

Windows 7

# **Tabla de contenido**

[Tabla de contenido 2](#_Toc48426240)

[Tabla de ilustraciones 2](#_Toc48426241)

[Parte 1: 3](#_Toc48426242)

[Parte 1.1: 3](#_Toc48426243)

[Parte 1.1.1; 3](#_Toc48426244)

[Parte 2: 3](#_Toc48426245)

[Parte 2.1: 3](#_Toc48426246)

[Parte 2.2: 3](#_Toc48426247)

# **Tabla de ilustraciones**

# Introducción

En esta memoria se desarrollará el trabajo realizado en este proyecto, denominado "ArduiCar" (Arduino Car), como parte del módulo de trabajo de fin de ciclo de Administración de Sistemas Informáticos en Red.

La finalidad que tiene este proyecto es implementar los conocimientos y técnicas aprendidos en los módulos cursados y aplicarlos en un trabajo.

En este proyecto se realizará el diseño de una aplicación web que permita el control del automóvil a través de ella, por lo que el ámbito de desarrollo estará centrado en dispositivos pc, aunque también se puede realizar una implementación en dispositivos móviles y smartphones.

También se realizará la configuración de dispositivos de Arduino para crear un sistema domótico en el vehículo para mejorar su calidad, y la codificación del código que permita controlar el coche mediante la aplicación web antes citada.

Para la implementación física se este proyecto se emplea el hardware necesario para construir la estructura del vehículo (chasis, llantas...) y los componentes de hardware de control necesarios (motores, puente H, módulo bluetooth...), además de los dispositivos de recopilación de datos para la domótica (sensores de temperatura, proximidad, sonido, etc.) y los de conexión de los periféricos (placas Arduino).

## Objetivos del proyecto

Los objetivos propuestos para este proyecto son controlar el coche a través de una aplicación web para el manejo de este mediante ordenadores con tecnología bluetooth, crear la aplicación web en la que se pueda registrar usuarios para controlar el vehículo y domótica, y crear un sistema domótico para el vehículo para recopilar datos mientras de maneja el coche y que puedan mejorar la calidad de conducción.

## Distribución del tiempo invertido

Durante los meses disponibles para la realización del proyecto se ha utilizado la metodología por objetivos, la cual se basa en establecer unos objetivos de progreso los cuales se realizarán para una fecha concreta.

Como herramienta de apoyo se ha utilizado [*Jira Software*](https://www.atlassian.com/es/software/jira), la cual fue desarrollada por la empresa Atlassian, con el objetivo de administrar las tareas de un proyecto, hacer seguimiento de los errores e incidencias que surgen y realizar la gestión operativa del proyecto.



Ilustración : Vista general de Jira Software

## Costes de realización



Ilustración : Lista de tareas en Jira Software

Coche

Kit Arduino

Sensores proximidad

Sensor temperatura

# Servicios utilizados

## Xampp

Se trata de una herramienta basada en software libre utilizada, tanto en dispositivos con distribución Windows, Linux o Mac, para el desarrollo web la cual sirve para probar los códigos de las aplicaciones web creadas sin la necesidad de disponer de los servicios utilizados de forma local ni de una conexión a internet.

Entre los servicios que ofrece esta herramienta se encuentran:

* Apache: el servidor web de código abierto es la aplicación usado globalmente para la entrega de contenidos web. Las aplicaciones del servidor son ofrecidas como software libre por la Apache Software Foundation.
* MySQL/MariaDB: XAMPP cuenta con uno de los sistemas relacionales de gestión de bases de datos más populares del mundo. En combinación con el servidor web Apache y el lenguaje PHP, MySQL sirve para el almacenamiento de datos para servicios web. En las versiones actuales de XAMPP esta base de datos se ha sustituido por MariaDB.
* PHP: es un lenguaje de programación de código de lado del servidor que permite crear páginas web o aplicaciones dinámicas. Es independiente de plataforma y soporta varios sistemas de bases de datos.
* Perl: este lenguaje de programación se usa en la administración del sistema, en el desarrollo web y en la programación de red. También permite programar aplicaciones web dinámicas.

Además de estos componentes principales, esta distribución gratuita también incluye, según el sistema operativo, otras herramientas como el servidor de correo Mercury, el programa de administración de bases de datos phpMyAdmin, el software de analítica web Webalizer, OpenSSL, Apache Tomcat y los servidores FTP FileZilla o ProFTPd.

## Node.js

**Node.js** es un entorno de tiempo de ejecución en tiempo real de JavaScript, creado por los **desarrolladores originales de JavaScript,** transformando algo que solo podía ejecutarse en el navegador en algo que se podría ejecutar en los ordenadores como si fuesen aplicaciones independientes y permitiendo hacer cosas que solo tenían la capacidad de hacer otros lenguajes de secuencia de comandos como Python.

**Node.js** utiliza un**modelo de entrada y salida sin bloqueo**controlado por eventos que lo hace ligero y eficiente con la finalidad principal de no se centra en operaciones intensivas del procesador.

Donde Node.js realmente brilla es en la **creación de aplicaciones de red rápidas,** ya que es capaz de manejar una gran cantidad de conexiones simultáneas con un alto nivel de rendimiento, lo que equivale a una **alta escalabilidad.**

A diferencia de las técnicas tradicionales de los servicios web en los que cada conexión genera un nuevo subproceso, ocupando la RAM del sistema y regularmente maximizando la cantidad de RAM disponible, **Node.js opera en un solo subproceso,** utilizando el modelo entrada y salida sin bloqueo, lo que le permite soportar decenas de miles de conexiones al mismo tiempo mantenidas en el bucle de eventos.

Cuando hay una nueva solicitud se genera un tipo de evento. El servidor empieza a procesarlo y, cuando hay una operación de bloqueo de entrada y salida, no espera hasta que se complete y en su lugar crea una función de devolución de llamada. El servidor comienza en el acto a procesar otro evento (tal vez otra solicitud) y cuando finaliza la operación de entrada y salida, continuará trabajando en la solicitud ejecutando la devolución de llamada tan pronto como tenga tiempo.

Por lo tanto, el servidor **nunca necesita crear más subprocesos** o cambiar entre subprocesos, lo que significa que **tiene muy poca sobrecarga.**

Hay que mencionar que en el nivel más bajo de codificación (código C++), encontramos varios subprocesos en Node.js: hay un grupo de eventos de entrada y salida cuyo trabajo es recibir las interrupciones de los propios eventos de entrada y salida y poner los eventos correspondientes en la cola para ser procesados por el hilo principal evitando así la posible interrupción del mismo.

## Arduino

ddd

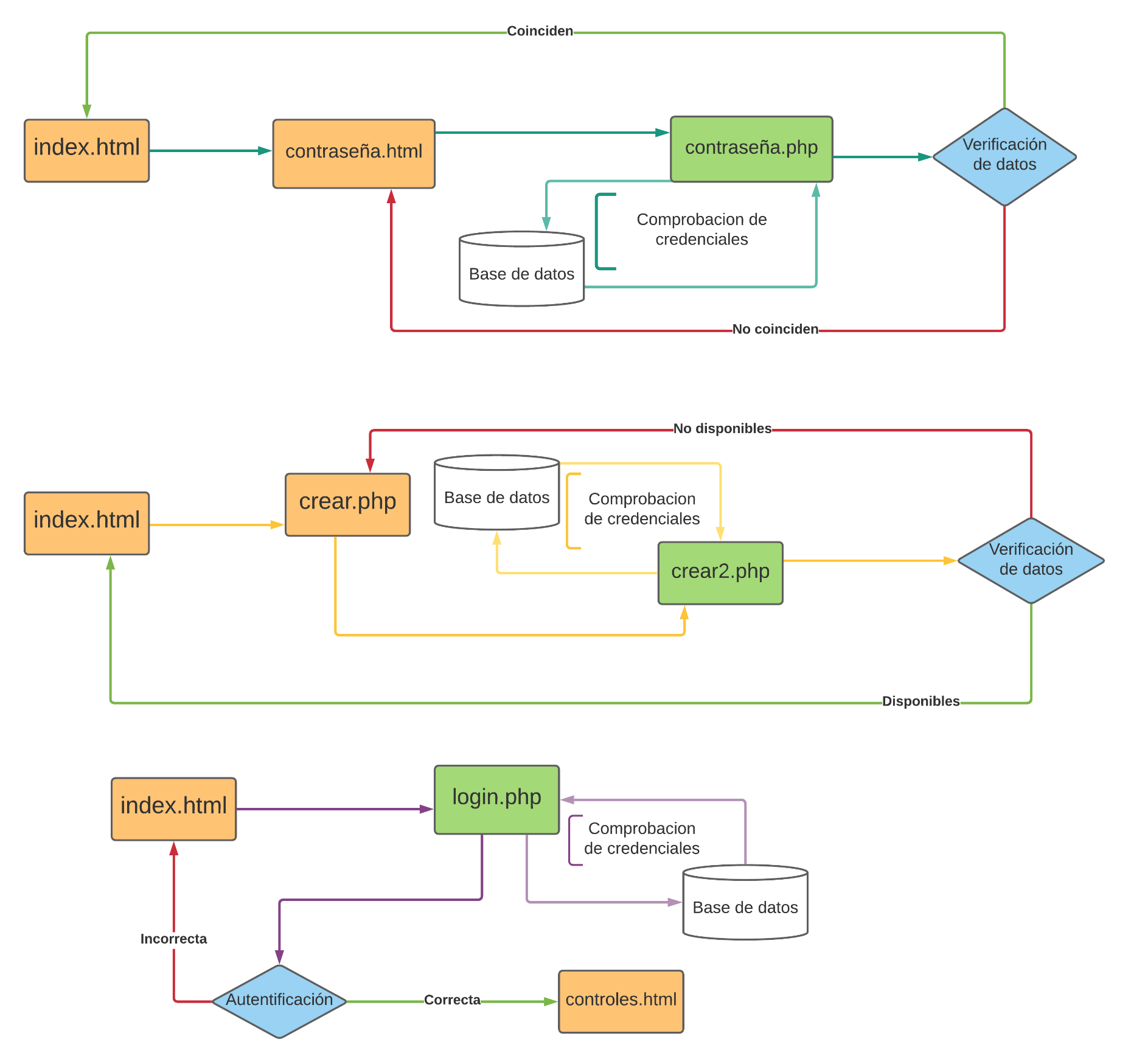
# Implementación web

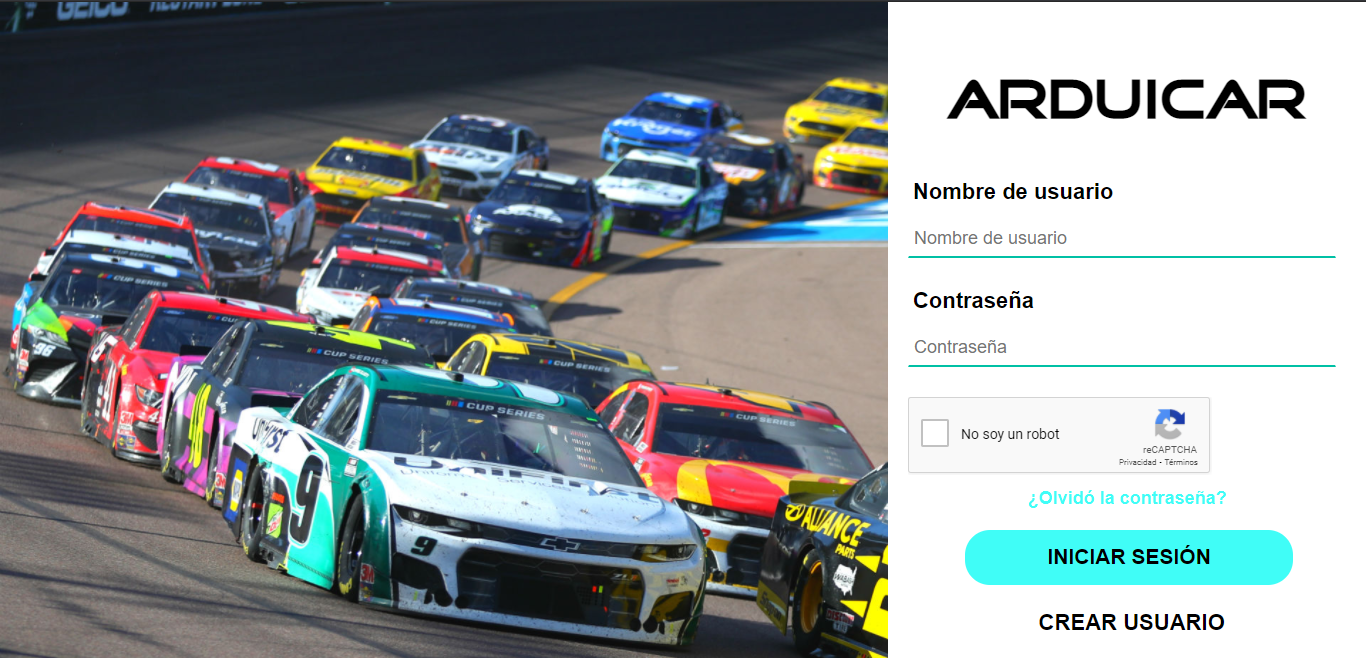
En esta parte de la memoria se desarrollará la parte relacionada con la aplicación web, y para ello se ha utilizado, en primera instancia, los servicios Apache y MySQL, ofrecidos por XAMPP, para manejar páginas creando un servidor web, y para crear la base de datos del proyecto, la cual esta formada por una sola tabla en la que se conforman los tipos de datos de los usuarios.

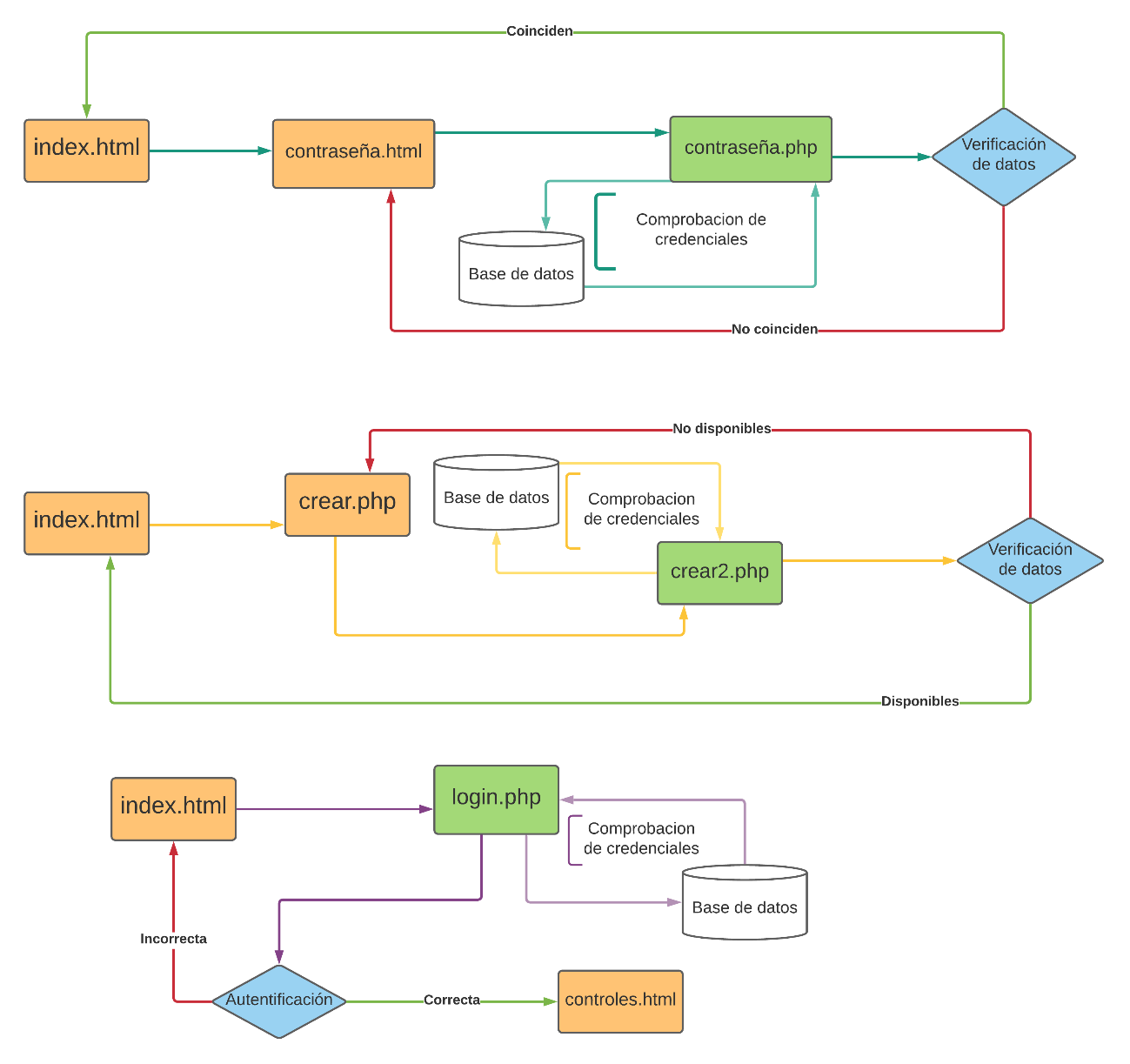
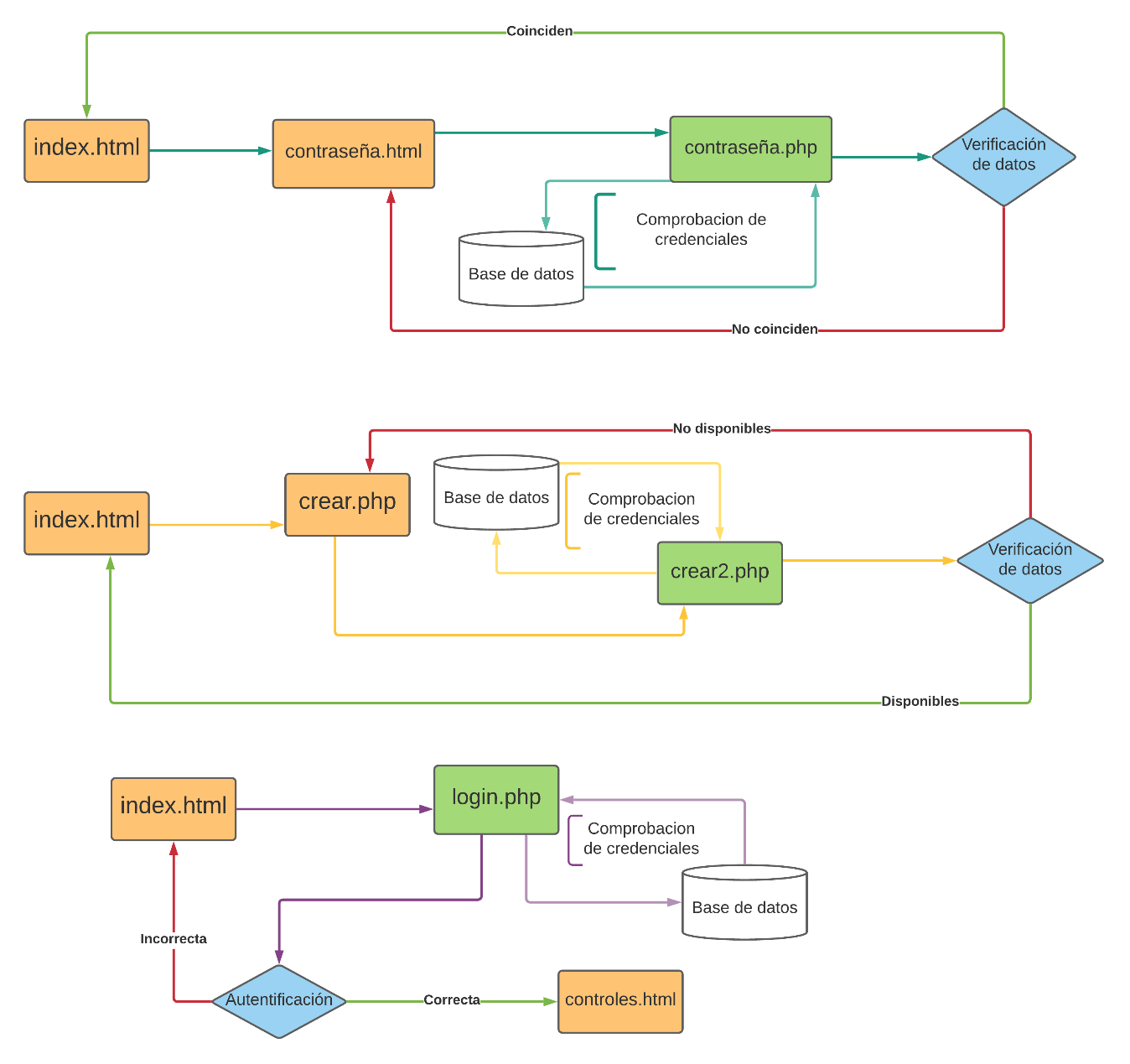
***Añadir introducción a esta parte***

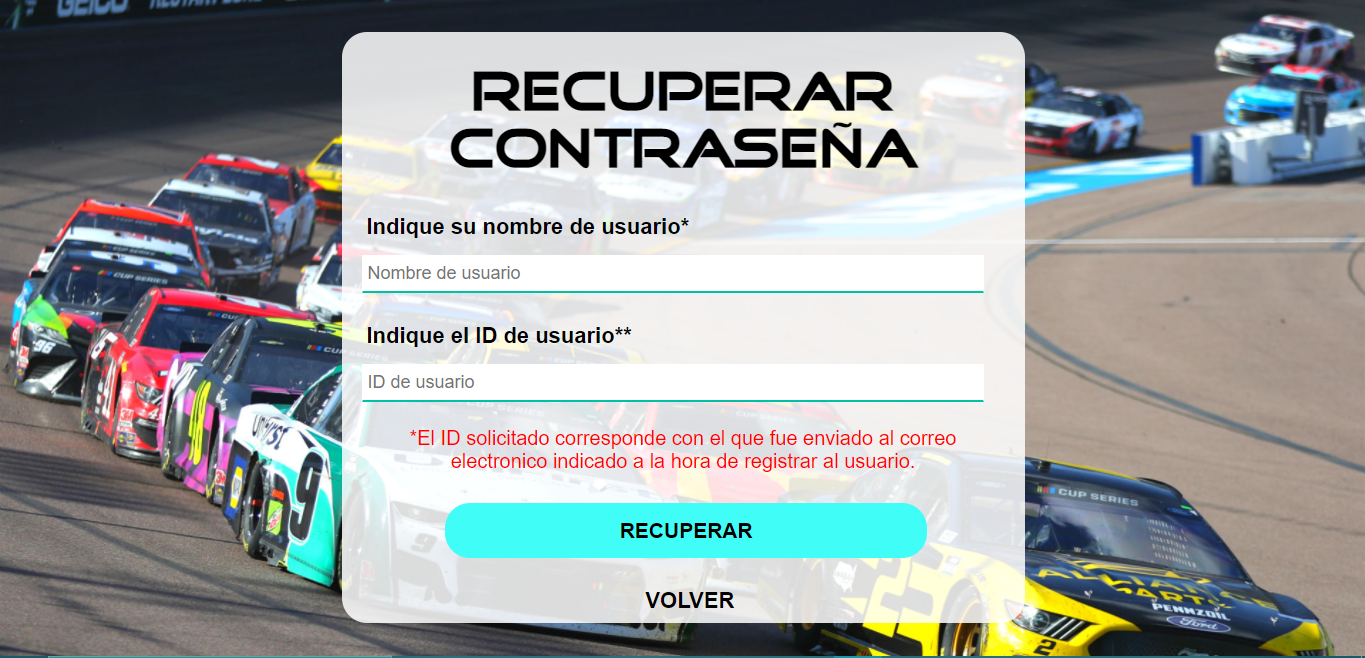
***Añadir imagen de la base de datos***

Tras iniciar el servidor web en el navegador, se accede a la página de inicio en donde se solicita al cliente sus credenciales de acceso para manejar el vehículo, pero pueden darse tres situaciones:

* El cliente dispone de las credenciales solicitadas y las introduce, mientras que por “debajo” se accede a la bases de datos donde se comprueba la validez de las credenciales, pudiendo estas credenciales ser correctas y ser así redirigido a la página de los controles, o ser incorrectas y se redirigido de vuelta al formulario de inicio.



* El cliente no dispone de credenciales para iniciar sesión, teniendo así que acceder a través de **crear usuario** al formulario de registro, en donde se solicita el correo electrónico, el nombre de usuario que desea asignarse y una contraseña. Tras rellenarlo se accede a la base de datos para comprobar que el nombre de usuario no está seleccionado, ya que si lo estuviese se le volvería a solicitar los datos de registro con la indicación de que tiene que elegir otro nombre, y si estuviera disponible se le mandaría un email al correo ofrecido para enviarle su ID de usuario.
* El cliente dispone de sus credenciales de acceso, pero no recuerda su contraseña, teniendo que acceder al formulario para recuperar la contraseña, en donde se solicita el nombre de usuario y el ID de usuario, el cual fue enviado al correo ofrecido a la hora del registro, para enviar un email a ese correo con las credenciales del usuario.



Tanto en las situaciones en la que hay que registrarse como en las que se ha olvidado la contraseña, se ha utilizado el servidor XAMPP el cual, al ofrecer servicio de PHP, también permite la mensajería electrónica a través del puerto SMTP desde el dispositivo local, correspondiente al *Protocolo de Transferencia Simple de Correo (Simple Mail Transfer Protocol)*.

Para ello se tienen que modificar en el archivo php.ini, localizado en ‘~/xampp/php/php.ini’, los comandos de la sección “mail function” indicando respectivamente:

* El servidor de correo electrónico a utilizar, Gmail o Hotmail.
* El puerto para el protocolo STMP, el cual será el 587 ya que permite envío seguro de correos, y no como el 25 que se utiliza para la retransmisión o el 465 para el envío con SSL.
* La dirección de correo desde donde se van a enviar.
* La ruta del fichero que posee la configuración de esta funcionalidad.

Después

Después de iniciar sesión correctamente

# Implementación de los sensores

# Implementación del código

# Parte 1:

## Parte 1.1:

### Parte 1.1.1:

# Parte 2:

## Parte 2.1:

## Parte 2.2: