberry Pi* ha asociado una marca de tiempo a una detección, transmite esta información al subsistema central (la aplicación *web*). La combinación del EPC de una etiqueta RFID con una marca de tiempo que son enviados desde un equipo de cronometraje a la aplicación *web* se denominan *envíos de cronometraje*.

Sin embargo, los cometidos de un equipo de cronometraje no se reducen únicamente a la identificación de los participantes, sino que también debe encargarse de comunicar el estado del equipo y el funcionamiento del lector/antena a la aplicación *web*. Gracias a los envíos de información de estado, desde la aplicación *web* se puede conocer en todo momento si existen problemas en los equipos de cronometraje que se encuentran registrados en el sistema, para ello se definen tres posibles estados:

**Operativo** El equipo de cronometraje está conectado y funciona correctamente.

**Conectado, no Operativo** El equipo de cronometraje está conectado al sistema, pero se detecta un problema con el lector/antena RFID. Es decir, no puede detectar participantes.

**Desconectado** El equipo de cronometraje no está enviando información de estado a la aplicación *web*, por lo cual se deduce que no dispone de conexión a Internet o no se encuentra con alimentación eléctrica. Por supuesto, no puede transmitir información de detecciones de participantes.



Figura 4.1: Esquema de funcionamiento y componentes de un equipo de cronometraje

Es necesario tener en cuenta la posibilidad de que un equipo de cronometraje pierda momentáneamente la conexión a internet mientras está siendo empleado para cronometrar un evento deportivo. Por esta razón, los equipos de cronometraje almacenan en local los *envíos de cronometraje* que no ha sido posible transmitir, para intentar su envío de nuevo posteriormente.

## 4.2. Investigaciones previas

Gran parte del trabajo realizado para desarrollar los equipos de cronometraje ha consistido en establecer una comunicación correcta entre la *Raspberry Pi* y el lector/antena RFID. Para ello han sido necesarias una serie de investigaciones y pruebas de dificultad gradual.

En el transcurso de estas pruebas, en ocasiones se ha empleado una *Raspberry Pi 2*, y en ocasiones una *Raspberry Pi 3*. De cara a la implementación de los equipos de cronometraje es válido cualquiera de los dos modelos.

### 4.2.1. Características de la *Raspberry Pi* para la comunicación

El dispositivo cuenta con conexión UART (*Universal Asynchronous Receiver-Transmitter*) [Fou18g], a través de 2 puertos denominados *TXD* y *RXD* en forma de pines. Estos pines forman parte de los GPIO (*General Purpose Input/Output*) de la *Raspberry Pi*, todos ellos con unos niveles lógicos de voltaje hasta 3,3 V. En estos pines, el estado *LOW* se corresponde con un voltaje de entre 0 V y 0,8 V, mientras que el estado *HIGH* se corresponde con un voltaje entre 2 V y 3,3 V. Por lo tanto, no se deben superar los 3,3 V ya que podría suponer daños para el dispositivo.

En cuanto a los parámetros opcionales de la conexión serie, el dispositivo soporta diferentes configuraciones en cuanto a velocidad de transmisión, paridad, etc. Por tanto, se puede adaptar el dispositivo a las necesidades de la antena en el código de la aplicación.

#### 4.2.2. Características del lector UHF RFID para la comunicación

El lector RFID *INNOD IND8001* cuenta con varios modos de operación, pudiendo elegir dependiendo de las necesidades de cada caso. Este lector cuenta con una conexión serie, que permite configurar el lector y obtener las etiquetas RFID que va detectando cuando está en funcionamiento. El fabricante ha proporcionado documentación donde se explica entre otras cosas los modos de funcionamiento, comandos y formato de los paquetes para la comunicación con el lector.

El lector cuenta con un *buffer* donde puede almacenar las etiquetas que va detectando, sin embargo, las necesidades del proyecto no requieren que se almacenen las etiquetas en un *buffer*, si no que transmitan las mismas en tiempo real a la *Raspberry Pi*.

Por tanto, se debe emplear el modo de operación denominado *cmd-real-time-inventory*. Este modo hace que el lector de la antena no almacene en su *buffer* las etiquetas RFID detectadas, sino que las transmita en tiempo real. En este modo de operación se envía al lector un comando *cmd-real-time-inventory*, entonces el lector detecta las etiquetas que hay dentro de su alcance y transmite un comando de respuesta por cada etiqueta que detecta.

Los pines de la conexión en serie del lector tienen un voltaje de  $\pm 9$  V, esto significa que no es compatible con los niveles de voltaje de la *Raspberry Pi*. Por ello es necesario emplear un adaptador.

En cuanto a los parámetros posibles de la conexión serie, el dispositivo soporta poco margen de configuraciones, únicamente permitiendo entre 2 valores para la velocidad de transmisión, y siendo inalterables el resto de parámetros. La velocidad de transmisión que empleamos en este proyecto es la más alta que permite, y también su valor por defecto. Estos son todos los parámetros de la conexión serie empleados:

- Velocidad de transmisión (*boud rate*): 115200
- Bits de datos: 8
- Paridad: sin paridad
- Bits de parada: 1

#### 4.2.3. Pruebas realizadas

Se han realizado pruebas de dificultad incremental que han posibilitado una comunicación correcta entre el lector RFID y la *Raspberry Pi*. Estas pruebas se han dividido en varios pasos.

##### 4.2.3.1. Paso 1: Pruebas sobre el funcionamiento del lector UHF RFID

El fabricante del lector/antena RFID ha proporcionado información muy útil sobre el funcionamiento de la antena y del lector integrado, destacando los formatos de los paquetes y modos de operación del lector.

Sin embargo, para asegurarnos de su funcionamiento se han realizado unas pruebas con un *Software Demo* para *Windows* que ha proporcionado el fabricante. Este *Software Demo* permite operar con el lector, estableciendo parámetros de configuración y trabajar con los diferentes modos de operación. Además, permite visualizar los datos enviados y recibidos en formato hexadecimal.

Las Figuras 4.2 y 4.3 corresponden a capturas de pantalla del *Software Demo* mostrando la información en diferentes formatos, lo cual ayuda a comprender el funcionamiento del lector RFID. Mientras en la Figura 4.2 se muestra la información decodificada y en un formato amigable, en la Figura 4.3 se muestran los paquetes enviados y recibidos en formato hexadecimal.

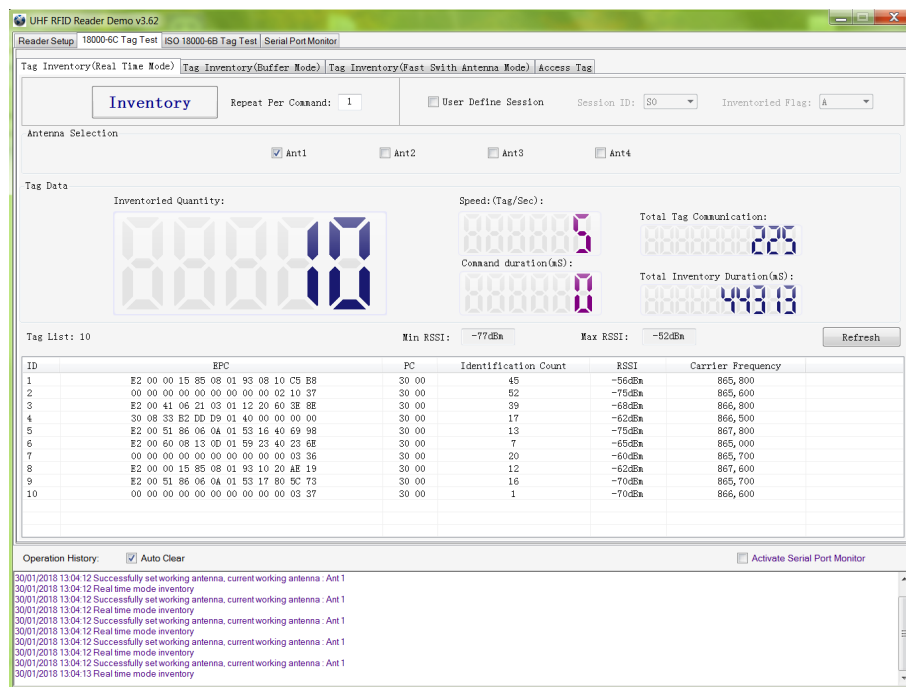


Figura 4.2: Captura de pantalla del *Software Demo* del fabricante del lector/antena RFID. Lista de etiquetas RFID detectadas.

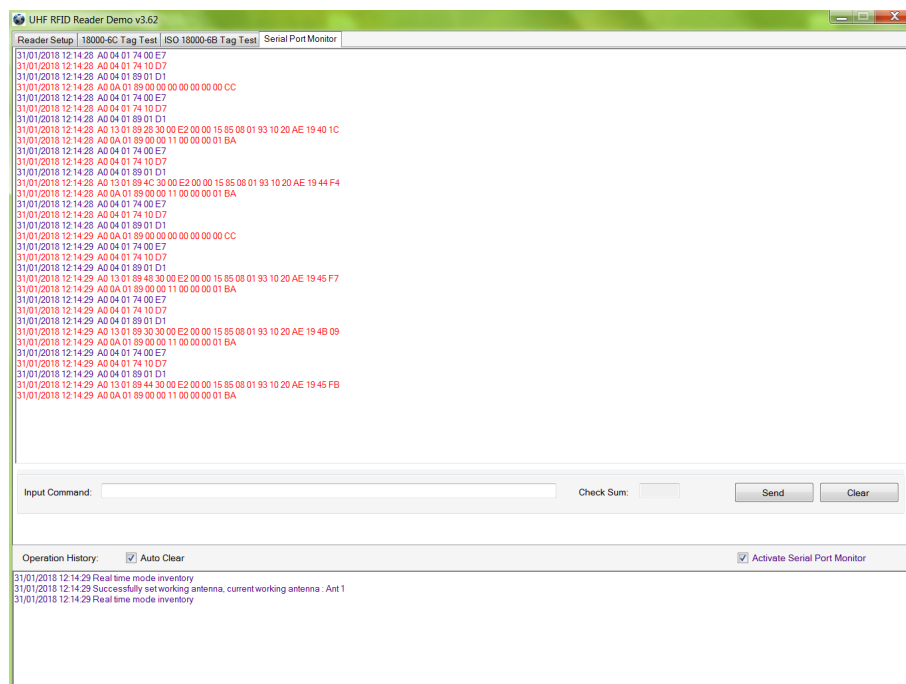


Figura 4.3: Captura de pantalla del *Software Demo* del fabricante del lector/antena RFID. Paquetes transmitidos y recibidos en formato hexadecimal.



En la Figura 4.3 se muestra un paquete de datos en cada línea. Los paquetes de color rojo son los transmitidos por el lector RFID y los paquetes en azul son transmitidos por el PC en el que se ejecuta el *Software Demo* del fabricante. Esta prueba ha servido para identificar la secuencia de paquetes enviados y recibidos por el lector RFID en condiciones normales. Esta es la secuencia de acontecimientos que se ha observado:

1. El *Software Demo* comienza transmitiendo un paquete *cmd-set-work-antenna* (*A0 04 01 74 00 E7*).
2. El lector RFID transmite un paquete de respuesta *cmd-set-work-antenna* indicando que no hay errores (*A0 04 01 74 10 D7*).
3. El PC transmite un paquete *cmd-real-time-inventory* (*A0 04 01 89 01 D1*). Con este comando el lector RFID comprueba si hay etiquetas RFID en ese momento y transmite tantos paquetes de datos como etiquetas haya leído en tiempo real (en caso de haber etiquetas en su entorno).
4. El lector RFID transmite un único paquete de respuesta *cmd-real-time-inventory* en el que se observa que no ha leído ninguna etiqueta en ese momento (al comenzar a realizar esta prueba no se encontraba ninguna etiqueta RFID en el entorno del lector). Este paquete muestra que no lleva información de ninguna etiqueta porque el campo *RSSI* tiene valor 0. (*A0 0A 01 89 00 00 00 00 00 00 CC*).
5. El PC transmite un paquete *cmd-set-work-antenna*.
6. El lector RFID transmite un paquete de respuesta *cmd-set-work-antenna* indicando que no hay errores.
7. El PC transmite un paquete *cmd-real-time-inventory*.
8. El lector RFID transmite dos respuestas:
  - La primera respuesta contiene la información leída de una etiqueta RFID (*A0 13 01 89 28 30 00 E2 00 00 15 85 08 01 93 10 20 AE 19 40 1C*). Se observa que el campo *RSSI* contiene un valor mayor que 1.
  - La segunda indica que no hay más etiquetas en ese momento (*A0 0A 01 89 00 00 11 00 00 00 01 BA*).

Esta secuencia de paquetes en la comunicación entre el *Software Demo* y el Lector RFID se repite siempre, en las múltiples pruebas realizadas.

Existe una particularidad en el caso de que exista más de una etiqueta RFID en el entorno del lector. En este caso, el lector RFID transmite varias respuestas a un paquete *cmd-real-time-inventory*:

- Un paquete por cada etiqueta detectada
- Un paquete posterior indicando que no hay más etiquetas en su entorno.

Una vez comprobado el funcionamiento del lector RFID en el modo de operación que interesa para este proyecto, se prueba transmitir y recibir directamente estos paquetes en formato hexadecimal sin emplear el *Software Demo* proporcionado por el fabricante. Para ello se emplea el programa YAT (*Yet Another Terminal*).

Gracias a esta prueba se ha observado que no es necesario enviar siempre el paquete *cmd-set-work-antenna* previo envío del paquete *cmd-real-time-inventory* para que el lector RFID detecte las etiquetas. Únicamente es necesario enviar un paquete *cmd-set-work-antenna* para poner en funcionamiento el lector RFID, y enviar un paquete *cmd-real-time-inventory* cada vez que se desee que el lector detecte etiquetas.

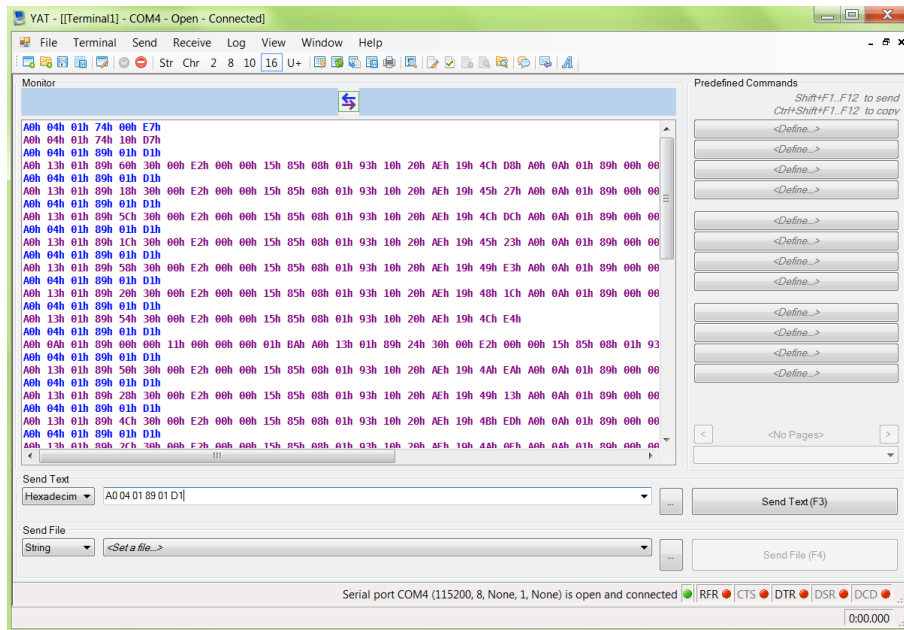


Figura 4.4: Captura de pantalla de pruebas realizadas con el Software YAT

Sin embargo, si se transmite siempre un paquete *cmd-set-work-antenna* previo envío del paquete *cmd-real-time-inventory*, se está comprobando en todo momento que la antena funciona correctamente. Si hubiera algún tipo de error en la antena la respuesta al paquete *cmd-set-work-antenna* contendría un código de error. Por tanto, la mejor opción a la hora de desarrollar la aplicación será implementar esta secuencia.

#### 4.2.3.2. Paso 2: Pruebas sobre la conexión serie de la *Raspberry Pi*

El objetivo de esta prueba ha sido comprobar el correcto funcionamiento del puerto serie de la *Raspberry Pi* y del adaptador *MAX3232*. Además, realizar pruebas con la biblioteca *pyserial*.

La conexión del adaptador *MAX3232* a la *Raspberry Pi* se ha realizado según se describe en la 4.3. La conexión del adaptador *MAX3232* al PC se ha realizado directamente mediante un cable serie.

Una vez realizado el montaje, se ha comprobado que los componentes *hardware* se encuentran conectados y funcionan correctamente. Para ello se ha ejecutado directamente *python* desde la terminal *linux* de la *Raspberry Pi*, para ejecutar funciones de la biblioteca *pyserial* que realizan operaciones de lectura y escritura en el puerto serie. Mientras tanto, en el PC se han estado enviando y recibiendo *bytes* de datos a través del puerto serie empleando el *software* YAT.

#### 4.2.3.3. Paso 3: Pruebas de funcionamiento del lector RFID con la *Raspberry Pi*

El objetivo de esta prueba ha sido comprobar que el lector RFID opera correctamente con la *Raspberry Pi* y realiza lecturas de etiquetas que se encuentran en su entorno.

La conexión del adaptador *MAX3232* a la *Raspberry Pi* y la conexión del adaptador *MAX3232* al lector RFID, se han realizado según se describe en la sección 4.3.

Una vez realizado el montaje se ha ejecutado directamente *python* desde la terminal *linux* de la *Raspberry Pi*, para ejecutar funciones de la biblioteca *pyserial* que realizan operaciones de lectura y escritura en el puerto serie.

Primero, se ha transmitido un paquete *cmd-set-work-antenna*, y se ha comprobado que la respuesta del lector RFID ha sido un paquete *cmd-set-work-antenna* en el que el campo de errores indica que el lector/antena RFID funciona correctamente. En segundo lugar, se ha transmitido un paquete *cmd-real-time-inventory* y se ha analizado la respuesta, comprobando si se ha detectado una etiqueta RFID o no.

Esta prueba se ha repetido en numerosas ocasiones, para los casos en los que no había ninguna etiqueta en el entorno, para el caso de haber una etiqueta y el caso de haber varias etiquetas en el entorno. Con todas estas pruebas se ha podido verificar que la *Raspberry Pi* y el lector RFID operan perfectamente entre sí.

### 4.3. Manipulaciones Hardware

Como ya se ha mencionado, la comunicación entre la *Raspberry Pi* y el lector RFID requiere emplear un adaptador, concretamente se emplea un *Serial Port To TTL Converter Module MAX3232*. Este módulo dispone de un puerto serie (hembra) al cual se conecta un cable serie (macho - macho), que es conectado en el otro extremo al puerto serie del lector RFID (hembra). Por otro lado, el módulo dispone de 4 pines donde conectar por medio de cables puente (hembra - hembra) los pines de la *Raspberry Pi*.

#### 4.3.1. Conexión *Raspberry Pi* - Módulo *MAX3232*

La *Raspberry Pi* cuenta con un total de 40 pines en placa [How18] que siguen una numeración según su posición física. Algunos de estos pines tienen características especiales (UART, SPI, I2C, etc.), mientras que otros únicamente pueden ser empleados como entradas/salidas digitales de propósito general. Además, existen varios pines destinados a proporcionar alimentación a 3,3 o 5 V, y como tierra. De cara a realizar la conexión con el módulo *MAX3232*, los pines se conectan de la forma mostrada en la siguiente tabla:

| Módulo conversor | Raspberry Pi                     |
|------------------|----------------------------------|
| GND              | GROUND (pin físico número 6)     |
| RXD              | RXD (pin físico número 10)       |
| TXD              | TXD (pin físico número 8)        |
| VCC              | 3.3V POWER (pin físico número 1) |

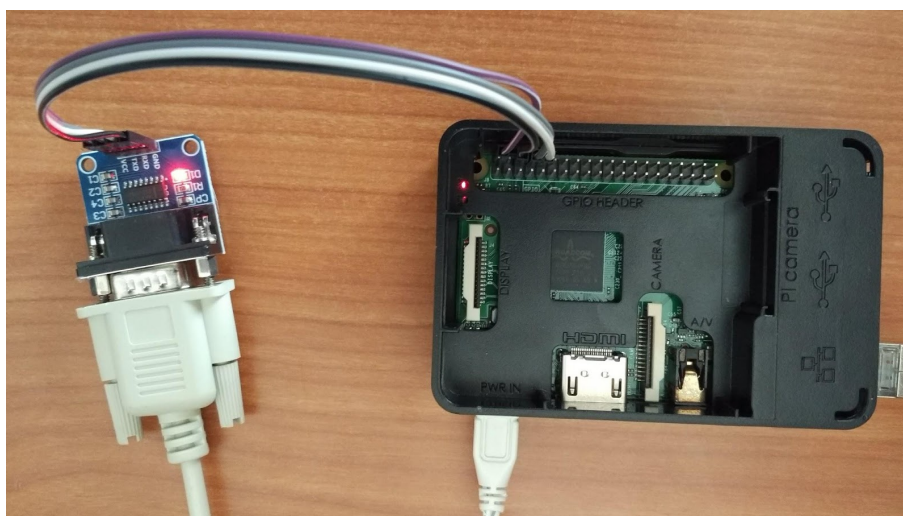


Figura 4.5: Fotografía de una Raspberry Pi 2 conectada a un módulo conversor MAX3232

### 4.3.2. Conexión lector RFID - Módulo MAX3232

La conexión entre el lector RFID y el módulo *MAX3232* requiere conectar un cable serie (macho - macho) entre los dispositivos. Sin embargo, cuando se realiza esta conexión, se está conectando la salida *TXD* de la *Raspberry Pi* a la salida *TXD* del lector RFID, y lo mismo ocurre para las salidas *RXD*.

Por tanto, es necesario intercambiar los pines de la conexión serie para que el *TXD* de la *Raspberry Pi* se conecte al *RXD* del lector RFID, y el *RXD* de la *Raspberry Pi* se conecte al *TXD* del lector RFID. Para lograr este objetivo, una opción podría haber sido emplear un adaptador intermedio que intercambia los pines. Una segunda opción (por la cual se ha optado) consiste en desmontar el conector serie del lector RFID, soltar las conexiones con un estañador, para después soldar las conexiones intercambiando los pines 2 y 3 que son los correspondientes a los *RXD* y *TXD*, y volver a montar el conector serie.

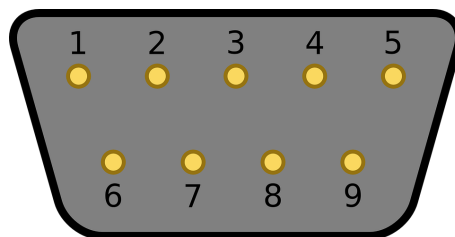


Figura 4.6: Imagen descriptiva de los pines de un conector serie (RS232 DB9)

| Pin conector | Función                |
|--------------|------------------------|
| 1            | Carrier Detect         |
| 2            | RXD (Received Data)    |
| 3            | TXD (Transmitted Data) |
| 4            | Data Terminal Ready    |
| 5            | Common Ground          |
| 6            | Data Set Ready         |
| 7            | Request To Send        |
| 8            | Clear To Send          |
| 9            | Ring Indicator         |

Cuadro 4.1: Función de cada pin en un conector serie (RS232 DB9)

## 4.4. Desarrollo Software

Para llevar a cabo los cometidos de un equipo de cronometraje ha sido necesario desarrollar una aplicación que se ejecuta en la *Raspberry Pi*.

### 4.4.1. Configuraciones e instalaciones previas

Es necesario llevar a cabo unas configuraciones e instalaciones en la *Raspberry Pi* para tener el equipo listo de cara a ejecutar una aplicación.

La *Raspberry Pi* debe tener instalado el sistema operativo *Raspbian*. Para ello, la *Raspberry Pi Foundation* proporciona la imagen del sistema e instrucciones para instalarlo [Fou18c].

El sistema operativo no tiene habilitado por defecto el puerto serie de la *Raspberry Pi*. Para ello se debe habilitar la conexión *serial* y desactivar el *login shell* a través de *serial*. La necesidad de desactivar el *login shell* viene motivada para evitar que el puerto sea empleado para acceso *shell*, y se emplee en exclusiva para las funciones del proyecto. Para llevar a cabo esta tarea es necesario ejecutar en un terminal con permisos *root* el comando *raspi-config*, tras ejecutar este comando se muestra una interfaz gráfica que permite seleccionar fácilmente las opciones necesarias.

Por otro lado, ya que para el desarrollo de esta aplicación se emplea la biblioteca *pyserial*, es necesario su instalación. Para instalar esta biblioteca se ejecuta el comando *pip install pyserial*.

#### 4.4.2. Desarrollo de aplicación para *Raspberry Pi*

El desarrollo se ha realizado siguiendo un enfoque de programación funcional. La aplicación desarrollada implementa las funcionalidades propias de un equipo de cronometraje. Éstas funcionalidades consisten en la comunicación con el lector RFID para detectar las etiquetas que se encuentren en su entorno, así como asignar tiempos a la lectura de dichas etiquetas y transferir la información a la aplicación *web*. Además, la aplicación desarrollada detecta errores del lector RFID y transmite el estado del mismo a la aplicación *web*.

Un aspecto importante de la aplicación es la capacidad de operar aun cuando existen problemas en la red, es decir, cuando el envío de la información a la aplicación *web* no obtiene respuesta del servidor. En estos casos, la aplicación almacena la información en un fichero local para reintentar el envío posteriormente.

Para llevar a cabo estas funcionalidades se ha desarrollado la aplicación dividiendo las responsabilidades en dos hilos de ejecución. La necesidad de dos hilos de ejecución viene motivada por la posibilidad de que se produzca un bloqueo durante la ejecución de la aplicación. Los bloqueos que podrían producirse si no se emplearan dos hilos ocurrirían según la siguiente secuencia de acontecimientos:

1. La aplicación solicita al lector RFID que realice una lectura de etiquetas.
2. El lector detecta una etiqueta RFID y responde a la aplicación con un paquete que contiene el EPC de la misma.
3. La aplicación asigna una marca de tiempo asociada al EPC de la etiqueta detectada.
4. La aplicación intenta transmitir la información a la aplicación *web* que se encuentra en un servidor y queda a la espera de la respuesta del servidor.
5. Una nueva etiqueta RFID aparece en el entorno del lector, sin embargo la aplicación sigue esperando la respuesta del servidor.
6. La nueva etiqueta RFID desaparece del entorno del lector.
7. La aplicación recibe la respuesta del servidor y continúa su comunicación con el lector RFID para detectar más etiquetas.

Con la secuencia de acontecimientos descrita, una etiqueta RFID aparece en el entorno del lector RFID mientras se realiza un envío de información al servidor, por lo tanto, no es detectada. A efectos prácticos, dicha etiqueta podría estar asignada a un participante, que quedaría sin ser detectado por el sistema. Para evitar esta situación, la aplicación se divide en dos hilos que se ejecutan al mismo tiempo y comparten información: un hilo encargado de la comunicación con el lector RFID (*hilo-antena*) y un hilo encargado del resto de responsabilidades de la aplicación (*hilo-gestion*).

### Hilo encargado de la comunicación con el lector RFID

Como se ha analizado en la sección 4.2, el lector RFID cuenta con varios modos de operación, de entre los cuales nos interesa el denominado *Real time inventory*. Con este método de operación, cada vez que el lector RFID recibe un paquete *cmd-real-time-inventory* responde con un paquete por cada una de las etiquetas que se encuentran en su entorno seguidos de un paquete indicando que no hay más etiquetas.

En la aplicación desarrollada, la comunicación con el lector consiste en el envío continuo de paquetes *cmd-set-work-antenna*, según la respuesta del lector RFID la aplicación puede determinar si el lector/antena RFID está funcionando correctamente o si hay errores. En caso de que el funcionamiento sea correcto, tras la respuesta al paquete *cmd-set-work-antenna* se transmite un paquete *cmd-real-time-inventory*. En caso de que no recibir respuesta se transmite de nuevo un paquete *cmd-set-work-antenna* tras 40 milisegundos. Este valor temporal ha sido obtenido mediante pruebas con diferentes valores, partiendo del mínimo valor que puede ser configurado: 20 milisegundos que se obtienen del número máximo de etiquetas detectables por segundo (50 etiquetas/s).

Con cada respuesta *cmd-set-work-antenna* se obtiene el estado de la antena, que puede estar funcionando correctamente o encontrarse bajo un error de operación. En cualquiera de los casos se almacena el estado en una variable que es compartida con el hilo de ejecución encargado de la gestión.

Con cada respuesta *cmd-real-time-inventory* se obtiene el EPC de la etiqueta RFID detectada, se le asigna una marca de tiempo y se almacena en una variable que es compartida con el hilo de ejecución encargado de la gestión. Los paquetes cuyo valor RSSI (*Received Signal Strength Indicator*) es muy bajo son descartadas, ya que indican que el lector no detecta más etiquetas en su entorno.

| Head                     | Len     | Address | Cmd  | Data    | Check  |
|--------------------------|---------|---------|--|---------|--------|
| 0xA0                     | 1 Byte  | 1 Byte  | 1 Byte   | N Bytes | 1 Byte |
|                          |         |         |  |         |        |
| Parameter<br>Description | Head    |         | Head of the packet, every packet starts with 0xA0.   |         |        |
|                          | Len     |         | Length of the packet bytes. Starts from the third byte, the Head, Len bytes are exclusive.   |         |        |
|                          | Address |         | Reader's address for RS-485 connection. The common addresses are 0~254(0xFE), 255 (0xFF) is the public address. The reader accepts the address of itself and the public address. |         |        |
|                          | Cmd     |         | Command byte.  |         |        |
|                          | Data    |         | Command parameters.  |         |        |
|                          | Check   |         | Checksum. Check all the bytes except itself.   |         |        |

Figura 4.7: Definición de un paquete de datos transmitido al lector RFID.

### Hilo encargado de la gestión

Las responsabilidades que recaen en este hilo de ejecución consisten principalmente en enviar a la aplicación web la información obtenida por el hilo de comunicación con el lector RFID. Además, en caso que el envío no obtenga respuesta del servidor, se almacena la información en local y se reintenta posteriormente.

En este hilo se está continuamente escaneando la variable compartida con el hilo de comunicación con la antena que almacena las etiquetas leídas y su marca de tiempo (se trata de un *diccionario*

| Head                     | Len     | Address | Data   | Check  |
|--------------------------|---------|---------|--|--------|
| 0xA0                     | 1 Byte  | 1 Byte  | N Bytes  | 1 Byte |
|                          |         |         |  |        |
| Parameter<br>Description | Head    |         | Head of the packet, every packet starts with 0xA0.   |        |
|                          | Len     |         | Length of the packet bytes. Starts from the third byte, the Head, Len bytes are exclusive. |        |
|                          | Address |         | Reader's address.  |        |
|                          | Data    |         | Data from the reader.  |        |
|                          | Check   |         | Checksum. Check all the bytes except itself.   |        |

Figura 4.8: Definición de un paquete de datos transmitido al lector RFID.

◆ **Succeeded:**

Response packet:

| Head | Len  | Address | Cmd  | ErrorCode      | Check |
|------|------|---------|------|----------------|-------|
| 0xA0 | 0x04 |         | 0x74 | CommandSuccess |       |

◆ **Failed:**

Response packet:

| Head                     | Len       | Address | Cmd         | ErrorCode | Check |
|--------------------------|-----------|---------|-------------|-----------|-------|
| 0xA0                     | 0x04      |         | 0x74        |           |       |
|                          |           |         |             |           |       |
| Parameter<br>Description | ErrorCode |         | Error code. |           |       |

Figura 4.9: Definición de un paquete cmd-set-work-antenna recibido del lector RFID.

| Head                     | Len     | Address | Cmd  | FreqAnt   | PC      | EPC     | RSSI | Check |
|--------------------------|---------|---------|------|---|---------|---------|------|-------|
| 0xA0                     |         |         | 0x89 |   | 2 bytes | N bytes |      |       |
|                          |         |         |      |   |         |         |      |       |
| Parameter<br>Description | FreqAnt |         |      | The high 6 bits are frequency parameter; the low 2 bits are antenna ID. |         |         |      |       |
|                          | PC      |         |      | Tag's PC. 2 bytes.  |         |         |      |       |
|                          | EPC     |         |      | Tag's EPC.  |         |         |      |       |
|                          | RSSI    |         |      | The RSSI when tag is identified.  |         |         |      |       |

Figura 4.10: Definición de un paquete cmd-real-time-inventory recibido del lector RFID.

*python*). En caso de existir una nueva etiqueta, extrae la misma de la variable e intenta enviarla a la aplicación web. En este TFG el envío de esta información se denomina envío de cronometraje, y consiste en el envío de un *HTTP Request* en formato JSON cuyo contenido es un identificador de la etiqueta RFID leída (EPC) y una marca de tiempo. En caso de que no se reciba respuesta del servidor, se almacena esta información en un fichero local.

Cada 60 segundos se comprueba el valor de la variable compartida que contiene el estado del lector RFID y se comunica el estado a la aplicación web. En caso de que no se reciba respuesta del servidor, se almacena esta información en un fichero local. En caso de recibir respuesta del servidor, se procede a reintentar los envíos de cronometraje que se encontraban pendientes de reenvío.

#### 4.4.3. Instalación de aplicación en *Raspberry Pi*

Un requisito de los equipos de cronometraje es que comiencen a operar con solo ser conectados. Para conseguir este objetivo ha sido necesario configurar la aplicación para que se inicie durante el arranque de la *Raspberry Pi*. Por lo tanto, la instalación de la aplicación consiste en ubicar el fichero que contiene el código en un directorio local y, configurar la ejecución del fichero con el arranque de la *Raspberry Pi*.

La ubicación elegida para el fichero es el directorio *home* del usuario por defecto de *Raspbian pi*, esta es la ruta completa: */home/pi/sced* (sced es un acrónimo de sistema de cronometraje de eventos deportivos).

Para conseguir que la aplicación desarrollada se inicie con el arranque de la *Raspberry Pi* se han realizado configuraciones en el fichero *rc.local*. El fichero *rc.local* [Fou18f] permite configurar la ejecución de *scripts* al final del arranque del sistema operativo tras el *runlevel*. Para ello ha sido necesario editar el fichero */etc/rc.local* con permisos *root*, añadiendo en una nueva línea: *python3 /home/pi/sced/sced-RPi.py*  $\mathcal{E}$ .

### 4.5. Componentes y precios

En esta sección se muestra una tabla con el desglose de precios de cada componente de un equipo de cronometraje. Es necesario tener en cuenta que estos precios pueden sufrir variaciones, sobre todo el precio del lector/antena RFID ya que se está importando de China y el tipo de cambio de divisas fluctúa constantemente.

Los precios mostrados incluyen transporte e IVA. En el caso de los componentes importados de China es de gran importancia este aspecto, ya que al realizar la compra a través de Internet se muestran los precios antes de impuestos, y éstos se pagan al estado al llegar el producto a España (habitualmente la empresa de transportes se encarga de la gestión).

El precio de la *Raspberry Pi* incluye la carcasa, el adaptador de corriente y la tarjeta SD. El precio del lector/antena RFID incluye amarres para poder ser instalado en un poste de hierro o similar.

|   |              |
|---|--------------|
| <b>Raspberry Pi</b>                     | 80 €         |
| <b>Módulo MAX3232</b>                   | 1 €          |
| <b>Lector/antena RFID INNOD IND8001</b> | 180 €        |
| <b>Cable Serie</b>                      | 2 €          |
| <b>Total</b>                            | <b>263 €</b> |

Cuadro 4.2: Desglose de precios de los componentes de un equipo de cronometraje.



## Capítulo 5

# Desarrollo de la aplicación web

### 5.1. Descripción de la aplicación web

La aplicación *web* es el núcleo del sistema de cronometraje, y contiene la mayor parte de la lógica del sistema. Mientras que los equipos de cronometraje únicamente se encargan de identificar a los participantes sin necesidad de dar sentido a la información recabada, en la aplicación *web* reside toda la complejidad necesaria para determinar si la información recabada por los equipos de cronometraje es válida, procesar la información y ofrecer resultados a partir de dicha información, es decir, generar clasificaciones mediante tablas de tiempos. Además, la aplicación *web* proporciona las funcionalidades necesarias para la gestión de los eventos deportivos a través de un panel de administración, así como permitir el seguimiento de los eventos deportivos en directo.

#### 5.1.1. Panel de administración

La aplicación *web* cuenta con un panel de administración accesible mediante usuario y contraseña. En este panel se encuentran diversas secciones que permiten gestionar los equipos de cronometraje, así como los eventos deportivos y todas las particularidades de los mismos. En el panel de administración se encuentran las siguientes funcionalidades:

##### Gestionar equipos de cronometraje

Desde el panel de administración se puede acceder a una lista de los equipos de cronometraje registrados en el sistema, ver en detalle un equipo de cronometraje, así como añadir, editar y eliminar un equipo de cronometraje. Cada equipo de cronometraje cuenta con un nombre del equipo (nombre amigable para poder ser reconocido fácilmente) que puede ser definido desde el panel de administración.

La aplicación web genera automáticamente un identificador de dispositivo, una *API KEY*, y unas URLs donde el equipo podrá transmitir los envíos de cronometraje y la información de estado. Estos parámetros son los necesarios para configurar la aplicación en la *Raspberry Pi*.

Además, en la aplicación *web* se puede visualizar el estado actual de cada equipo de cronometraje, así como el tiempo transcurrido desde la última comunicación sobre el estado del mismo. El estado de un equipo de cronometraje puede ser cualquiera de los siguientes: *Operativo*, *Conectado*, *no Operativo*, *Desconectado*.

##### Gestionar eventos deportivos

Desde el panel de administración se puede ver en una lista de los eventos deportivos registrados en el sistema, ver en detalle un evento, así como añadir, editar y eliminar un evento. Cada evento deportivo cuenta con varios parámetros que pueden ser definidos desde el panel de administración: nombre del evento, descripción, fecha y hora, zona horaria, lugar y, tiempo mínimo de medición. Tanto la fecha y hora como la zona horaria son parámetros informativos sin ninguna validez para el cálculo

de tiempos. El tiempo mínimo de medición es un valor que se emplea para determinar cuanto tiempo debe transcurrir desde el comienzo del evento hasta que se comienzan a aceptar envíos de cronometraje procedentes de los equipos de cronometraje, es un parámetro especialmente útil en los casos en los que se dispone de un recorrido circular (la llegada se sitúa en el mismo lugar de la meta) a fin de evitar la obtención de tiempos incorrectos durante la salida.

Además, un evento deportivo cuenta con parámetros que no pueden ser editados manualmente, si no a través de funciones específicas, éstos son el estado del evento y el instante de comienzo. El estado de un evento deportivo puede ser: *Evento por comenzar*, *En directo* y *Finalizado*. El instante de comienzo contiene la fecha y hora en la que se ha dado la salida y, sirve para conocer el tiempo transcurrido desde que ha dado comienzo, así como para calcular los tiempos obtenidos por los participantes. Ambos parámetros son modificados mediante las funciones de *Comenzar evento* y *Finalizar evento*.

Estos parámetros componen los datos básicos de un evento deportivo. Se consideran datos básicos ya que el evento deportivo alberga mayor complejidad que se gestiona a través de subpantallas:

**Comenzar evento** Desde el panel de un evento deportivo individual se muestra un botón que permite dar comienzo al evento deportivo (dar la salida). Al pulsar este botón, el estado del evento se establece como comenzado y se fija el tiempo a partir del cual se realizan los cálculos de tiempo obtenidos por los participantes. Además, se comenzarán a aceptar envíos de cronometraje una vez que haya transcurrido el tiempo mínimo de medición configurado.

**Finalizar evento** Desde el panel de un evento deportivo individual se muestra un botón que permite dar por finalizado el evento deportivo. Al pulsar este botón, el estado del evento se establece como finalizado y a partir de ese instante no se admitirán nuevos envíos de cronometraje.

**Gestionar recorridos** Desde el panel de un evento deportivo se muestra una subsección para la gestión de los recorridos del evento. En esta subsección se puede ver en una lista los recorridos registrados en el evento deportivo, ver en detalle un recorrido, así como añadir, editar y eliminar un recorrido. Cada recorrido cuenta con varios parámetros que pueden ser definidos desde el panel de administración: nombre del recorrido, descripción, número de vueltas y, tiempo mínimo por vuelta. Los parámetros número de vueltas y, tiempo mínimo por vuelta se emplean para determinar si un envío de cronometraje debe ser aceptado o no. Además, dentro de la subsección de un recorrido, se muestran los puntos de control que están configurados para dicho recorrido ordenados por punto kilométrico.

**Gestionar puntos de control** Desde la subsección de un recorrido se pueden gestionar los puntos de control de dicho recorrido. En esta subsección se pueden ver los puntos de control que están configurados para dicho recorrido ordenados por punto kilométrico, ver en detalle un punto de control, así como añadir, editar y eliminar un punto de control. Cada punto de control cuenta con varios parámetros que pueden ser definidos desde el panel de administración: nombre del punto de control, punto kilométrico y equipo de cronometraje. El punto kilométrico se emplea para ordenar los puntos de control dentro de un recorrido, y es de gran importancia para la clasificación final. El equipo de cronometraje es el equipo que se emplea para identificar a los participantes a su paso por el punto de control, desde la pantalla de añadir punto de control y la pantalla de editar punto de control se puede seleccionar el equipo de cronometraje de entre los que se encuentran registrados en la aplicación web siempre que no se esté empleando en otro punto de control del mismo recorrido.

**Gestionar categorías** Desde el panel de un evento deportivo se muestra una subsección para la gestión de las categorías del evento. En esta subsección se puede ver en una lista las categorías registradas en el evento deportivo, ver en detalle una categoría, así como añadir, editar y eliminar una categoría. Cada categoría cuenta con varios parámetros que pueden ser definidos desde el panel de administración: nombre de la categoría, recorrido a realizar y descripción. Desde la pantalla de añadir categoría y la pantalla de editar categoría se puede seleccionar el recorrido a realizar de entre los que

se encuentran configurados para el mismo evento deportivo. Cada categoría debe tener configurado un recorrido y, existe la posibilidad de que varias categorías realicen el mismo recorrido.

**Gestionar participantes** Desde el panel de un evento deportivo se muestra una subsección para la gestión de los participantes del evento. En esta subsección se puede ver en una lista los participantes registrados en el evento deportivo, ver en detalle un participante, así como añadir, editar y eliminar un participante. Cada participante cuenta con varios parámetros que pueden ser definidos desde el panel de administración: categoría, nombre, dorsal y etiqueta RFID. Desde la pantalla de añadir participante y la pantalla de editar participante se puede seleccionar la categoría a la que pertenece el participante de entre las que se encuentran configuradas para el mismo evento deportivo. Cada participante debe tener asignada una categoría.

El parámetro etiqueta RFID contiene el EPC de la etiqueta RFID que tiene asociada el participante, cuyo formato se compone de doce bytes en decimal separados por guiones, por ejemplo: *226-0-81-134-6-10-1-83-22-64-105-152*. Este parámetro puede ser introducido manualmente o empleando el botón de leer etiqueta, mediante este botón, la etiqueta RFID puede ser leída empleando un equipo de cronometraje que se encuentre registrado en la aplicación *web*.

### 5.1.2. Seguimiento de eventos deportivos

La aplicación *web* permite a visitantes acceder al seguimiento en directo de un evento deportivo a través de una URL pública. En caso de que el evento deportivo no haya comenzado, se indica que el evento está por comenzar, además de mostrar los datos básicos del evento deportivo. En caso de que el evento deportivo haya comenzado y se encuentre en directo, se muestran las categorías del evento deportivo con una clasificación en forma de tabla de tiempos por cada categoría. En caso de que el evento deportivo haya finalizado, se muestran las mismas tablas de tiempos indicando que los tiempos son definitivos.

## 5.2. Usuarios y acceso al sistema

La aplicación es accesible desde un navegador *web* que se encuentre en un ordenador o un dispositivo móvil, estos dispositivos deben disponer de conexión a Internet. Además, la aplicación atenderá solicitudes HTTP realizadas en formato JSON.

El sistema podrá ser accedido simultáneamente desde múltiples lugares. El número de accesos simultáneos no será muy alto, pero si se esperan unos pocos accesos al mismo tiempo durante la realización de eventos deportivos.

El sistema considera los siguientes actores:

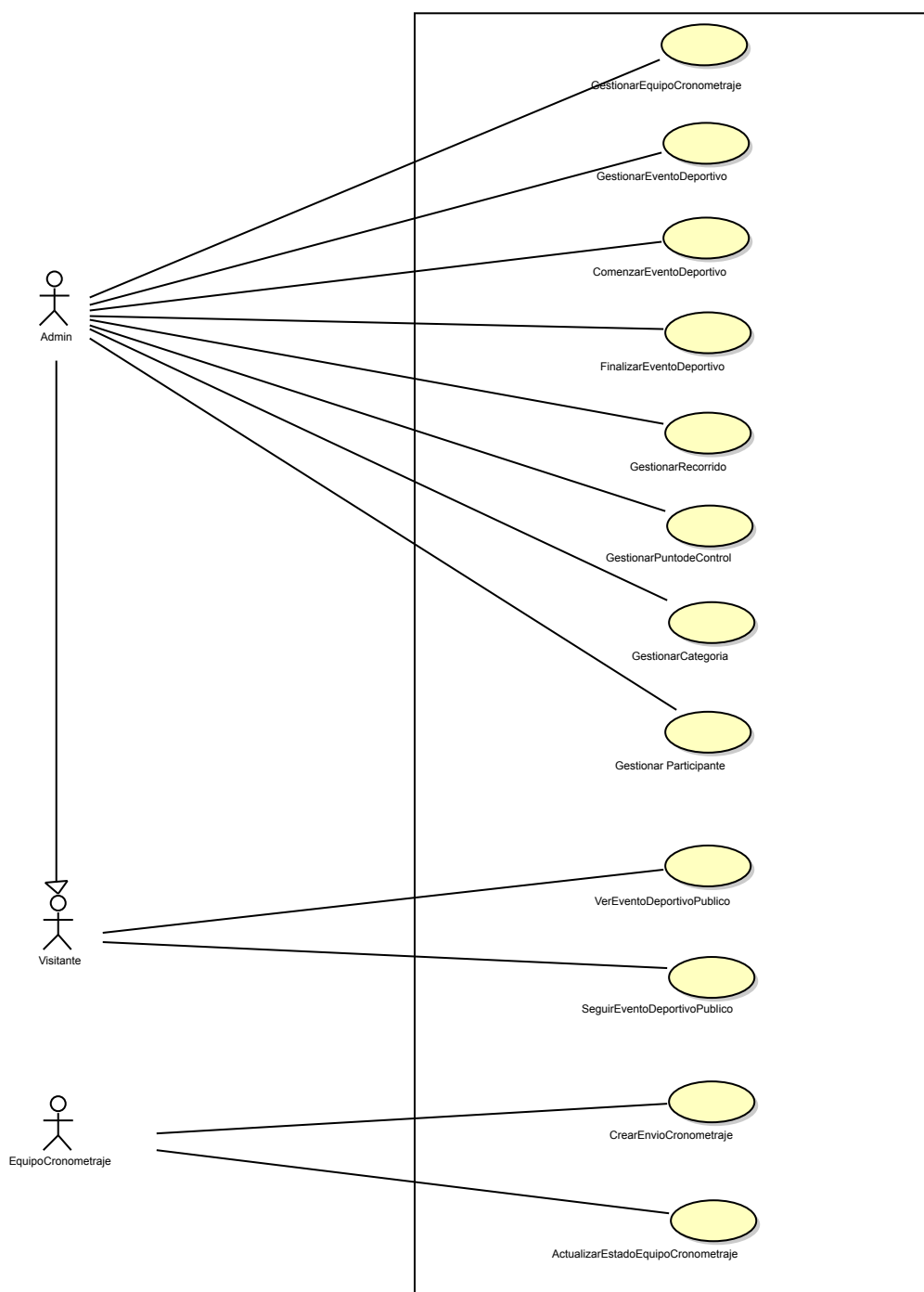
- **Visitante.** Un *Visitante* puede acceder a través de una URL pública a la información general de un evento deportivo, así como acceder al seguimiento del mismo en directo visualizando las tablas de tiempos durante la realización del evento y tras haber finalizado el mismo.
- **Administrador.** El *Administrador* puede acceder a un panel de administración previa autenticación mediante su usuario y contraseña. En dicho panel de administración puede gestionar toda la complejidad de los eventos deportivos, así como gestionar los equipos de cronometraje.
- **EquipoCronometraje.** Un *EquipoCronometraje* puede transmitir información en formato JSON a la aplicación a través de unas URL habilitadas específicamente para dicho propósito, previa autorización mediante un identificador de equipo y una *API KEY*. La información transmitida por un *EquipoCronometraje* puede ser de dos tipos:
  - **Envíos de cronometraje,** que contienen la información básica cuando los equipos físicos han detectado una etiqueta RFID: EPC de la etiqueta y marca de tiempo.

- **Envíos de estado**, que contienen el estado actual de un equipo.

### 5.3. Casos de uso

En esta sección se incluyen los casos de uso del sistema de cronometraje que capturan las funcionalidades del sistema correspondientes a la aplicación web. Para ello, se presenta un diagrama de casos de uso siguiendo notación UML y, los casos de uso detallados donde se describe la interacción de los actores con la aplicación *web*.

#### 5.3.1. Diagrama de casos de uso



## 5.3.2. Casos de uso detallados

|                              |  |   |
|------------------------------|--|---|
| <b>Id</b>                    | <b>CrearEquipoCronometraje</b>   |   |
| <b>Versión</b>               | 1.0  |   |
| <b>Actores</b>               | <i>Administrador.</i>  |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede registrar un nuevo equipo de cronometraje en el sistema.   |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.  |   |
| <b>Post-condición</b>        | El sistema genera automáticamente un identificador de dispositivo y un <i>API KEY</i> . Se crea un nuevo equipo de cronometraje en el sistema. |   |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> selecciona <i>Añadir Equipo de Cronometraje</i> .   |
|                              | 2  | El sistema muestra un formulario para añadir el equipo de cronometraje. Este formulario cuenta con un único campo: <i>Nombre del equipo de cronometraje</i> . |
|                              | 3  | El <i>Administrador</i> introduce el nombre del equipo y pulsa <i>Guardar</i> .   |
|                              | 4  | El sistema comprueba que el dato introducido es válido.   |
|                              | 5  | El sistema genera automáticamente un identificador de dispositivo y un <i>API KEY</i> .   |
|                              | 6  | El sistema almacena el equipo de cronometraje en la base de datos.  |
|                              | 7  | El sistema redirige al usuario a un panel donde se muestran todos los datos del equipo de cronometraje creado.  |
| <b>Flujos alternativos</b>   | 3-B  | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |
|                              | 4-C  | El sistema reconoce un error en el nombre del equipo de cronometraje, y muestra el formulario con un mensaje de error.  |
| <b>Frecuencia de uso</b>     | Baja.  |   |
| <b>Requisitos especiales</b> | Ninguno.   |   |
| <b>Comentarios</b>           | No hay.  |   |

|                              |   |   |  |
|------------------------------|---|---|--|
| <b>Id</b>                    | <b>VerEquipoCronometraje</b>  |   |  |
| <b>Versión</b>               | 1.0   |   |  |
| <b>Actores</b>               | <i>Administrador.</i>   |   |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede acceder a todos los datos y estado de un equipo de cronometraje en el sistema.  |   |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.   |   |  |
| <b>Post-condición</b>        | Los datos del equipo de cronometraje han sido mostrados por pantalla.   |   |  |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona un equipo de cronometraje pulsando en el enlace del mismo. |  |
|                              | 2   | El sistema comprueba que el equipo de cronometraje existe.                                    |  |
|                              | 3   | El sistema muestra toda la información del equipo de cronometraje, así como su estado actual. |  |
| <b>Flujos alternativos</b>   | 3-B   | El sistema no encuentra el equipo de cronometraje y muestra un mensaje de error.              |  |
| <b>Frecuencia de uso</b>     | Media.  |   |  |
| <b>Requisitos especiales</b> | Ninguno.  |   |  |
| <b>Comentarios</b>           | <p>El estado actual de un Equipo de Cronometraje puede tener 3 valores:</p> <ul style="list-style-type: none"><li>▪ <b>Operativo:</b> El dispositivo de equipo de cronometraje se encuentra conectado al sistema, con capacidad para leer etiquetas RFID y enviarlas al sistema.</li><li>▪ <b>Conectado, no operativo:</b> El dispositivo de equipo de cronometraje se encuentra conectado al sistema, sin capacidad para leer etiquetas RFID y enviarlas al sistema.</li><li>▪ <b>Desconectado:</b> El dispositivo de equipo de cronometraje no se encuentra conectado al sistema.</li></ul> |   |  |

| Id                    |     | EditarEquipoCronometraje   |
|-----------------------|-----|--|
| Versión               |     | 1.0  |
| Actores               |     | Administrador.   |
| Descripción           |     | El <i>Administrador</i> puede editar el nombre de un de un equipo de cronometraje en el sistema.   |
| Pre-condición         |     | El <i>Administrador</i> debe estar autenticado en el sistema.  |
| Post-condición        |     | Se actualiza el nombre del equipo de cronometraje en el sistema.   |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Editar Equipo de Cronometraje</i> .  |
|                       | 2   | El sistema muestra un formulario para editar el equipo de cronometraje. Este formulario cuenta con un único campo: <i>Nombre del equipo de cronometraje</i> , que contiene el valor actual.  |
|                       | 3   | El <i>Administrador</i> modifica el nombre del equipo y pulsa <i>Guardar</i> .   |
|                       | 4   | El sistema comprueba que el dato introducido es válido.  |
|                       | 5   | El sistema actualiza la información del equipo de cronometraje en la base de datos.  |
|                       | 6   | El sistema redirige al <i>Administrador</i> a un panel donde muestra todos los datos del equipo de cronometraje editado.   |
| Flujos alternativos   | 3-B | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
|                       | 4-C | El sistema reconoce un error en el nombre del equipo de cronometraje, y muestra el formulario junto con un mensaje de error.   |
| Frecuencia de uso     |     | Baja.  |
| Requisitos especiales |     | Ninguno.   |
| Comentarios           |     | Se denomina <i>Editar</i> a este caso de uso a pesar de que el único valor que se está editando es el nombre. Esto es debido a que en próximas versiones existe la posibilidad de ofrecer más datos a editar por el <i>Administrador</i> en un equipo de cronometraje. |

| Id                    |     | EliminarEquipoCronometraje   |
|-----------------------|-----|--|
| Versión               |     | 1.0  |
| Actores               |     | Administrador.   |
| Descripción           |     | El <i>Administrador</i> puede eliminar un equipo de cronometraje del sistema.  |
| Pre-condición         |     | El <i>Administrador</i> debe estar autenticado en el sistema. Debe existir el equipo de cronometraje registrado en el sistema.   |
| Post-condición        |     | Se elimina el equipo de cronometraje del sistema.  |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Eliminar Equipo de Cronometraje</i> .  |
|                       | 2   | El sistema muestra un mensaje consultando al <i>Administrador</i> si está seguro de llevar a cabo la acción.                     |
|                       | 3   | El <i>Administrador</i> pulsa <i>Confirmar</i> .   |
|                       | 4   | El sistema elimina el equipo de cronometraje de la base de datos.  |
|                       | 5   | El sistema redirige al <i>Administrador</i> a una pantalla donde se muestran los equipos de cronometraje actuales en el sistema. |
| Flujos alternativos   | 3-B | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
| Frecuencia de uso     |     | Baja.  |
| Requisitos especiales |     | Ninguno.   |
| Comentarios           |     | No hay.  |

|                              |   |  |  |
|------------------------------|---|--|--|
| <b>Id</b>                    | <b>AñadirEventoDeportivo</b>  |  |  |
| <b>Versión</b>               | 1.0   |  |  |
| <b>Actores</b>               | <i>Administrador.</i>   |  |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede añadir un nuevo evento deportivo en el sistema.   |  |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.   |  |  |
| <b>Post-condición</b>        | Se crea un nuevo evento deportivo en el sistema.  |  |  |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona <i>Añadir Evento Deportivo</i> .  |  |
|                              | 2   | El sistema muestra un formulario cuyos campos a rellenar son los datos básicos del evento (nombre, fecha, lugar, etc.)         |  |
|                              | 3   | El <i>Administrador</i> rellena los campos y pulsa <i>Guardar</i> .  |  |
|                              | 4   | El sistema comprueba que los datos introducidos son válidos. Si los datos son válidos, almacena el evento en la base de datos. |  |
|                              | 5   | El sistema muestra al <i>Administrador</i> una pantalla con los datos del evento que acaba de añadir.                          |  |
| <b>Flujos alternativos</b>   | 3-B   | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |  |
|                              | 4-C   | El sistema comprueba que los datos no son válidos, y muestra de nuevo el formulario con mensajes de error.                     |  |
| <b>Frecuencia de uso</b>     | Media.  |  |  |
| <b>Requisitos especiales</b> | Ninguno.  |  |  |
| <b>Comentarios</b>           | La fecha del evento deportivo introducida a la hora de añadir un nuevo evento es de carácter orientativo, sin que sea necesario que el evento deportivo comience exactamente en esa fecha y hora. |  |  |



|                              |  |  |  |
|------------------------------|--|--|--|
| <b>Id</b>                    | <b>VerEventoDeportivo</b>  |  |  |
| <b>Versión</b>               | 1.0  |  |  |
| <b>Actores</b>               | <i>Administrador.</i>  |  |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede acceder a una pantalla en la que se muestra la información general de un evento deportivo, además de las opciones que permiten gestionar los detalles del evento. Estas opciones son: editar la información general del evento, gestionar sus recorridos, categorías, participantes, comenzar y finalizar el evento.   |  |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.  |  |  |
| <b>Post-condición</b>        | Los datos del evento, así como las opciones del mismo han sido mostrados en pantalla.  |  |  |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> selecciona un evento deportivo pulsando en el enlace del mismo.            |  |
|                              | 2  | El sistema comprueba que el Evento Deportivo existe.   |  |
|                              | 3  | El sistema muestra toda la información y opciones del evento deportivo, así como su estado actual. |  |
| <b>Flujos alternativos</b>   | 3-B  | El sistema no encuentra el evento deportivo y muestra un mensaje de error.                         |  |
| <b>Frecuencia de uso</b>     | Alta.  |  |  |
| <b>Requisitos especiales</b> | Ninguno.   |  |  |
| <b>Comentarios</b>           | <p>Uno de los parámetros de un evento deportivo es el estado. El estado se genera automáticamente en función de si el evento ha comenzado y finalizado. Puede tener tres valores:</p> <ul style="list-style-type: none"><li>▪ <b>Por comenzar:</b> El evento no ha comenzado.</li><li>▪ <b>En directo:</b> El evento ha comenzado, pero no ha finalizado.</li><li>▪ <b>Finalizado:</b> El evento ha comenzado y ha finalizado.</li></ul> |  |  |

|                              |  |   |  |
|------------------------------|--|---|--|
| <b>Id</b>                    | <b>EditarEventoDeportivo</b>   |   |  |
| <b>Versión</b>               | 1.0  |   |  |
| <b>Actores</b>               | <i>Administrador.</i>  |   |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede editar los datos básicos de un evento deportivo en el sistema (nombre, fecha, descripción, etc.) |   |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.  |   |  |
| <b>Post-condición</b>        | Se actualizan los datos básicos de un evento deportivo en el sistema.  |   |  |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> selecciona <i>Editar Evento Deportivo</i> .   |  |
|                              | 2  | El sistema muestra un formulario para editar el evento deportivo. Este formulario cuenta con los campos de información básica, y contiene los valores actuales. |  |
|                              | 3  | El <i>Administrador</i> modifica los campos del formulario y pulsa <i>Guardar</i> .   |  |
|                              | 4  | El sistema comprueba que la información introducida es válida.  |  |
|                              | 5  | El sistema actualiza la información del evento deportivo en la base de datos.   |  |
|                              | 6  | El sistema redirige al <i>Administrador</i> a un panel de <i>VerEventoDeportivo</i> .   |  |
| <b>Flujos alternativos</b>   | 3-B  | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |  |
|                              | 4-C  | El sistema reconoce un error en la información introducida y muestra de nuevo el formulario con los mensajes de error pertinentes.                              |  |
| <b>Frecuencia de uso</b>     | Media.   |   |  |
| <b>Requisitos especiales</b> | Ninguno.   |   |  |
| <b>Comentarios</b>           | No hay.  |   |  |

|                              |  |   |  |
|------------------------------|--|---|--|
| <b>Id</b>                    | <b>EliminarEventoDeportivo</b>   |   |  |
| <b>Versión</b>               | 1.0  |   |  |
| <b>Actores</b>               | <i>Administrador.</i>  |   |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede eliminar un evento deportivo del sistema.  |   |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.  |   |  |
| <b>Post-condición</b>        | El sistema elimina el evento deportivo, así como toda aquella información que depende del evento deportivo: recorridos, puntos de control, categorías, participantes y envíos de cronometraje. |   |  |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> selecciona <i>Eliminar Evento Deportivo</i> .   |  |
|                              | 2  | El sistema muestra un mensaje consultando al <i>Administrador</i> si está seguro de llevar a cabo la acción.                |  |
|                              | 3  | El <i>Administrador</i> pulsa <i>Confirmar</i> .  |  |
|                              | 4  | El sistema elimina el evento deportivo de la base de datos, así como toda la información que depende del mismo.             |  |
|                              | 5  | El sistema redirige al <i>Administrador</i> a una pantalla donde se muestran los eventos deportivos actuales en el sistema. |  |
| <b>Flujos alternativos</b>   | 3-B  | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |  |
| <b>Frecuencia de uso</b>     | Baja.  |   |  |
| <b>Requisitos especiales</b> | Ninguno.   |   |  |
| <b>Comentarios</b>           | No hay.  |   |  |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>ComenzarEventoDeportivo</b>  |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede dar comienzo a un evento deportivo registrado en el sistema.  |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.   |   |
| <b>Post-condición</b>        | El sistema establece el estado del evento deportivo como comenzado y el instante de comienzo en el momento actual. El sistema comienza a aceptar datos procedentes de los equipos de cronometraje una vez haya pasado el tiempo mínimo de medición desde el instante de comienzo del evento deportivo. El sistema permite el acceso a la pantalla de seguimiento del evento en directo y clasificaciones. |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> pulsa el botón <i>Comenzar Evento</i> , desde la pantalla de <i>VerEventoDeportivo</i> .  |
|                              | 2   | El sistema comprueba que ninguno de los equipos de cronometraje configurados para el evento deportivo se encuentra en uso en eventos deportivos que se estén realizando en directo. |
|                              | 3   | El sistema establece el evento como comenzado, y devuelve al usuario a la pantalla de <i>VerEventoDeportivo</i> .   |
| <b>Flujos alternativos</b>   |   |   |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | No hay.   |   |

|                              |  |  |
|------------------------------|--|--|
| <b>Id</b>                    | <b>FinalizarEventoDeportivo</b>  |  |
| <b>Versión</b>               | 1.0  |  |
| <b>Actores</b>               | <i>Administrador.</i>  |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede finalizar un evento deportivo registrado en el sistema.  |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema. El evento deportivo debe haber comenzado.  |  |
| <b>Post-condición</b>        | El sistema establece el estado del evento deportivo como finalizado. El sistema deja de aceptar datos procedentes de los equipos de cronometraje. El sistema muestra en la pantalla de seguimiento del evento en directo y clasificaciones un mensaje de que el evento ha terminado y los tiempos son definitivos. |  |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> pulsa el botón <i>Finalizar Evento</i> , desde la pantalla de <i>VerEventoDeportivo</i> .  |
|                              | 2  | El sistema establece el evento como finalizado, y devuelve al usuario a la pantalla de <i>VerEventoDeportivo</i> . |
| <b>Flujos alternativos</b>   |  |  |
| <b>Frecuencia de uso</b>     | Media.   |  |
| <b>Requisitos especiales</b> | Ninguno.   |  |
| <b>Comentarios</b>           | No hay.  |  |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>AñadirRecorrido</b>  |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede añadir un nuevo recorrido en el sistema, que pertenece a un evento deportivo previamente registrado.  |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema. Debe existir un evento deportivo previamente registrado en el sistema. El evento deportivo para el cual se va a añadir el recorrido no puede haber comenzado. |   |
| <b>Post-condición</b>        | Se crea un nuevo recorrido en el sistema.   |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona <i>Añadir Recorrido</i> dentro del panel de un evento deportivo existente.                     |
|                              | 2   | El sistema muestra un formulario con los campos del recorrido.  |
|                              | 3   | El <i>Administrador</i> rellena los campos y pulsa <i>Guardar</i> .   |
|                              | 4   | El sistema comprueba que los datos introducidos son válidos. Si los datos son válidos, almacena el recorrido en la base de datos. |
|                              | 5   | El sistema muestra al <i>Administrador</i> una pantalla con los datos del recorrido que acaba de añadir.                          |
| <b>Flujos alternativos</b>   | 3-B   | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |
|                              | 4-C   | El sistema comprueba que los datos no son válidos, y muestra de nuevo el formulario con mensajes de error.                        |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | No hay.   |   |

|                              |   |  |
|------------------------------|---|--|
| <b>Id</b>                    | <b>VerRecorrido</b>   |  |
| <b>Versión</b>               | 1.0   |  |
| <b>Actores</b>               | <i>Administrador.</i>   |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede acceder a una pantalla con la información de un recorrido y sus puntos de control.  |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.   |  |
| <b>Post-condición</b>        | Los datos del recorrido y sus puntos de control, así como las opciones del mismo han sido mostrados en pantalla.  |  |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona un recorrido que pertenece a un evento deportivo pulsando en el enlace del mismo. |
|                              | 2   | El sistema comprueba que el recorrido existe.  |
|                              | 3   | El sistema muestra toda la información y opciones del recorrido, así como sus puntos de control.                     |
| <b>Flujos alternativos</b>   | 3-B   | El sistema no encuentra el recorrido y muestra un mensaje de error.  |
| <b>Frecuencia de uso</b>     | Media.  |  |
| <b>Requisitos especiales</b> | Ninguno.  |  |
| <b>Comentarios</b>           | La información que se muestra en la pantalla no es únicamente la que se introduce en el formulario de <i>AñadirRecorrido</i> , sino que también se muestran los puntos de control que se han configurado ordenados por punto kilométrico. |  |

| Id                    | EditarRecorrido   |  |
|-----------------------|---|--|
| Versión               | 1.0   |  |
| Actores               | <i>Administrador.</i>   |  |
| Descripción           | El <i>Administrador</i> puede editar los datos de un recorrido en el sistema. |  |
| Pre-condición         | El <i>Administrador</i> debe estar autenticado en el sistema.                 |  |
| Post-condición        | Se actualizan los datos de un recorrido en el sistema.                        |  |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Editar Recorrido</i> .   |
|                       | 2   | El sistema muestra un formulario para editar el recorrido. Este formulario cuenta con los mismos campos de información que se emplean a la hora de añadir un recorrido, y contiene los valores actuales. |
|                       | 3   | El <i>Administrador</i> modifica los campos del formulario y pulsa <i>Guardar</i> .  |
|                       | 4   | El sistema comprueba que la información introducida es válida.   |
|                       | 5   | El sistema actualiza la información del recorrido en la base de datos.   |
|                       | 6   | El sistema redirige al <i>Administrador</i> a un panel de <i>VerRecorrido</i> .  |
| Flujos alternativos   | 3-B   | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
|                       | 4-C   | El sistema reconoce un error en la información introducida y muestra de nuevo el formulario con los mensajes de error pertinentes.   |
| Frecuencia de uso     | Baja.   |  |
| Requisitos especiales | Ninguno.  |  |
| Comentarios           | No hay.   |  |

| Id                    | EliminarRecorrido  |  |
|-----------------------|--|--|
| Versión               | 1.0  |  |
| Actores               | <i>Administrador.</i>  |  |
| Descripción           | El <i>Administrador</i> puede eliminar un recorrido del sistema.                             |  |
| Pre-condición         | El <i>Administrador</i> debe estar autenticado en el sistema.                                |  |
| Post-condición        | El sistema elimina el recorrido, así como los puntos de control que pertenecen al recorrido. |  |
| Flujo Normal          | 1  | El <i>Administrador</i> selecciona <i>Eliminar Recorrido</i> .   |
|                       | 2  | El sistema muestra un mensaje consultando al <i>Administrador</i> si está seguro de llevar a cabo la acción.                                   |
|                       | 3  | El <i>Administrador</i> pulsa <i>Confirmar</i> .   |
|                       | 4  | El sistema elimina el recorrido de la base de datos, así como los puntos de control que pertenecen al recorrido.                               |
|                       | 5  | El sistema redirige al <i>Administrador</i> a una pantalla donde se muestran los recorridos actuales que pertenecen al mismo evento deportivo. |
| Flujos alternativos   | 3-B  | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
| Frecuencia de uso     | Baja.  |  |
| Requisitos especiales | Ninguno.   |  |
| Comentarios           | No hay.  |  |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>AñadirPuntoDeControl</b>   |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede añadir un nuevo punto de control en el sistema, que pertenece a un recorrido previamente registrado.  |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema. Debe existir un recorrido previamente registrado en el sistema. Debe existir al menos un equipo de cronometraje registrado en el sistema. El evento deportivo para el cual se va a añadir el punto de control no puede haber comenzado. |   |
| <b>Post-condición</b>        | Se crea un nuevo punto de control en el sistema.  |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona <i>Añadir Punto de Control</i> dentro del panel de un recorrido existente.   |
|                              | 2   | El sistema muestra un formulario con los campos del punto de control.   |
|                              | 3   | El <i>Administrador</i> rellena los campos y pulsa <i>Guardar</i> .   |
|                              | 4   | El sistema comprueba que los datos introducidos son válidos. La validación de mayor importancia se refiere al equipo de cronometraje asignado al punto de control, ya que el sistema no debe permitir que un mismo equipo de cronometraje se configure para más de un punto de control en un mismo recorrido. Si los datos son válidos, almacena el punto de control en la base de datos. |
|                              | 5   | El sistema muestra al <i>Administrador</i> una pantalla con la información del recorrido al que pertenece el punto de control añadido.  |
| <b>Flujos alternativos</b>   | 3-B   | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |
|                              | 4-C   | El sistema comprueba que los datos no son válidos, y muestra de nuevo el formulario con mensajes de error.  |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | En los campos del formulario que se deben rellenar, un campo designa el equipo de cronometraje que se encargará de tomar los tiempos en ese punto de control. En este campo, aparecerán como opciones los equipos de cronometraje que se encuentren registrados en el sistema.                        |   |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>VerPuntoDeControl</b>  |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede acceder a una pantalla con la información de un punto de control. |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.                                   |   |
| <b>Post-condición</b>        | Los datos del punto de control han sido mostrados en pantalla.                                  |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona un punto de control que pertenece a un recorrido pulsando en el enlace del mismo.. |
|                              | 2   | El sistema comprueba que el punto de control existe.  |
|                              | 3   | El sistema muestra toda la información y opciones del punto de control.   |
| <b>Flujos alternativos</b>   | 3-B   | El sistema no encuentra el punto de control y muestra un mensaje de error.  |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | No hay.   |   |

|                              |  |  |  |
|------------------------------|--|--|--|
| <b>Id</b>                    | <b>EditarPuntoDeControl</b>  |  |  |
| <b>Versión</b>               | 1.0  |  |  |
| <b>Actores</b>               | <i>Administrador.</i>  |  |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede editar los datos de un punto de control en el sistema. |  |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.                        |  |  |
| <b>Post-condición</b>        | Se actualizan los datos de un punto de control en el sistema.                        |  |  |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> selecciona <i>Editar Punto de Control</i> .  |  |
|                              | 2  | El sistema muestra un formulario para editar el punto de control. Este formulario contiene los valores actuales.   |  |
|                              | 3  | El <i>Administrador</i> modifica los campos del formulario y pulsa <i>Guardar</i> .  |  |
|                              | 4  | El sistema comprueba que la información introducida es válida. La validación de mayor importancia se refiere al equipo de cronometraje asignado al punto de control, el sistema no debe permitir que un mismo equipo de cronometraje se configure para más de un punto de control en un mismo recorrido. |  |
|                              | 5  | El sistema actualiza la información del punto de control en la base de datos.  |  |
|                              | 6  | El sistema redirige al <i>Administrador</i> a un panel de <i>VerRecorrido</i> , que contiene toda la información del recorrido al que pertenece el punto de control.   |  |
| <b>Flujos alternativos</b>   | 3-B  | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |  |
|                              | 4-C  | El sistema reconoce un error en la información introducida y muestra de nuevo el formulario con los mensajes de error pertinentes.   |  |
| <b>Frecuencia de uso</b>     | Baja.  |  |  |
| <b>Requisitos especiales</b> | Ninguno.   |  |  |
| <b>Comentarios</b>           | No hay.  |  |  |

|                              |   |   |  |
|------------------------------|---|---|--|
| <b>Id</b>                    | <b>EliminarPuntoDeControl</b>   |   |  |
| <b>Versión</b>               | 1.0   |   |  |
| <b>Actores</b>               | <i>Administrador.</i>   |   |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede eliminar un punto de control del sistema. |   |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.           |   |  |
| <b>Post-condición</b>        | El sistema elimina el punto de control.                                 |   |  |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona <i>Eliminar Punto de Control</i> .   |  |
|                              | 2   | El sistema muestra un mensaje consultando al <i>Administrador</i> si está seguro de llevar a cabo la acción.                                    |  |
|                              | 3   | El <i>Administrador</i> pulsa <i>Confirmar</i> .  |  |
|                              | 4   | El sistema elimina el punto de control de la base de datos.   |  |
|                              | 5   | El sistema redirige al <i>Administrador</i> a una pantalla donde se muestra la información del recorrido al que pertenecía el punto de control. |  |
| <b>Flujos alternativos</b>   | 3-B   | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |  |
| <b>Frecuencia de uso</b>     | Baja.   |   |  |
| <b>Requisitos especiales</b> | Ninguno.  |   |  |
| <b>Comentarios</b>           | No hay.   |   |  |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>AñadirCategoria</b>  |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede añadir una nueva categoría en el sistema, que pertenece a un evento deportivo previamente registrado.   |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema. Debe existir un evento deportivo previamente registrado en el sistema. Debe existir al menos un recorrido previamente registrado en el sistema para el evento deportivo al cual se va a añadir la categoría. El evento deportivo para el cual se va a añadir la categoría no puede haber comenzado. |   |
| <b>Post-condición</b>        | Se crea una nueva categoría en el sistema.  |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona <i>Añadir Categoría</i> dentro del panel de un evento deportivo existente.                     |
|                              | 2   | El sistema muestra un formulario con los campos de la categoría.  |
|                              | 3   | El <i>Administrador</i> rellena los campos y pulsa <i>Guardar</i> .   |
|                              | 4   | El sistema comprueba que los datos introducidos son válidos. Si los datos son válidos, almacena la categoría en la base de datos. |
|                              | 5   | El sistema muestra al <i>Administrador</i> una pantalla con la información de la categoría que acaba de añadirse.                 |
| <b>Flujos alternativos</b>   | 3-B   | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |
|                              | 4-C   | El sistema comprueba que los datos no son válidos, y muestra de nuevo el formulario con mensajes de error.                        |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | Uno de los campos del formulario que deberá ser rellenado es el recorrido a realizar por los participantes de la categoría. El sistema mostrará como opciones disponibles los recorridos previamente registrados en el evento deportivo al que pertenece la categoría.  |   |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>VerCategoria</b>   |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede acceder a una pantalla con la información de una categoría. |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.                             |   |
| <b>Post-condición</b>        | Los datos de la categoría han sido mostrados en pantalla.                                 |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona una categoría pulsando en el enlace de la misma. |
|                              | 2   | El sistema comprueba que la categoría existe.                                       |
|                              | 3   | El sistema muestra toda la información y opciones de la categoría.                  |
| <b>Flujos alternativos</b>   | 3-B   | El sistema no encuentra la categoría y muestra un mensaje de error.                 |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | No hay.   |   |



| Id                    |     | EditarCategoría  |
|-----------------------|-----|--|
| Versión               |     | 1.0  |
| Actores               |     | Administrador.   |
| Descripción           |     | El <i>Administrador</i> puede editar los datos de una categoría en el sistema.   |
| Pre-condición         |     | El <i>Administrador</i> debe estar autenticado en el sistema.  |
| Post-condición        |     | Se actualizan los datos de una categoría en el sistema.  |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Editar Categoría</i> .   |
|                       | 2   | El sistema muestra un formulario para editar la categoría. Este formulario contiene los valores actuales.                          |
|                       | 3   | El <i>Administrador</i> modifica los campos del formulario y pulsa <i>Guardar</i> .  |
|                       | 4   | El sistema comprueba que la información introducida es válida.   |
|                       | 5   | El sistema actualiza la información de la categoría en la base de datos.   |
|                       | 6   | El sistema redirige al <i>Administrador</i> a un panel de <i>VerCategoría</i> .  |
| Flujos alternativos   | 3-B | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
|                       | 4-C | El sistema reconoce un error en la información introducida y muestra de nuevo el formulario con los mensajes de error pertinentes. |
| Frecuencia de uso     |     | Baja.  |
| Requisitos especiales |     | Ninguno.   |
| Comentarios           |     | No hay.  |

| Id                    |     | EliminarCategoría   |
|-----------------------|-----|---|
| Versión               |     | 1.0   |
| Actores               |     | Administrador.  |
| Descripción           |     | El <i>Administrador</i> puede eliminar una categoría del sistema.   |
| Pre-condición         |     | El <i>Administrador</i> debe estar autenticado en el sistema.   |
| Post-condición        |     | El sistema elimina la categoría.  |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Eliminar Categoría</i> .  |
|                       | 2   | El sistema muestra un mensaje consultando al <i>Administrador</i> si está seguro de llevar a cabo la acción.  |
|                       | 3   | El <i>Administrador</i> pulsa <i>Confirmar</i> .  |
|                       | 4   | El sistema elimina la categoría de la base de datos.  |
|                       | 5   | El sistema redirige al <i>Administrador</i> a una pantalla donde se muestra la lista de categorías que existen en el evento deportivo al que pertenecía la categoría eliminada. |
| Flujos alternativos   | 3-B | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |
| Frecuencia de uso     |     | Baja.   |
| Requisitos especiales |     | Ninguno.  |
| Comentarios           |     | No hay.   |

|                              |  |  |
|------------------------------|--|--|
| <b>Id</b>                    | <b>AñadirParticipante</b>  |  |
| <b>Versión</b>               | 1.0  |  |
| <b>Actores</b>               | <i>Administrador.</i>  |  |
| <b>Descripción</b>           | El <i>Administrador</i> puede añadir un nuevo participante en el sistema, que pertenece a un evento deportivo previamente registrado.  |  |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema. Debe existir un evento deportivo previamente registrado en el sistema. Debe existir al menos una categoría previamente registrada en el sistema para el evento deportivo al cual se va a añadir el participante. El evento deportivo para el cual se va a añadir el participante no puede haber comenzado. |  |
| <b>Post-condición</b>        | Se crea un nuevo participante en el sistema.   |  |
| <b>Flujo Normal</b>          | 1  | El <i>Administrador</i> selecciona <i>Añadir Participante</i> dentro del panel de un evento deportivo existente.                     |
|                              | 2  | El sistema muestra un formulario con los campos del participante.  |
|                              | 3  | El <i>Administrador</i> rellena los campos y pulsa <i>Guardar</i> .  |
|                              | 4  | El sistema comprueba que los datos introducidos son válidos. Si los datos son válidos, almacena el participante en la base de datos. |
|                              | 5  | El sistema muestra al <i>Administrador</i> una pantalla con la información del participante que acaba de añadir.                     |
| <b>Flujos alternativos</b>   | 3-B  | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
|                              | 4-C  | El sistema comprueba que los datos no son válidos, y muestra de nuevo el formulario con mensajes de error.                           |
| <b>Frecuencia de uso</b>     | Alta.  |  |
| <b>Requisitos especiales</b> | Ninguno.   |  |
| <b>Comentarios</b>           | Uno de los campos del formulario que deberá ser rellenado es la categoría a la que va a pertenecer el participante. El sistema mostrará como opciones disponibles las categorías previamente registradas dentro del mismo evento deportivo.  |  |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>VerParticipante</b>  |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Administrador.</i>   |   |
| <b>Descripción</b>           | El <i>Administrador</i> puede acceder a una pantalla con la información de un participante. |   |
| <b>Pre-condición</b>         | El <i>Administrador</i> debe estar autenticado en el sistema.                               |   |
| <b>Post-condición</b>        | Los datos del participante han sido mostrados en pantalla.                                  |   |
| <b>Flujo Normal</b>          | 1   | El <i>Administrador</i> selecciona un participante pulsando en el enlace del mismo. |
|                              | 2   | El sistema comprueba que el participante existe.                                    |
|                              | 3   | El sistema muestra toda la información y opciones del participante.                 |
| <b>Flujos alternativos</b>   | 3-B   | El sistema no encuentra el participante y muestra un mensaje de error.              |
| <b>Frecuencia de uso</b>     | Media.  |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | No hay.   |   |

| Id                    |     | EditarParticipante   |
|-----------------------|-----|--|
| Versión               |     | 1.0  |
| Actores               |     | Administrador.   |
| Descripción           |     | El <i>Administrador</i> puede editar los datos de un participante en el sistema.   |
| Pre-condición         |     | El <i>Administrador</i> debe estar autenticado en el sistema.  |
| Post-condición        |     | Se actualizan los datos de un participante en el sistema.  |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Editar Participante</i> .  |
|                       | 2   | El sistema muestra un formulario para editar el participante. Este formulario contiene los valores actuales.                       |
|                       | 3   | El <i>Administrador</i> modifica los campos del formulario y pulsa <i>Guardar</i> .  |
|                       | 4   | El sistema comprueba que la información introducida es válida.   |
|                       | 5   | El sistema actualiza la información del participante en la base de datos.  |
|                       | 6   | El sistema redirige al <i>Administrador</i> a un panel de <i>VerParticipante</i> .   |
| Flujos alternativos   | 3-B | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.   |
|                       | 4-C | El sistema reconoce un error en la información introducida y muestra de nuevo el formulario con los mensajes de error pertinentes. |
| Frecuencia de uso     |     | Baja.  |
| Requisitos especiales |     | Ninguno.   |
| Comentarios           |     | No hay.  |

| Id                    |     | EliminarParticipante  |
|-----------------------|-----|---|
| Versión               |     | 1.0   |
| Actores               |     | Administrador.  |
| Descripción           |     | El <i>Administrador</i> puede eliminar un participante del sistema.   |
| Pre-condición         |     | El <i>Administrador</i> debe estar autenticado en el sistema.   |
| Post-condición        |     | El sistema elimina el participante.   |
| Flujo Normal          | 1   | El <i>Administrador</i> selecciona <i>Eliminar Participante</i> .   |
|                       | 2   | El sistema muestra un mensaje consultando al <i>Administrador</i> si está seguro de llevar a cabo la acción.  |
|                       | 3   | El <i>Administrador</i> pulsa <i>Confirmar</i> .  |
|                       | 4   | El sistema elimina el participante de la base de datos.   |
|                       | 5   | El sistema redirige al <i>Administrador</i> a una pantalla donde se muestra la lista de participantes que existen en el evento deportivo al que pertenecía el participante eliminado. |
| Flujos alternativos   | 3-B | El <i>Administrador</i> pulsa <i>Cancelar</i> . El sistema permanece inalterado.  |
| Frecuencia de uso     |     | Baja.   |
| Requisitos especiales |     | Ninguno.  |
| Comentarios           |     | No hay.   |

|                              |  |  |
|------------------------------|--|--|
| <b>Id</b>                    | <b>VerEventoDeportivoPublico</b>   |  |
| <b>Versión</b>               | 1.0  |  |
| <b>Actores</b>               | <i>Visitante.</i>  |  |
| <b>Descripción</b>           | El <i>Visitante</i> puede acceder a una pantalla con la información pública de un evento deportivo, además de las subsecciones que permiten seguir el avance del evento deportivo si este se encuentra en directo, o ver los resultados finales si ya ha finalizado. |  |
| <b>Pre-condición</b>         | Ninguna.   |  |
| <b>Post-condición</b>        | Ninguna.   |  |
| <b>Flujo Normal</b>          | 1  | El <i>Visitante</i> accede a través de una URL a la pantalla de un evento deportivo. |
|                              | 2  | El sistema comprueba que el evento deportivo existe.                                 |
|                              | 3  | El sistema muestra la información del evento deportivo, así como su estado actual.   |
| <b>Flujos alternativos</b>   | 3-B  | El sistema no encuentra el evento deportivo y muestra un mensaje de error.           |
| <b>Frecuencia de uso</b>     | Alta.  |  |
| <b>Requisitos especiales</b> | Ninguno.   |  |
| <b>Comentarios</b>           | No hay.  |  |

|                              |   |   |
|------------------------------|---|---|
| <b>Id</b>                    | <b>SeguirEventoDeportivoPublico</b>   |   |
| <b>Versión</b>               | 1.0   |   |
| <b>Actores</b>               | <i>Visitante.</i>   |   |
| <b>Descripción</b>           | El <i>Visitante</i> puede acceder a una pantalla que muestra el seguimiento en directo de un evento deportivo, a través de una tabla en la que apareceren los participantes de una categoría ordenados por tiempo.                  |   |
| <b>Pre-condición</b>         | El evento deportivo debe haber comenzado.   |   |
| <b>Post-condición</b>        | Ninguna.  |   |
| <b>Flujo Normal</b>          | 1   | El <i>Visitante</i> accede a través de una URL al seguimiento de una categoría de un evento deportivo tras haber pulsado en uno de los enlaces que aparecen en la lista dentro de la vista de <i>VerEventoDeportivoPublico</i> .  |
|                              | 2   | El sistema realiza las siguientes acciones con el objetivo de generar la tabla de seguimiento: <ol style="list-style-type: none"> <li>1. Comprueba que existe la categoría accedida.</li> <li>2. Obtiene el recorrido vinculado a la categoría y los puntos de control del recorrido.</li> <li>3. Obtiene todos los envíos de cronometraje correspondientes.</li> <li>4. Obtiene los participantes vinculados a los envíos de cronometraje.</li> <li>5. Genera una clasificación, ordenando los participantes por tiempos obtenidos en el último punto del control en el que se hayan obtenido envíos de cronometraje.</li> </ol> |
|                              | 3   | El sistema muestra la tabla de seguimiento (clasificación en directo, o clasificación final si ya ha finalizado el evento).   |
| <b>Flujos alternativos</b>   | 3-B   | El sistema no encuentra la categoría y muestra un mensaje de error.   |
|                              | 3-C   | El sistema no encuentra envíos de cronometraje y muestra un mensaje informando de que ningún participante ha obtenido tiempos todavía.  |
| <b>Frecuencia de uso</b>     | Alta.   |   |
| <b>Requisitos especiales</b> | Ninguno.  |   |
| <b>Comentarios</b>           | Esta pantalla estará activa una vez que haya comenzado el evento. Si el evento se encuentra en directo, mostrará un mensaje indicativo, de la misma forma que si el evento ya ha finalizado mostrará también un mensaje indicativo. |   |

|                              |   |  |
|------------------------------|---|--|
| <b>Id</b>                    | <b>CrearEnvioCronometraje</b>   |  |
| <b>Versión</b>               | 1.0   |  |
| <b>Actores</b>               | <i>EquipoCronometraje</i> .   |  |
| <b>Descripción</b>           | El <i>EquipoCronometraje</i> puede transmitir envíos de cronometraje en formato JSON que contendrán los valores: EPC de una etiqueta RFID y marca de tiempo.  |  |
| <b>Pre-condición</b>         | El <i>EquipoCronometraje</i> debe estar registrado en el sistema. El evento deportivo debe haber comenzado o, se ha seleccionado leer etiqueta RFID desde el panel de administración a la hora de añadir un participante. |  |
| <b>Post-condición</b>        | El envío de cronometraje se ha almacenado en la base de datos.  |  |
| <b>Flujo Normal</b>          | 1   | El <i>EquipoCronometraje</i> realiza un <i>HTTP Request</i> en formato JSON a una URL del sistema, que contiene el EPC de una etiqueta RFID y una marca de tiempo.   |
|                              | 2   | El sistema comprueba si el envío de cronometraje debe ser aceptado o no.   |
|                              | 3   | El envío de cronometraje es aceptado y almacenado en la base de datos, el sistema responde con un <i>HTTP Response</i> en formato JSON indicando que se ha aceptado. |
| <b>Flujos alternativos</b>   | 3-B   | El envío de cronometraje es descartado, el sistema responde con un <i>HTTP Response</i> en formato JSON indicando que el envío no se ha aceptado.                    |
| <b>Frecuencia de uso</b>     | Alta.   |  |
| <b>Requisitos especiales</b> | Ninguno.  |  |
| <b>Comentarios</b>           | La complejidad del sistema a la hora de determinar si un envío de cronometraje debe ser aceptado o no se analiza en la sección <i>Desarrollo de la lógica y control del sistema de cronometraje</i> .                     |  |

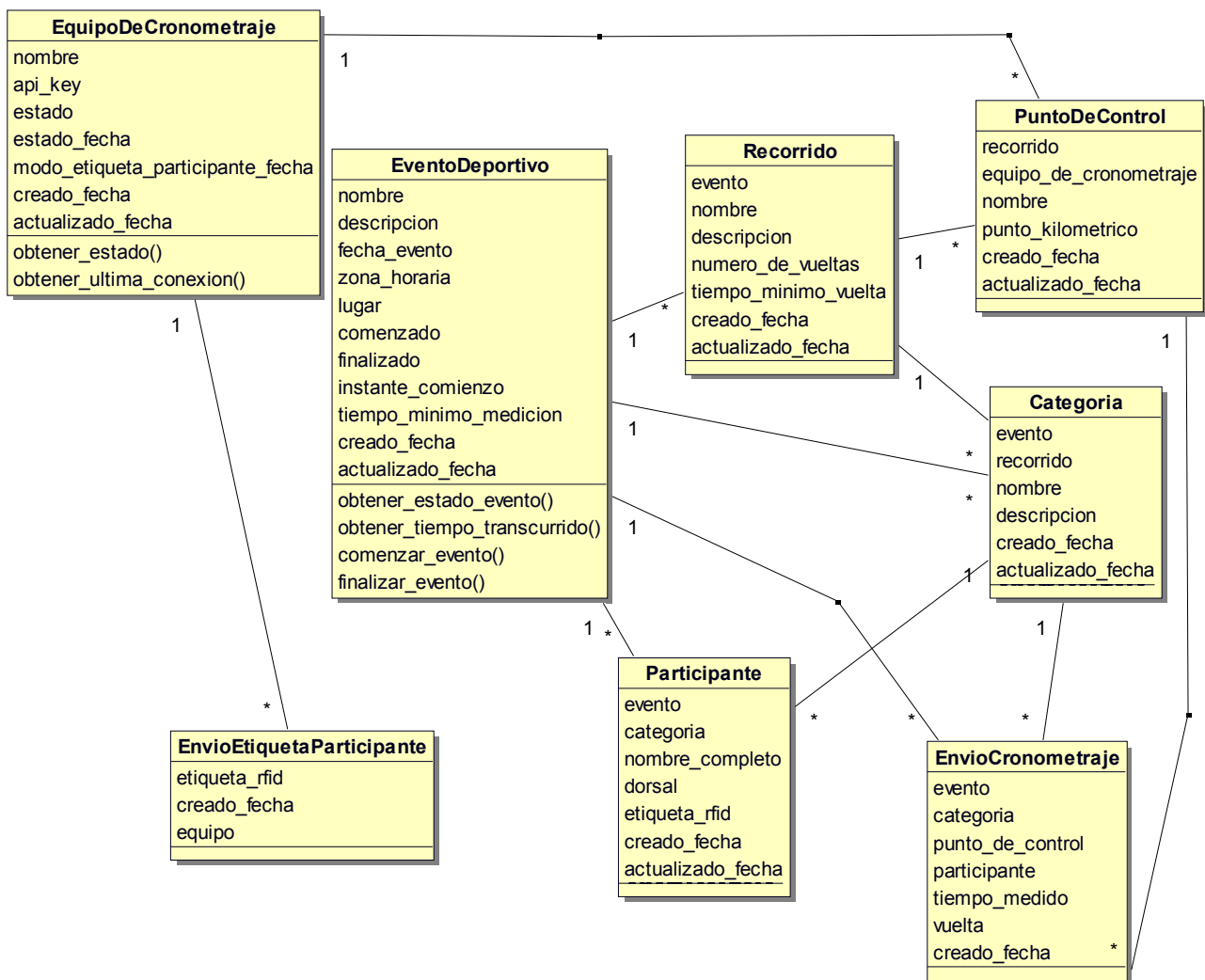
|                              |  |  |
|------------------------------|--|--|
| <b>Id</b>                    | <b>ActualizarEstadoEquipoCronometraje</b>                                      |  |
| <b>Versión</b>               | 1.0  |  |
| <b>Actores</b>               | <i>EquipoCronometraje</i> .  |  |
| <b>Descripción</b>           | El <i>EquipoCronometraje</i> puede transmitir su estado en formato JSON.       |  |
| <b>Pre-condición</b>         | El <i>EquipoCronometraje</i> debe estar registrado en el sistema.              |  |
| <b>Post-condición</b>        | El estado del <i>EquipoCronometraje</i> se ha actualizado en la base de datos. |  |
| <b>Flujo Normal</b>          | 1  | El <i>EquipoCronometraje</i> realiza un <i>HTTP Request</i> en formato JSON a una URL del sistema, que contiene su estado actual.  |
|                              | 2  | El sistema comprueba que el <i>EquipoCronometraje</i> existe, el estado es actualizado en el sistema, el sistema responde con un <i>HTTP Response</i> en formato JSON indicando que se ha actualizado. |
| <b>Flujos alternativos</b>   | 2-B  | El sistema detecta que el <i>EquipoCronometraje</i> no está registrado y responde con un <i>HTTP Response</i> en formato JSON indicando que no se ha aceptado.   |
| <b>Frecuencia de uso</b>     | Alta.  |  |
| <b>Requisitos especiales</b> | Ninguno.   |  |
| <b>Comentarios</b>           | No hay.  |  |

## 5.4. Diagrama de clases

La aplicación *web* se ha desarrollado empleando el framework *Django* que sigue el patrón Modelo-Vista-Plantilla, este patrón ya ha sido analizado en el capítulo 2. Cada modelo al igual que cada vista se define con una clase, en total la aplicación *web* desarrollada se compone de 8 modelos y 54 vistas.

*Django* proporciona clases predefinidas para modelos y vistas que facilitan y permiten el desarrollo rápido mediante la herencia de estas clases. Todos los modelos implementados heredan los métodos y propiedades de la clase *Model*. Las vistas implementadas heredan los métodos y propiedades de diferentes clases que incorpora el *framework*: *TemplateView*, *DetailView*, *CreateView*, etc.

Se presenta a continuación el diagrama de clases correspondiente a los modelos de datos, siendo los modelos de datos las clases que mejor representan a un sistema debido a su estrecha relación con la estructura de la información en la base de datos.



## 5.5. Desarrollo de la lógica y control del sistema de cronometraje

Gran parte del desarrollo de la aplicación *web* consiste en operaciones habituales de una aplicación *web* al uso, como son obtener información de una base de datos, introducir información en una base de datos, validar formularios, etc.

Sin embargo, existen partes de la aplicación *web* que revisten de mayor complejidad, y además, componen las partes fundamentales del sistema de cronometraje. Se trata de la parte lógica y de control que componen las operaciones correspondientes a la decisión de aceptar o no los envíos de cronometraje procedentes de los equipos de cronometraje y, las operaciones que se encargan de generar la clasificación con la tabla de tiempos.

### 5.5.1. Operaciones para determinar la validez de los envíos de cronometraje

Estas operaciones corresponden a la vista *EnvioDatosCronometrajeView*. Dicha vista recibe peticiones HTTP en formato JSON cuyos datos se componen de un identificador de participante (EPC de la etiqueta RFID) y una marca de tiempo. Además, en la cabecera de la petición HTTP debe figurar un identificador y una *API KEY* del equipo de cronometraje del que procede la petición HTTP.

La primera comprobación que se realiza es determinar si el equipo de cronometraje es válido o no, es decir, si está registrado en el sistema y si sus credenciales (identificador y *API KEY*) son correctas. En caso de fallar esta comprobación se descarta el envío de cronometraje.

La segunda comprobación es determinar si el envío de cronometraje recibido corresponde a un evento deportivo que se está realizando, o bien, pertenece a una lectura de etiqueta RFID destinada a asignar dicha etiqueta a un participante desde la vista de añadir participante o de editar participante. En caso de que el envío de cronometraje sea producto de una asignación de una etiqueta a un participante, se le asignará directamente el identificador de dicha etiqueta al participante correspondiente. En caso de que el envío de cronometraje corresponda a un evento en directo, se siguen realizando comprobaciones necesarias. En caso de que el envío de cronometraje no pertenezca a ninguna de las dos opciones, el envío de cronometraje es descartado.

La tercera comprobación en caso de que el envío de cronometraje corresponda a un evento en directo, se basa en la verificación del participante. Se comprueba que exista un participante en el evento deportivo en directo que a su vez tenga configurado el equipo de cronometraje verificado. En caso de no existir ningún participante con dicha etiqueta RFID en el evento deportivo, se descarta el envío de cronometraje.

La cuarta comprobación en caso de que el envío de cronometraje corresponda a un evento en directo, se basa en la verificación del punto de control y la vuelta. Esta verificación se realiza para evitar que un mismo participante disponga de varias mediciones en el sistema cuando es detectado en varias ocasiones por un mismo equipo de cronometraje al pasar junto al mismo. Para evitar que esto ocurra, se determina si el envío de cronometraje es válido o no en función de los envíos de cronometraje anteriores almacenados en el sistema. Además, en el caso de que un circuito esté configurado para más de una vuelta, se verifica el número de envíos de cronometraje anteriores que se encuentran almacenados en el sistema, para basándose en este número y en el tiempo transcurrido entre el último envío de cronometraje almacenado y el actual junto con el tiempo mínimo de vuelta, determinar si el envío de cronometraje es válido o no.

En caso de que un envío de cronometraje supere satisfactoriamente esta última comprobación, es almacenado en la base de datos. En caso contrario, es descartado.



### 5.5.2. Operaciones para generar clasificación con la tabla de tiempos

Estas operaciones corresponden a la vista *SeguimientoView*. En esta vista se comienza realizando las comprobaciones pertinentes: El evento deportivo existe, la categoría existe y el evento deportivo ha comenzado.

Una vez realizadas las comprobaciones, se procede a obtener de la base de datos todos los envíos de cronometraje pertenecientes a participantes de la categoría y el evento deportivo en cuestión. Los envíos de cronometraje obtenidos se dividen por puntos de control, para poder tener una clasificación en cada uno de los puntos en los que se toman tiempos.

A continuación, se calculan los tiempos de cada participante en cada punto de control. Esta operación se realiza restando el instante de tiempo de la detección de cada envío de cronometraje al instante de tiempo del comienzo del evento deportivo. Es necesario mencionar que para evitar errores, todos los componentes del sistema de cronometraje operan con el formato e fecha y hora UTC.

Una vez que se han obtenido todos los tiempos, se genera una clasificación ordenando los participantes de menor tiempo a mayor en cada punto de control.



## Capítulo 6

# Pruebas y simulaciones

### 6.1. Pruebas de software

A fin de comprobar el correcto funcionamiento del sistema de cronometraje deportivo desarrollado, se ha procedido a realizar pruebas que verifican que el sistema cumple con todas las funcionalidades de la forma esperada siguiendo los casos de uso.

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>CrearEquipoCronometraje</b>   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se añade un equipo de cronometraje en el sistema a través de un formulario, rellenando correctamente el mismo.                                 |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema genera automáticamente un identificador de dispositivo y un <i>API KEY</i> . Se crea un nuevo equipo de cronometraje en el sistema. |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>CrearEquipoCronometraje</b>   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se añade un equipo de cronometraje en el sistema a través de un formulario, dejando en blanco el campo obligatorio del formulario. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con un mensaje de error.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>VerEquipoCronometraje</b>   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se visualizan los parámetros de un equipo de cronometraje y el estado del mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema muestra toda la información y el estado.                              |
| <b>Resultado prueba</b>      | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>EditarEquipoCronometraje</b>  |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se edita un equipo de cronometraje existente en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | Se guardan los cambios del equipo de cronometraje en el sistema.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>EditarEquipoCronometraje</b>  |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se edita un equipo de cronometraje existente en el sistema a través de un formulario, dejando en blanco el campo obligatorio del formulario. |
| <b>Pasos</b>                       | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>          | El sistema muestra de nuevo el formulario con un mensaje de error.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |   |
|------------------------------------|---|
| <b>Caso de uso vincula-<br/>do</b> | <b>EliminarEquipoCronometraje</b>                                 |
| <b>Versión</b>                     | 1.0   |
| <b>Descripción</b>                 | Se elimina un equipo de cronometraje existente en el sistema.     |
| <b>Pasos</b>                       | Flujo normal del caso de uso.                                     |
| <b>Resultado esperado</b>          | El sistema elimina el equipo de cronometraje de la base de datos. |
| <b>Resultado prueba</b>            | OK.   |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>AñadirEventoDeportivo</b>   |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se añade un evento deportivo en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | Se crea un nuevo evento deportivo en el sistema.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |   |
|------------------------------------|---|
| <b>Caso de uso vincula-<br/>do</b> | <b>AñadirEventoDeportivo</b>  |
| <b>Versión</b>                     | 1.0   |
| <b>Descripción</b>                 | Se añade un evento deportivo en el sistema a través de un formulario, introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                       | Flujo alternativo C del caso de uso.  |
| <b>Resultado esperado</b>          | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.  |
| <b>Resultado prueba</b>            | OK.   |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>VerEventoDeportivo</b>  |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se visualizan los parámetros de un evento deportivo y el estado del mismo. |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | El sistema muestra toda la información y el estado.                        |
| <b>Resultado prueba</b>            | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>EditarEventoDeportivo</b>   |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se edita un evento deportivo existente en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | Se guardan los cambios del evento deportivo en el sistema.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>EditarEventoDeportivo</b>   |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se edita un evento deportivo existente en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                       | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>          | El sistema muestra de nuevo el formulario con los mensajes de error pertinentes.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |   |
|------------------------------------|---|
| <b>Caso de uso vincula-<br/>do</b> | <b>EliminarEventoDeportivo</b>  |
| <b>Versión</b>                     | 1.0   |
| <b>Descripción</b>                 | Se elimina el evento deportivo del sistema.   |
| <b>Pasos</b>                       | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>          | El sistema elimina el evento deportivo de la base de datos, así como toda la información que depende del mismo (recorridos, puntos de control, categorías, participantes y envíos de cronometraje). |
| <b>Resultado prueba</b>            | OK.   |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>ComenzarEventoDeportivo</b>   |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se da comienzo al evento deportivo.  |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | El sistema actualiza en la base de datos el estado del evento deportivo, así como la fecha y hora de comienzo. |
| <b>Resultado prueba</b>            | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>FinalizarEventoDeportivo</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se da por finalizado el evento deportivo.                                |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema actualiza en la base de datos el estado del evento deportivo. |
| <b>Resultado prueba</b>      | OK.  |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>AñadirRecorrido</b>  |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se añade un recorrido en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>    | Se crea un nuevo recorrido en el sistema.   |
| <b>Resultado prueba</b>      | OK.   |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>AñadirRecorrido</b>  |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se añade un recorrido en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.  |
| <b>Resultado prueba</b>      | OK.   |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>VerRecorrido</b>   |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se visualizan los parámetros de un recorrido y los puntos de control del mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra toda la información del recorrido.                           |
| <b>Resultado prueba</b>      | OK.   |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>EditarRecorrido</b>  |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se edita un recorrido existente en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>    | Se guardan los cambios del recorrido en el sistema.   |
| <b>Resultado prueba</b>      | OK.   |

|                                    |   |
|------------------------------------|---|
| <b>Caso de uso vincula-<br/>do</b> | <b>EditarRecorrido</b>  |
| <b>Versión</b>                     | 1.0   |
| <b>Descripción</b>                 | Se edita un recorrido existente en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                       | Flujo alternativo C del caso de uso.  |
| <b>Resultado esperado</b>          | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.  |
| <b>Resultado prueba</b>            | OK.   |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>EliminarRecorrido</b>   |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se elimina un recorrido existente en el sistema.   |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | El sistema elimina el recorrido de la base de datos, así como los puntos de control vinculados al mismo. |
| <b>Resultado prueba</b>            | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>AñadirPuntoDeControl</b>  |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se añade un punto de control en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                       | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>          | Se crea un nuevo punto de control en el sistema.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>AñadirPuntoDeControl</b>  |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se añade un punto de control en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                       | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>          | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.   |
| <b>Resultado prueba</b>            | OK.  |

|                                    |  |
|------------------------------------|--|
| <b>Caso de uso vincula-<br/>do</b> | <b>VerPuntoDeControl</b>                                     |
| <b>Versión</b>                     | 1.0  |
| <b>Descripción</b>                 | Se visualizan los parámetros de un punto de control.         |
| <b>Pasos</b>                       | Flujo normal del caso de uso.                                |
| <b>Resultado esperado</b>          | El sistema muestra toda la información del punto de control. |
| <b>Resultado prueba</b>            | OK.  |

|  |  |
|--|--|
| <b>Caso de uso vincula-</b><br><b>do</b> | <b>EditarPuntoDeControl</b>  |
| <b>Versión</b>                           | 1.0  |
| <b>Descripción</b>                       | Se edita un punto de control existente en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                             | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>                | Se guardan los cambios del punto de control en el sistema.   |
| <b>Resultado prueba</b>                  | OK.  |

|  |  |
|--|--|
| <b>Caso de uso vincula-</b><br><b>do</b> | <b>EditarPuntoDeControl</b>  |
| <b>Versión</b>                           | 1.0  |
| <b>Descripción</b>                       | Se edita un punto de control existente en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                             | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>                | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.   |
| <b>Resultado prueba</b>                  | OK.  |

|  |   |
|--|---|
| <b>Caso de uso vincula-</b><br><b>do</b> | <b>EliminarPuntoDeControl</b>                               |
| <b>Versión</b>                           | 1.0   |
| <b>Descripción</b>                       | Se elimina un punto de control existente en el sistema.     |
| <b>Pasos</b>                             | Flujo normal del caso de uso.                               |
| <b>Resultado esperado</b>                | El sistema elimina el punto de control de la base de datos. |
| <b>Resultado prueba</b>                  | OK.   |

|  |  |
|--|--|
| <b>Caso de uso vincula-</b><br><b>do</b> | <b>AñadirCategoria</b>   |
| <b>Versión</b>                           | 1.0  |
| <b>Descripción</b>                       | Se añade una categoría en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                             | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>                | Se crea una categoría en el sistema.   |
| <b>Resultado prueba</b>                  | OK.  |

|  |   |
|--|---|
| <b>Caso de uso vincula-</b><br><b>do</b> | <b>AñadirCategoria</b>  |
| <b>Versión</b>                           | 1.0   |
| <b>Descripción</b>                       | Se añade una categoría en el sistema a través de un formulario, introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                             | Flujo alternativo C del caso de uso.  |
| <b>Resultado esperado</b>                | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.  |
| <b>Resultado prueba</b>                  | OK.   |



|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>VerCategoria</b>                                     |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se visualizan los parámetros de una categoría.          |
| <b>Pasos</b>                 | Flujo normal del caso de uso.                           |
| <b>Resultado esperado</b>    | El sistema muestra toda la información de la categoría. |
| <b>Resultado prueba</b>      | OK.   |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>EditarCategoria</b>   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se edita una categoría existente en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | Se guardan los cambios de la categoría en el sistema.  |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>EditarCategoria</b>   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se edita una categoría existente en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>EliminarCategoria</b>                             |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se elimina una categoría existente en el sistema.    |
| <b>Pasos</b>                 | Flujo normal del caso de uso.                        |
| <b>Resultado esperado</b>    | El sistema elimina la categoría de la base de datos. |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>AñadirParticipante</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se añade un participante en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | Se crea un participante en el sistema.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>AñadirParticipante</b>   |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se añade un participante en el sistema a través de un formulario, introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.  |
| <b>Resultado prueba</b>      | OK.   |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>AñadirParticipante</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se añade un participante en el sistema a través de un formulario introduciendo valores correctos, pero con una etiqueta RFID que ya ha sido asignada a otro participante del evento deportivo. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>AñadirParticipante</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se añade un participante en el sistema a través de un formulario introduciendo valores correctos, pero con un dorsal que ya ha sido asignado a otro participante del evento deportivo. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>VerParticipante</b>                                   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se visualizan los parámetros de un participante.         |
| <b>Pasos</b>                 | Flujo normal del caso de uso.                            |
| <b>Resultado esperado</b>    | El sistema muestra toda la información del participante. |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>EditarParticipante</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se edita un participante existente en el sistema a través de un formulario, rellenando correctamente el mismo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | Se guardan los cambios del participante en el sistema.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>EditarParticipante</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se edita un participante existente en el sistema a través de un formulario introduciendo valores incorrectos. Esta prueba se ha realizado dejando en blanco uno o varios campos obligatorios sin rellenar, así como introduciendo valores inválidos. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra de nuevo el formulario con los mensajes de error adecuados.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>EliminarParticipante</b>                             |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se elimina un participante existente en el sistema.     |
| <b>Pasos</b>                 | Flujo normal del caso de uso.                           |
| <b>Resultado esperado</b>    | El sistema elimina el participante de la base de datos. |
| <b>Resultado prueba</b>      | OK.   |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>VerEventoDeportivoPublico</b>   |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se visualiza la información pública de un evento deportivo siguiendo el enlace proporcionado por el sistema. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema muestra toda la información del evento deportivo, así como su estado.                             |
| <b>Resultado prueba</b>      | OK.  |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>SeguirEventoDeportivoPublico</b>   |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se visualiza la clasificación con la tabla de tiempos de un evento deportivo que ha comenzado. En esta prueba se han producido varios envíos de cronometraje pertenecientes a dicho evento. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra la tabla de tiempos con la clasificación generada de forma correcta.   |
| <b>Resultado prueba</b>      | OK.   |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>SeguirEventoDeportivoPublico</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se visualiza la clasificación con la tabla de tiempos de un evento deportivo que ha comenzado. En esta prueba no se ha producido ningún envío de cronometraje. |
| <b>Pasos</b>                 | Flujo alternativo C del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema muestra un mensaje indicando que ningún participante ha obtenido tiempos todavía.   |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>CrearEnvioCronometraje</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se realiza un envío de cronometraje con toda la información correcta, es decir, identificador y <i>API KEY</i> de equipo de cronometraje, y etiqueta RFID de un participante de un evento deportivo que se está celebrando en directo. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema almacena el envío de cronometraje en la base de datos.  |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>CrearEnvioCronometraje</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se realiza un envío de cronometraje con un identificador y <i>API KEY</i> de equipo de cronometraje incorrectos. |
| <b>Pasos</b>                 | Flujo alternativo B del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema descarta el envío de cronometraje.  |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>CrearEnvioCronometraje</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se realiza un envío de cronometraje con una etiqueta que no pertenece a ningún participante. |
| <b>Pasos</b>                 | Flujo alternativo B del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema descarta el envío de cronometraje.  |
| <b>Resultado prueba</b>      | OK.  |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>CrearEnvioCronometraje</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se realiza un envío de cronometraje con los parámetros correctos, pero que pertenecen a un evento deportivo ya finalizado. |
| <b>Pasos</b>                 | Flujo alternativo B del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema descarta el envío de cronometraje.  |
| <b>Resultado prueba</b>      | OK.  |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>CrearEnvioCronometraje</b>   |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se realiza un envío de cronometraje con los parámetros correctos, pero antes de que haya transcurrido el tiempo mínimo de medición. |
| <b>Pasos</b>                 | Flujo alternativo B del caso de uso.  |
| <b>Resultado esperado</b>    | El sistema descarta el envío de cronometraje.   |
| <b>Resultado prueba</b>      | OK.   |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>ActualizarEstadoEquipoCronometraje</b>   |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se realiza un envío de estado con un identificador y <i>API KEY</i> de equipo de cronometraje correctos, estando el lector RFID operando correctamente. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema actualiza el estado del equipo de cronometraje como Operativo.   |
| <b>Resultado prueba</b>      | OK.   |

|                              |   |
|------------------------------|---|
| <b>Caso de uso vinculado</b> | <b>ActualizarEstadoEquipoCronometraje</b>   |
| <b>Versión</b>               | 1.0   |
| <b>Descripción</b>           | Se realiza un envío de estado con un identificador y <i>API KEY</i> de equipo de cronometraje correctos, con el lector RFID desconectado. |
| <b>Pasos</b>                 | Flujo normal del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema actualiza el estado del equipo de cronometraje como Conectado, no operativo.   |
| <b>Resultado prueba</b>      | OK.   |

|                              |  |
|------------------------------|--|
| <b>Caso de uso vinculado</b> | <b>ActualizarEstadoEquipoCronometraje</b>  |
| <b>Versión</b>               | 1.0  |
| <b>Descripción</b>           | Se realiza un envío de estado con un identificador y <i>API KEY</i> de equipo de cronometraje incorrectos. |
| <b>Pasos</b>                 | Flujo alternativo B del caso de uso.   |
| <b>Resultado esperado</b>    | El sistema descarta la información recibida.   |
| <b>Resultado prueba</b>      | OK.  |

## 6.2. Simulaciones de eventos deportivos

Una vez que el sistema ha superado las pruebas de software, para finalizar las verificaciones sobre el buen funcionamiento del sistema se han realizado simulaciones de eventos deportivos que se acercan en la medida de lo posible a casos reales.

|                                   |  |
|-----------------------------------|--|
| <b>Título</b>                     | <b>Evento de un recorrido y una categoría.</b>   |
| <b>Versión</b>                    | 1.0  |
| <b>Descripción y pasos</b>        | El objetivo es testear el sistema completo para el caso más simple. Consiste en la creación de un evento deportivo, que se compone de un recorrido y una categoría. El recorrido dispone de un único punto de control y una sola vuelta. Se registran varios participantes en el evento, todos pertenecientes a la única categoría del evento. Una vez configurado, se comienza el evento deportivo. |
| <b>Resultado esperado</b>         | <ul style="list-style-type: none"> <li>■ Todos los tiempos de los participantes han sido correctamente registrados en el sistema.</li> <li>■ El evento se ha podido seguir en directo correctamente.</li> <li>■ La tabla de tiempos y clasificación es correcta.</li> </ul>  |
| <b>Resultado de la simulación</b> | OK.  |

|                                   |  |
|-----------------------------------|--|
| <b>Título</b>                     | <b>Evento de dos recorridos y dos categorías.</b>  |
| <b>Versión</b>                    | 1.0  |
| <b>Descripción y pasos</b>        | El objetivo es testear el sistema completo para el caso de un evento deportivo que tiene dos categorías, y en cada categoría se realiza un recorrido diferente. Consiste en la creación de un evento deportivo, que contiene dos recorridos y dos categorías. Ambos recorridos disponen de un único punto de control y son de una sola vuelta. El equipo de cronometraje configurado para los puntos de control de ambos recorridos es el mismo. Se registran varios participantes en el evento, repartidos en las dos categorías. Una vez configurado, se comienza el evento deportivo. |
| <b>Resultado esperado</b>         | <ul style="list-style-type: none"> <li>■ Todos los tiempos de los participantes han sido correctamente registrados en el sistema.</li> <li>■ El evento se ha podido seguir en directo correctamente.</li> <li>■ Las tablas de tiempos y clasificaciones son correctas, estando cada participante en la clasificación de su categoría correspondiente.</li> </ul>   |
| <b>Resultado de la simulación</b> | OK.  |

|                                   |  |
|-----------------------------------|--|
| <b>Título</b>                     | <b>Evento de un recorrido y dos categorías.</b>  |
| <b>Versión</b>                    | 1.0  |
| <b>Descripción y pasos</b>        | El objetivo es testear el sistema completo para el caso de un evento deportivo que tiene dos categorías, y en ambas categorías se realiza el mismo recorrido. Consiste en la creación de un evento deportivo, que contiene un recorrido y dos categorías. El recorrido dispone de un único punto de control y es de una sola vuelta. Se registran varios participantes en el evento, repartidos en las dos categorías. Una vez configurado, se comienza el evento deportivo. |
| <b>Resultado esperado</b>         | <ul style="list-style-type: none"> <li>■ Todos los tiempos de los participantes han sido correctamente registrados en el sistema.</li> <li>■ El evento se ha podido seguir en directo correctamente.</li> <li>■ Las tablas de tiempos y clasificaciones son correctas, estando cada participante en la clasificación de su categoría correspondiente.</li> </ul>   |
| <b>Resultado de la simulación</b> | OK.  |

|                                   |   |
|-----------------------------------|---|
| <b>Título</b>                     | <b>Evento de un recorrido de dos vueltas y una categoría.</b>   |
| <b>Versión</b>                    | 1.0   |
| <b>Descripción y pasos</b>        | El objetivo es testear el sistema completo para cuando un recorrido tiene más de una vuelta, en el caso más simple posible. Consiste en la creación de un evento deportivo, que contiene un recorrido y una categoría. El recorrido dispone de un único punto de control y es de dos vueltas. Se registran varios participantes en el evento, todos pertenecientes a la única categoría del evento. Una vez configurado, se comienza el evento deportivo. |
| <b>Resultado esperado</b>         | <ul style="list-style-type: none"> <li>■ Todos los tiempos de los participantes han sido correctamente registrados en el sistema.</li> <li>■ El evento se ha podido seguir en directo correctamente.</li> <li>■ En la vista de seguimiento, se muestra cada vuelta como un lugar de medición independiente.</li> <li>■ La tabla de tiempos y clasificación es correcta.</li> </ul>  |
| <b>Resultado de la simulación</b> | OK.   |

|                                   |  |
|-----------------------------------|--|
| <b>Título</b>                     | <b>Evento de dos recorridos (uno de ellos de dos vueltas) y dos categorías.</b>  |
| <b>Versión</b>                    | 1.0  |
| <b>Descripción y pasos</b>        | <p>El objetivo es testear el sistema completo para cuando un evento deportivo dispone de dos recorridos, cuya única diferencia es que uno de ellos es de una vuelta y el otro es de dos vueltas. Es decir, físicamente el circuito es el mismo, pero en un caso se recorre dos veces. Para ello se procede a la creación de un evento deportivo, que contiene dos recorridos y dos categorías. Ambos recorridos disponen de un único punto de control, siendo uno de ellos de una vuelta y el otro de dos vueltas. Se registran varios participantes en el evento, repartidos en las dos categorías. Una vez configurado, se comienza el evento deportivo.</p> |
| <b>Resultado esperado</b>         | <ul style="list-style-type: none"><li>▪ Todos los tiempos de los participantes han sido correctamente registrados en el sistema.</li><li>▪ El evento se ha podido seguir en directo correctamente.</li><li>▪ En la vista de seguimiento se muestran correctamente para ambas categorías, cada vuelta como un lugar de medición independiente.</li><li>▪ La tabla de tiempos y clasificación es correcta en ambas categorías.</li></ul>   |
| <b>Resultado de la simulación</b> | OK.  |



## Capítulo 7

# Conclusiones y líneas futuras

### 7.1. Conclusión sobre el sistema desarrollado

El sistema de cronometraje deportivo desarrollado en este TFG supone un sistema perfectamente válido para eventos deportivos de baja participación cuyas modalidades deportivas sean el ciclismo de montaña o el running, y en las que los tiempos de los participantes requieran una precisión de segundos. Este sistema de cronometraje permite configurar las características y particularidades de la mayoría de eventos deportivos de estas modalidades de forma fácil e intuitiva. Además, es una solución bastante económica, sobre todo teniendo en cuenta los costes en los que se mueven soluciones con unas prestaciones similares en cuanto a precisión y fiabilidad.

Sin embargo, el sistema tiene debilidades que le dificultan ser utilizado en eventos deportivos cuyas modalidades deportivas sean de otro tipo y, en eventos deportivos en los que las llegadas son disputadas entre varios participantes ya que se requiere mayor precisión.

Una debilidad es la precisión en la toma de tiempos a la hora de identificar los participantes a su paso por los puntos de control. Para los eventos deportivos a los que está destinado el sistema de cronometraje desarrollado es perfectamente válida la precisión actual, ya que en eventos de baja participación cuyos participantes son aficionados, no se producen llegadas a meta disputadas. Sin embargo, en un evento en el que la llegada a meta está disputada por varios participantes, el sistema de cronometraje actual podría identificar a los participantes en un orden incorrecto, y por tanto generar una clasificación incorrecta. Esta debilidad podría ser solventada empleando un sistema de *foto finish* como apoyo.

Una segunda debilidad aparece en el caso de que sea necesario ubicar un punto de control en un lugar en el que no se dispone de conexión a internet ni toma de corriente eléctrica, ya que un equipo de cronometraje requiere de ambos servicios para poder operar. En estos casos será necesario emplear medios auxiliares para proporcionar tanto internet como alimentación eléctrica, se proponen a continuación algunas opciones que solventan estas necesidades:

- La toma de corriente puede ser proporcionada por un generador eléctrico portátil a gasolina. El consumo de un equipo de cronometraje viene determinada por el consumo de una *Raspberry Pi* y del lector/antena RFID, la suma de ambos dispositivos se encuentra por debajo de los 100 W. Dado que las necesidades de potencia son tan pequeñas, existen numerosos generadores en el mercado que pueden servir para este propósito.
- Una opción para proporcionar conexión a Internet es emplear un router 3G/4G, que opera con una tarjeta SIM con tarifa de datos contratados a una operadora de telecomunicaciones. En este caso sería necesario que hubiera cobertura móvil en el lugar donde se ubica el punto de control.
- Una segunda opción para proporcionar conexión a Internet es emplear antenas con tecnología *WI-FI* o *Air MAX* (tecnología desarrollada por Ubiquiti). Estas antenas permiten llevar la

conexión a internet a lugares que se encuentren a una distancia de hasta 15 km. Para ello se ubica una antena en un lugar cercano en el que exista conexión a internet (típicamente la localidad más cercana), y se ubica una segunda antena en el lugar donde se encuentra el punto de control. Esta opción únicamente sería válida para puntos de control que se encuentran a una distancia máxima de 15 km. de un lugar en el que se disponga de conexión a internet.

## 7.2. Líneas futuras: mejoras a implementar en el sistema

Si bien el sistema desarrollado cumple con el objetivo de ser un sistema de cronometraje deportivo adecuado para eventos deportivos de baja participación enfocado a las modalidades de ciclismo de montaña y *running*, se han detectado debilidades que podrían ser mejoradas en próximas versiones del sistema. Además, cabe la posibilidad de extender el sistema a otro tipo de eventos deportivos con ligeros cambios en el sistema.

### 7.2.1. Mejoras en el panel de administración de la aplicación web

**Permitir editar los envíos de cronometraje desde el panel de administración.** Cuando se producen llegadas a meta disputadas existe la posibilidad de que el sistema detecte con anterioridad al participante que se encuentra por detrás, en estos casos el sistema de cronometraje se debe apoyar en un sistema de *foto finish* asistido por jueces del evento deportivo. Para que los jueces puedan editar la clasificación dentro del sistema de cronometraje, es necesario que se puedan editar los envíos de cronometraje de cara a alterar mínimamente los tiempos y de esta manera corregir la clasificación.

**Exportación de clasificaciones.** Actualmente la clasificación con los tiempos de los eventos deportivos se puede consultar a través de una URL pública que se encuentra accesible desde cualquier lugar. Sin embargo, podría resultar de utilidad para las organizaciones de los eventos deportivos disponer de la clasificación en formato PDF o Excel. Para desarrollar esta mejora se podrían emplear módulos *python* que facilitan el desarrollo.

**Importación de participantes.** Actualmente para registrar los participantes en un evento deportivo desde el panel de administración se debe realizar el registro uno a uno. Sin embargo, en ocasiones puede resultar más cómodo para los organizadores disponer de la lista completa de participantes en una hoja Excel o CSV, y después importar la lista al completo desde el panel de administración. Para desarrollar esta funcionalidad se tendría que incorporar al panel de administración un botón de importar participantes junto con unas indicaciones sobre el formato de columnas que deben seguir los ficheros a importar. De nuevo, existen módulos *python* que facilitan el desarrollo de esta funcionalidad.

**Permitir salidas escalonadas.** Si bien en la mayoría de eventos deportivos de baja participación se producen salidas en masa, podría resultar interesante que el sistema de cronometraje también soportara salidas escalonadas. En las salidas escalonadas los participantes no toman la salida al mismo tiempo, sino que salen de uno en uno con un pequeño margen de tiempo entre participante y participante. Para que el sistema pueda proporcionar esta funcionalidad, es necesario añadir un nuevo parámetro en los eventos deportivos que defina el tipo de salida, además de crear un panel de salida con un botón asignado a cada participante basado en el dorsal.

**Implementar un selector de fecha y hora.** A la hora de añadir o editar un evento deportivo es necesario proporcionar una fecha y hora de comienzo manualmente. Aunque está suficientemente indicado el formato que se debe seguir la fecha y hora, incorporar un selector de fecha y hora supondría una mejora en cuanto a usabilidad.

### 7.2.2. Mejoras en la pantalla de seguimiento público

**Tiempos en una única tabla.** Actualmente en la pantalla de seguimiento, en aquellas categorías cuyos recorridos cuentan con varios puntos de control (o con varias vueltas) se muestra una tabla

de tiempos por cada punto de control en forma de pestañas ordenadas por punto kilométrico. La decisión de mostrar la información con este formato ha sido motivada por la posibilidad de realizar el seguimiento tanto desde ordenadores como desde móviles, con el problema de que los móviles tienen un tamaño de pantalla muy limitado que dificulta mostrar tablas de mayores dimensiones. Sin embargo, resultaría más esclarecedor mostrar todos los tiempos en una misma tabla, ordenados por columnas según el punto kilométrico. La mejora propuesta consiste en implementar el seguimiento de cada categoría en una única tabla cuando se accede desde un dispositivo con pantalla lo suficientemente grande (ordenadores) y mantener las tablas actuales cuando se accede desde un dispositivo con pantalla pequeña (móviles).

### 7.2.3. Mejoras en los equipos de cronometraje

**Incluir conexión a internet.** Los equipos de cronometraje deben disponer de una conexión a internet externa para poder operar, lo cual reduce los lugares posibles en los que pueden ser ubicados. Esta característica podría ser mejorada incorporando al equipo de cronometraje un módulo SIM que permita proporcionar conexión a internet a través de una tarjeta SIM de una compañía de telecomunicaciones. El módulo propuesto para realizar esta mejora es el *SIM808*, este módulo puede formar parte del equipo de cronometraje una vez se encuentre conectado y configurado con la *Raspberry Pi*.

**Incorporar batería** Los equipos de cronometraje deben disponer de alimentación eléctrica para poder operar, lo cual reduce los lugares posibles en los que pueden ser ubicados. Esta característica podría ser mejorada incorporando una batería al equipo de cronometraje.



# Bibliografía

- [atl18a] atlasRFIDstore. *Laird S9025PL/S8655PL (LHCP) Outdoor RFID Antenna (FCC/ETSI)*. 2018. URL: <https://www.atlasrfidstore.com/laird-s9025pl-s8655pl-lhcp-outdoor-rfid-antenna-fcc-etsi/>.
- [atl18b] atlasRFIDstore. *Times-7 RFID Race Timing Antenna System (FCC/ETSI)*. 2018. URL: <https://www.atlasrfidstore.com/times-7-rfid-race-timing-antenna-system-fcc-etsi/>.
- [atl18c] atlasRFIDstore. *Zebra FX7500 RFID Reader - 2 Port*. 2018. URL: <https://www.atlasrfidstore.com/zebra-fx7500-rfid-reader-2-port/>.
- [Chr18] Tom Christie. *Django REST framework*. 2018. URL: <http://www.django-rest-framework.org/>.
- [Cur+07] Jari-Pascal Curty y col. *Design and Optimization of Passive UHF RFID Systems*. Springer Science-FBusiness Media, 2007.
- [Est05] Agencia Estatal Boletín Oficial del Estado. *Agencia Estatal Boletín Oficial del Estado*. 2005. URL: [http://www.boe.es/diario\\_boe/txt.php?id=BOE-A-2008-11868](http://www.boe.es/diario_boe/txt.php?id=BOE-A-2008-11868).
- [Fou18a] Django Software Foundation. *Django, The Web framework for perfectionists with deadlines*. 2018. URL: <https://www.djangoproject.com/>.
- [Fou18b] jQuery Foundation. *jQuery*. 2018. URL: <http://jquery.com/>.
- [Fou18c] Raspberry Pi Foundation. *Installing operating system images - Raspberry Pi Documentation*. 2018. URL: <https://www.raspberrypi.org/documentation/installation/installing-images/>.
- [Fou18d] Raspberry Pi Foundation. *Raspberry Pi - Teach, Learn, and Make with Raspberry Pi*. 2018. URL: <https://www.raspberrypi.org/>.
- [Fou18e] Raspberry Pi Foundation. *Raspbian - Raspberry Pi Documentation*. 2018. URL: <https://www.raspberrypi.org/documentation/raspbian/>.
- [Fou18f] Raspberry Pi Foundation. *rc.local - Raspberry Pi Documentation*. 2018. URL: <https://www.raspberrypi.org/>.
- [Fou18g] Raspberry Pi Foundation. *The Raspberry Pi UARTs - Raspberry Pi Documentation*. 2018. URL: <https://www.raspberrypi.org/documentation/configuration/uart.md>.
- [Hip18] D. Richard Hipp. *SQLite Home Page*. 2018. URL: <https://www.sqlite.org/>.
- [How18] Phil Howard. *Raspberry Pi GPIO Pinout*. 2018. URL: <https://pinout.xyz/>.
- [Inc18] Webscorer Inc. *Webscorer: Race registration, race timing, race results*. 2018. URL: <https://www.webscorer.com/>.
- [Lie17] Chris Liechti. *pySerial 3.4 documentation*. 2017. URL: <https://www.raspberrypi.org/documentation/raspbian/>.
- [OTc18] Mark Otto, Jacob Thornton y Bootstrap contributors. *Bootstrap · The most popular HTML, CSS, and JS library in the world*. 2018. URL: <https://getbootstrap.com/>.
- [Rac18] RaceClocker. *RaceClocker - Easy do-it-yourself manual timing of sports races*. 2018. URL: <https://www.raceclocker.com/>.

- [SL18] Evorama Solutions S.L. *Evorama: Smart Racing*. 2018. URL: <http://www.evorama.com/es/>.
- [Tim18a] MYLAPS Sports Timing. *MYLAPS BibTag System for running and more*. 2018. URL: <https://www.mylaps.com/systems/bibtag-system/>.
- [Tim18b] MYLAPS Sports Timing. *MYLAPS ChampionChip System*. 2018. URL: <https://www.mylaps.com/systems/championchip-system/>.
- [Tim18c] MYLAPS Sports Timing. *MYLAPS: Automated Sports Timing and Live Performance Insights*. 2018. URL: <https://www.mylaps.com/>.
- [Tim18d] MYLAPS Sports Timing. *ProChip Timing System for Active Sports*. 2018. URL: <https://www.mylaps.com/systems/prochip-system/>.