

31-5-2022

RetroAssistant

Abel Fernández Caballero



RetroAssistent

Camino de la Miranda

2 DESARROLLO DE APLICACIONES WEB

Contenido

Introducción.....	2
Objeto del proyecto.....	2
Lenguajes empleados.....	3
Distribución.....	3
Requisitos de los Clientes.....	4
Licenciamiento.....	4
Recursos.....	4
Hardware.....	4
Software	5
Humanos.....	5
Investigación	5
Desarrollo.....	6
Testing	6
Coste del proyecto.....	6
Descripción de la aplicación.....	7
Funcionamiento general	7
Arquitectura	7
Diseño de la base de datos	8
Arquitectura del sistema.....	8
Interfaz	8
Características generales.....	8
Adaptación a dispositivos móviles	9
Usabilidad/accesibilidad	9
Autoevaluación y conclusiones	9
Valoración del trabajo y dificultades encontradas	9
Valoración de la herramienta o aplicación desarrollada	10
Conclusiones Finales.....	11
Bibliografía	11

Introducción

Objeto del proyecto

Mi Proyecto intenta solucionar un problema que suelen tener las personas con vehículos antiguos, la falta de información que sabemos de los parámetros más importantes de su vehículo ya sea porque los sensores del vehículo están averiados o porque directamente no los tienen, esto lo soluciono mediante un Arduino unos cuantos sensores y mucha programación web. Con la intención de ayudar a los propietarios de estos vehículos a seguir utilizándolos, ya sea día a día y a mantener la comunidad.

El proyecto trata de mantener la originalidad y la esencia de estos vehículos a la vez que mantener una instalación simple en los mismos, para ello se evita a toda costa modificar el comportamiento de este e intentando alterar el menor número de piezas posible.

La idea de este proyecto vino prácticamente al sacarme el carnet de conducir, ya que mi abuelo tenía un Suzuki Vitara de 1996 el cual consume mucha gasolina, pero me encanta conducir, pero nadie supo decirme cuanta gasolina consumía exactamente, me puse a pensar en cómo podía averiguarlo y como podría obtener esos datos del coche.

Este verano me puse a investigar y decidí que podría ser una gran idea como proyecto de fin de grado ya que descubrí la gran comunidad de personas con vehículos clásicos a los cuales les podría interesar.

Existen otros productos que solucionan este problema, pero tienen ciertos inconvenientes, como por ejemplo que están desarrollados para un modelo concreto o que modifican el comportamiento del vehículo o que solo muestran un dato como por ejemplo el nivel de gasolina.

Los aspectos más innovadores son:

- Mantener la integridad del vehículo
- Fácil instalación

- Precio económico
- Compatibilidad con todo tipo de vehículos y con cualquier modelo
- Tener los datos almacenados y posibilidad de obtener gráficos, para poder analizarlos y optimizar la conducción del vehículo.

Lenguajes empleados

El proyecto está programado en su mayoría en Angular, también se ha utilizado PHP, Node.js, TypeScript dentro de Angular al igual que HTML 5 y CSS 3.

Los IDEs utilizados han sido Visual Studio, Arduino IDE, con Maria DB como base de datos y Apache como servidor web, también hay que destacar el uso de Node.js con un servidor Express para obtener los datos de Arduino mediante la librería Johnny Five y el software para Arduino Firmata el cual facilita el uso de este, mediante Node.js.

Me gustaría señalar el uso de Majaro como distribución de Linux a lo largo de todo el proyecto, lo cual ha facilitado y automatizado ciertas tareas a lo largo de todo el desarrollo mediante scripts en Bash.

También he utilizado a lo largo de todo el desarrollo el sistema de control de versiones Git mediante Github lo cual me ha dado la posibilidad de restaurar el proyecto a puntos anteriores en los momentos donde me he quedado estancado.

Distribución

Actualmente la web se encuentra desplegada de forma local en un servidor Apache, en un futuro podría ser distribuida en cualquier servidor online, pero tendríamos más problemas con la obtención de datos dentro del coche, podríamos conectar nuestro teléfono móvil a la placa Arduino mediante un USB tipo b e instalar el servidor que recoge los datos en el teléfono móvil. En caso de no querer hacerlo de esta manera podríamos conectar el Arduino a un Raspberry con una pantalla para visualizar los datos podría no podríamos subir esos datos a la nube, esto lo podríamos solucionar compartiendo los datos móviles desde nuestro móvil.

La distribución de los productos podría ser de forma online mediante una empresa de transportes ya que parte del proyecto es la web de venta de este.

Requisitos de los Clientes

A la web de compra podrá acceder cualquier tipo de usuario, para comprar el producto, el usuario debe registrarse previamente, para acceder a las funcionalidades se deberá haber obtenido previamente alguno de nuestros productos y para almacenar y obtener los graficas analizando los datos se debe haber obtenido el producto “Almacenamiento en la nube” y para acceder a las opciones de administrador se deberá dar de alta como tal al usuario en la base de datos.

Licenciamiento

El producto será licenciado bajo copyleft debido a una fuerte convicción sobre el software libre y el apoyo a la comunidad y el apoyo que el producto puede recibir de la comunidad, ya que mi intención no es distribuir realmente el proyecto, es algo mucho más personal sin intención de un lucro real

Recursos

Hardware

Debido a las peculiaridades del proyecto necesitaremos más hardware que otros proyectos,

- Placa Arduino o compatibles
- Sensores varios
 - Caudalímetro YF-201
 - Sensor de efecto Hall
 - Sensor de ultrasonidos HC-SR04
- Cables de conexión dupont

- Protoboard
- Cable USB tipo B a USB tipo A o tipo C (para el móvil)
- Smartphone u Ordenador

Software

El software necesario ha sido:

- Majaro como sistema operativo
- Node.js
- Express
- PingFirmata
- Apache
- MariaDB
- PHPMyadmin
- Angular
- Johnny Five
- Visual Studio
- Arduino IDE
- Firefox o cualquier navegador

Humanos

El número de horas invertidas en este proyecto son muy elevadas, y es muy difícil de calcular, y las podemos desglosar en 3 fases

Investigación

La investigación ha sido el puto en el que más tiempo he tenido que invertir debido a que nunca había utilizado Arduino ni ningún sistema parecido, también he tenido que investigar sobre cómo obtener los datos que quería de un coche, y utilizar Express dentro de node.js para crear el servidor y como enviar los datos al cliente, esta fase puede llevarme más de 50 horas entre pura investigación y pruebas

Desarrollo

El desarrollo también ha llevado mucho tiempo, pero ha sido más llevadero y con menos quebraderos de cabeza y sobre todo más fluido, esta fase me ha podido llevar sobre unas 40 45 horas, sin contar lo desarrollado en el trabajo final para la asignatura de Desarrollo en el entorno del Cliente, sumando todo ha podido llevar más de 50 55 horas.

Testing

Esta ha sido la última fase y debido a la gran inversión en tiempo en las anteriores me ha quedado menos tiempo para ello, pero las podemos estimar en unas 10 horas solucionando pequeños errores y probando todos los aspectos del proyecto. También esta fase ha coincidido con mis ultimas semanas de estancia en Irlanda gracias a la beca Erasmus y he tenido menos tiempo del previsto.

Coste del proyecto

En la fase de desarrollo se podría estimar un coste de unos 100€ contando todo el hardware empleado, lo cual me parece un coste más que razonable debido a los resultados obtenidos.

El coste de los productos podría rondar sobre unos 120€, teniendo en cuenta que el coste de los materiales rondara los 60€ lo cual podría dejar un gran margen de ganancias, dando al usuario un producto a un precio muy atractivo, razonable y novedoso.

Teniendo en cuenta que los productos sustitutivos del nuestro se suelen vender pasando la barrera de los 1000€ es un precio muy muy competitivo que animaría a muchos usuarios a adquirirlo.

Descripción de la aplicación

Funcionamiento general

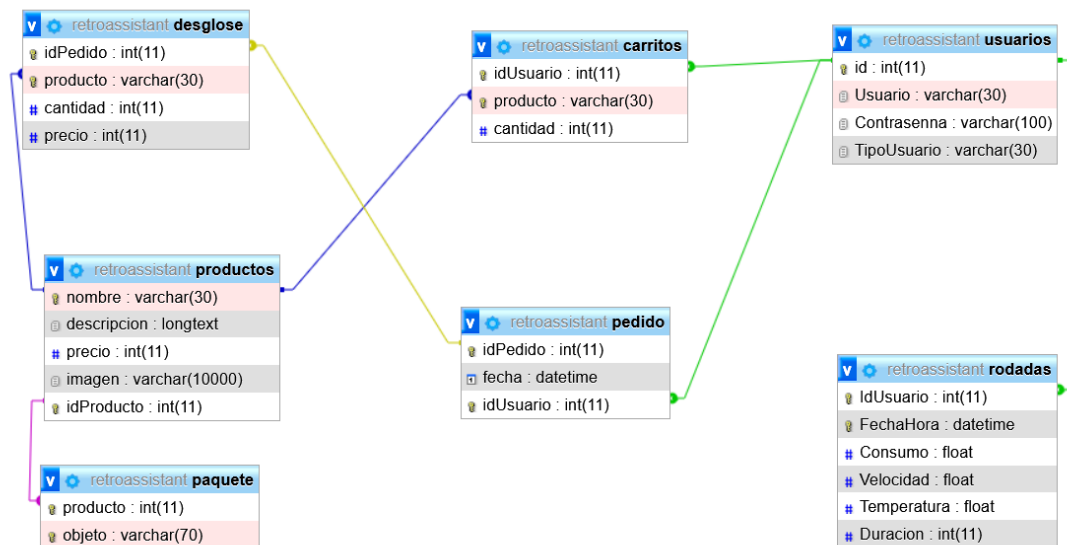
El proyecto, a grandes rasgos, se constituye de 2 partes componentes, una web de compra de una serie de productos, y la visualización de datos del vehículo, que están interrelacionadas mediante la misma web de compra del producto.

Diagrama de casos de uso



Arquitectura

Diseño de la base de datos



Arquitectura del sistema

He decidido no utilizar clases para el desarrollo del proyecto debido que los lenguajes empleados para ello no son orientados a objetos y aunque exista la posibilidad de hacerlo no he visto demasiadas ventajas en ello debido a que los datos que manejamos no tienen una gran interacción entre si más que insertarlos en la base de datos, mostrarlos en pantalla, actualizarlos y eliminarlos. Y debido a que me llevaría más tiempo programar las clases que las ventajas que me aportaría he decidido no utilizarlo.

Interfaz

Características generales

La interfaz tiene un estilo lo más minimalista posible para mantener la tranquilidad y no molestar al usuario con estilo innecesarios ni colores estridentes, debido a que eso a la larga provoca una menor interacción con el usuario y una menor capacidad de mantener su atención.

La interfaz está diseñada de forma en la que no se necesita cambiar de página mediante un desarrollo SAP (Single Application Page) mediante router outlets de angular para una mayor fluidez en el uso de la aplicación de cara al usuario.

Adaptación a dispositivos móviles

La web de compra en un principio está pensada para ordenador, pero es perfectamente usable en dispositivos móviles, teniendo en cuenta siempre la adaptación a los mismos mediante el uso de flex en css3.

En cambio, la parte de la web de visualización de los datos del coche esta más orientada a dispositivos móviles, debido a que va a ser su uso principal, pero obviamente también es perfectamente usable en ordenadores personales.

Usabilidad/accesibilidad

He procurado utilizar colores muy diferenciados y suaves para facilitar el uso a personas con problemas de visión, he intentado optimizar el uso del servidor y utilizar el menor número de llamadas posibles para acelerar lo máximo posible las conexiones y mejorar la experiencia de uso a los clientes con problemas de conexión o conexiones lentas.

Autoevaluación y conclusiones

Valoración del trabajo y dificultades encontradas

Ha sido un trabajo muy extenso con una gran parte de investigación que ha llevado más tiempo del que en un principio había pensado, pero a la vez ha sido muy estimulante y a pesar de ser mas largo y complejo de lo pedido ha sido muy

interesante y motivador realizar este proyecto, debido a que el propósito ha sido muy personal y que resuelve problemas que realmente tengo en mi día a día, también ha ayudado mucho al diseño de todo el sistema y ya que es algo que tenía muy interiorizado y no he tenido que informarme tanto sobre la estructura de los datos o sobre cómo funciona.

Las dificultades encontradas han sido básicamente la falta de tiempo debido a que 3 meses es muy poco tiempo para desarrollar un proyecto así, aunque empecé antes de terminar el curso ya que suponía que, sino no me iba a dar tiempo, también ha sido una dificultad añadida estar realizando las prácticas en Irlanda ya que a la falta de tiempo y a la falta de un lugar donde vivir las primeras semanas, se le suman las ganas de conocer la cultura y tener un gran Erasmus. También ha supuesto un problema la investigación y el aprendizaje a pasos forzados de Arduino ya que nunca había utilizado esta herramienta y es muy compleja con un gran abanico de posibilidades.

Valoración de la herramienta o aplicación desarrollada

La herramienta a mi parecer resuelve un gran número de problemas que pueden tener los usuarios de vehículos clásicos o antiguos, lo que le da una buena perspectiva de futuro, también es una herramienta muy expandible por lo tanto se le pueden aplicar nuevos sensores y desarrollar con poco esfuerzo nuevas funcionalidades.

Estoy muy contento con el resultado, obviamente siempre hay cosas que se pueden mejorar como el diseño de las interfaces y algunos pequeños puntos del diseño de módulos en angular que no se han hecho de la mejor manera posible pero debido a que es parte del aprendizaje de esta herramienta.

También debido a la falta de tiempo se han quedado cosas en el tintero como nuevas funcionalidades o mejoras en el diseño de las interfaces.

Conclusiones Finales

Personalmente estoy muy contento con el resultado del trabajo y sobre todo con todo lo aprendido en el proceso, ya que no ha sido solo aplicar los conocimientos obtenidos en el curso sino muchísimo aprendizaje y obviamente también el refuerzo y la puesta en practica de los conocimientos obtenidos durante el curso es muy satisfactorio.

Puntos Para Mejorar

Como en todos los proyectos con un tiempo tan limitado existen putos que podría mejorar, pero se imposibilita por la falta de tiempo

- Crear la posibilidad de que un usuario tenga varios productos para varios coches
- Añadir sensores como un termómetro y un sensor de aparcamiento trasero y delantero
- Montar el producto en un coche de pruebas
- Mejorar el caudalímetro a uno con más sensibilidad
- Agregar un panel de instrumentos “analógico”
- Mejorar un par de puntos dentro del diseño de módulos de Angular un poco redundantes
- Cambiar la obtención del tiempo de duración de una rodada a minutos en vez de segundos
- Cambiar la contraseña de un usuario mediante un email de confirmación

Bibliografía

- [Repositorio con diferentes sensores e implementación con Johnny-five sobre todo Pingfirmatra](#)
- [Api con información sobre cómo obtener información de sensores mediante JavaScript por medio de Firmata y Arduino](#)
- [Foro con una librería para obtener datos de un caudalímetro](#)
- [Video donde se explica cómo crear una API REST con Node.JS y Express](#)
- [Documentación técnica del caudalímetro YF-S201](#)
- [Hoja de datos del caudalímetro](#)
- [Para crear graficas con JQuery](#)