

ShoalTrack2017

Monitorización de vehículos para la
II Regata Solar de
MarineInstruments



II Regata Solar (2017)

- **Motivaciones**

- El propósito es el diseño y construcción de una embarcación eléctrica de radio control propulsada únicamente por energía solar.
- La motivación es incentivar las inquietudes de los alumnos por la tecnología y las energías renovables, respetando el medio ambiente.

- **Los datos**

- La empresa Marine Instruments organiza por segundo año consecutivo la Regata Solar en el Monte Real Club de yates de Baiona.
- Participarán 20 equipos de 9 colegios diferentes de ESO/Bachillerato y formación profesional.
- Concursarán con sus propias embarcaciones en tres pruebas distintas.
- Cada colegio diseñará y construirá su propia embarcación eléctrica de radio control propulsada únicamente por energía solar.



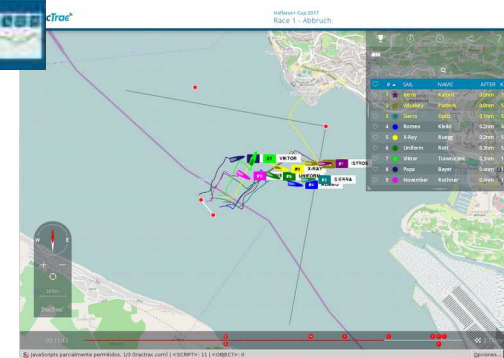
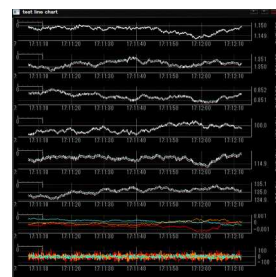
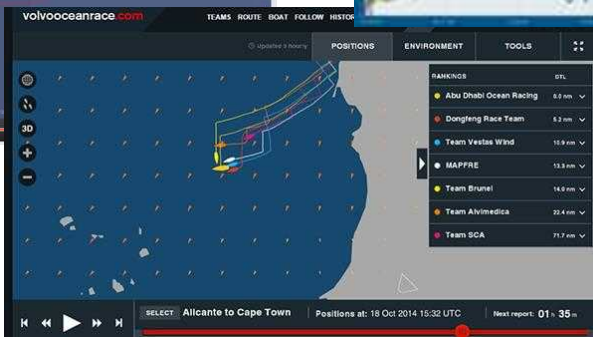
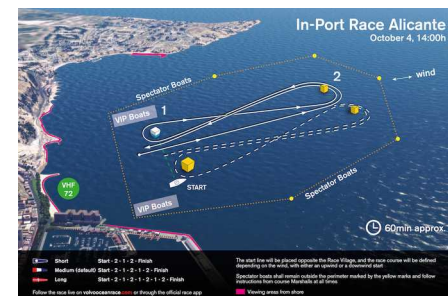
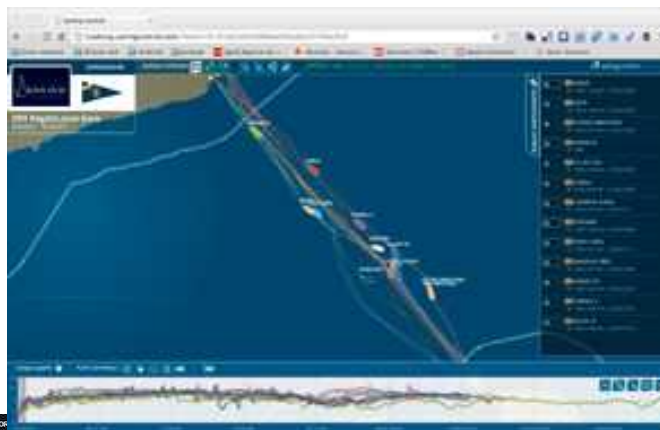
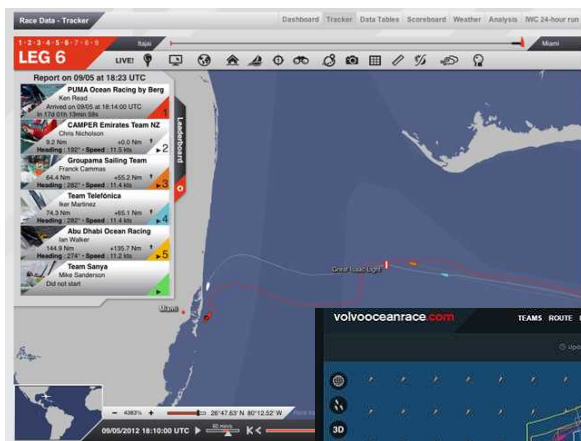
ShoalTrack2017

Propuesta

Requerimientos

- Sistema integrado de monitorización de barcos, proporcionando los sensores y el software de visualización.
- Este sistema permitirá conocer en tiempo real la posición de cada embarcación así como otros parámetros claves como la velocidad de desplazamiento, la estabilidad, la potencia instantánea o la intensidad de luz incidente.

Ejemplos



ShoalTrack2017

Especificaciones

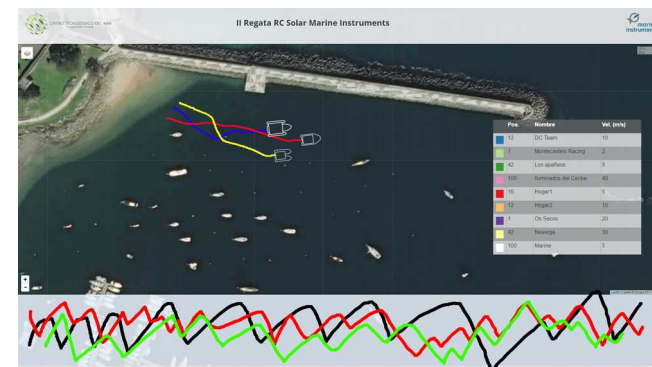
• Hardware

- Sistema de tracking y monitorización de variables
- Sistema de adquisición de datos de la posición y variables de temperatura, estabilidad, luz incidente y gasto energético
- Comunicaciones LoRa



• Software

- Visualización de los datos para el público
- Un sistema que a través de la web pueda mostrar la posición de los barcos, así como sus estelas.
- Que sea seleccionables los barcos en un frame lateral
- Que en un frame inferior se tenga una gráfica de las velocidades, posición en la regata



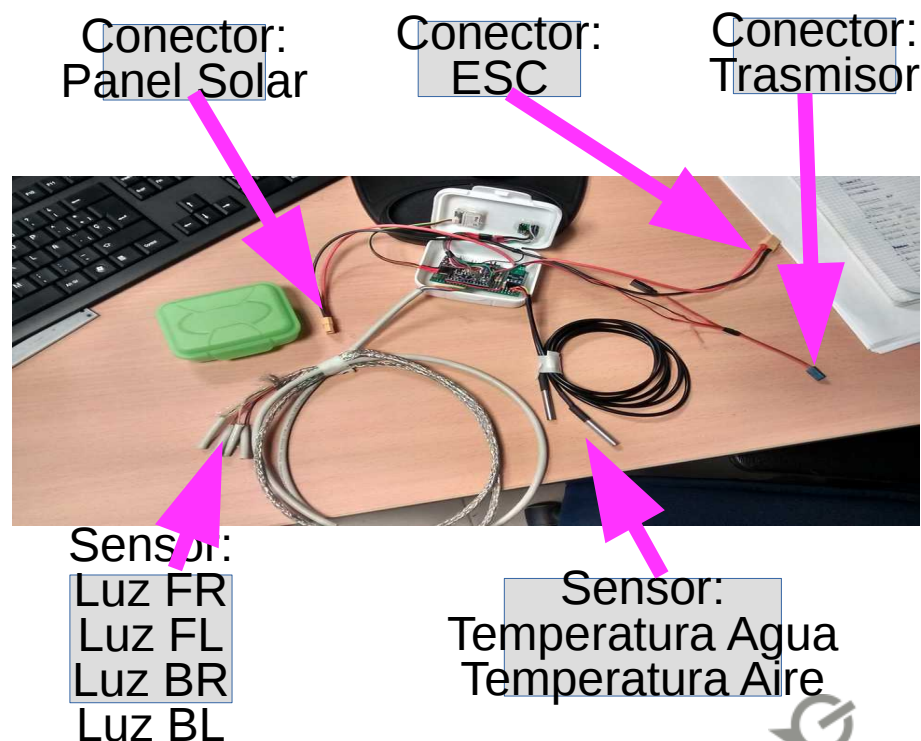
ShoalTrack2017

Hardware

Nodo Cliente

- Los barcos pertenecientes a la categoría CM/CS llevar instalado un sistema de monitorización proporcionado por la organización.
- una placa integrada con sensórica que se alimenta directamente del rail de 5v del ESC a través del conector estilo servo. Y puentea la conexión entre el panel solar y el ESC en la etapa de potencia.
- 95x70x35 mm

Nodo	
Componente	Descripción
Regulador 5 → 3.3v	Regulador 4.5V-7V to 3.3V AMS1117-3.3V para alimentar el transmisor LoRa
Arduino Pro Mini 16Mh 5v	Microcontrolador Atmel ATmega328 de la familia Arduino de 2KB RAM y 8 de ROM
Transmisor LoRa RF430	Transmisor/Receptor SX1278 long range RF wireless module DRF1278F en la frecuencia de los 430Mhz y tecnología LoRa
GY-91 MPU9250 BMP280	Circuito integrado con los sensores MPU9250 (acelerómetro, giroscopo y magnetómetro) y sensor de presión BMP280
GPS Ublox M8N	GPS con antena integrada
DS18B20	Sensor de temperatura digital con protocolo OneWire
ACS712	Sensor de Corriente +-20A
LDR	Sensor de luminosidad GL5549



ShoalTrack2017

Hardware

- **Nodo Master**
 - Sistema encargado de recibir la información de los nodos y transmitirla al PC
 - Sistema de referencia de GPS (para hacer correcciones)
 - Estación metereológica

Nodo	
Componente	Descripción
Regulador 5 → 12v	Elevador de tension a 12v para alimentar la estación metereológica
Arduino DUE 5v	Microcontrolador Atmel ATmega328 de la familia Arduino de 2KB RAM y 8 de ROM
Transmisor LoRa RF430	Transmisor/Receptor SX1278 long range RF wireless module DRF1278F en la frecuencia de los 430Mhz y tecnología LoRa
GPS Ublox M8N	GPS con antena integrada

ShoaTrack2017

Firmware

- **Nodos**

- Entorno ArduinoIDE
- Libreria de comunicaciones RadioHead
- Sistema de cifrado ChaCha20
- Parámetros a transmitir:
 - GPS: Longitud, Latitud
 - IMU: Acx, Acy, Acz, Gyx, Gyy, Gyz, Mgx, Mgy, Mgz
 - Barómetro: presión
 - Temperatura: Aire, Agua, Interior (Caja de electrónica)
 - Intensidad Luz: Luz delantera derecha, Luz delantera izquierda, Luz trasera derecha, Luz trasera izquierda
 - Voltaje: del panel solar
 - Amperios: del panel solar

- **Master**

- Ciclo request/response para cada Nodo (1 seg de espera)
- Aproximadamente 1seg por ciclo (si todo es correcto)

ShoalTrack2017

Software

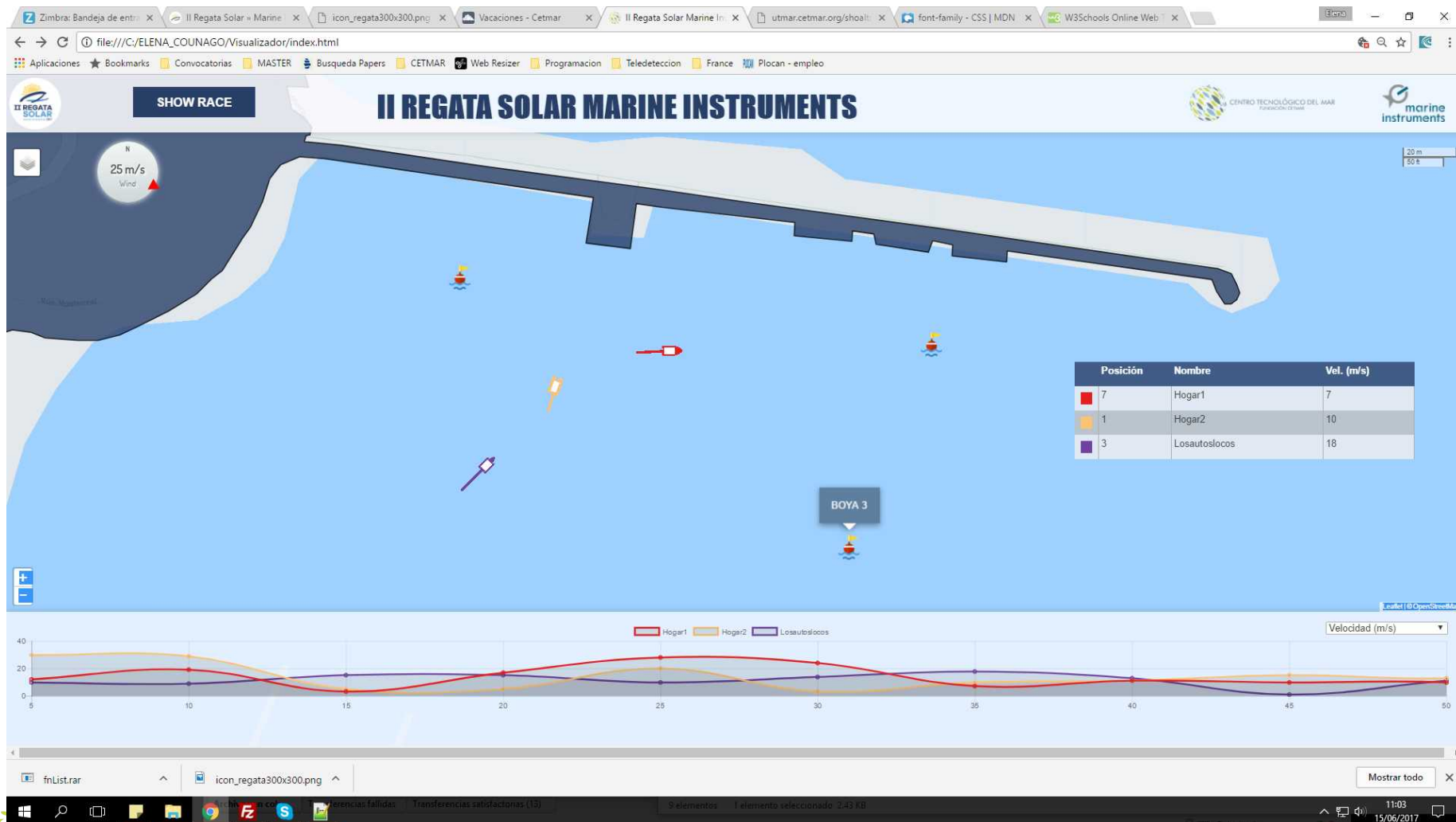
- **Gestión de datos**

- Sistema que a través de la web se pueden acceder a los parámetros de los dispositivos y cambiarlos en tiempo real.
- En esta interfaz web se podrán cambiar los parámetros de calibración de los diferentes sensores. Si bien, se proporcionarán unos valores predefinidos, en la fase de pruebas antes de la carrera se pueden cambiar para ajustarlo mas a valores mas precisos.
- La calibración puede tener 4 tipos de curvas características:
 - Lineal
 - Exponencial
 - Potencia
 - Logarítmica

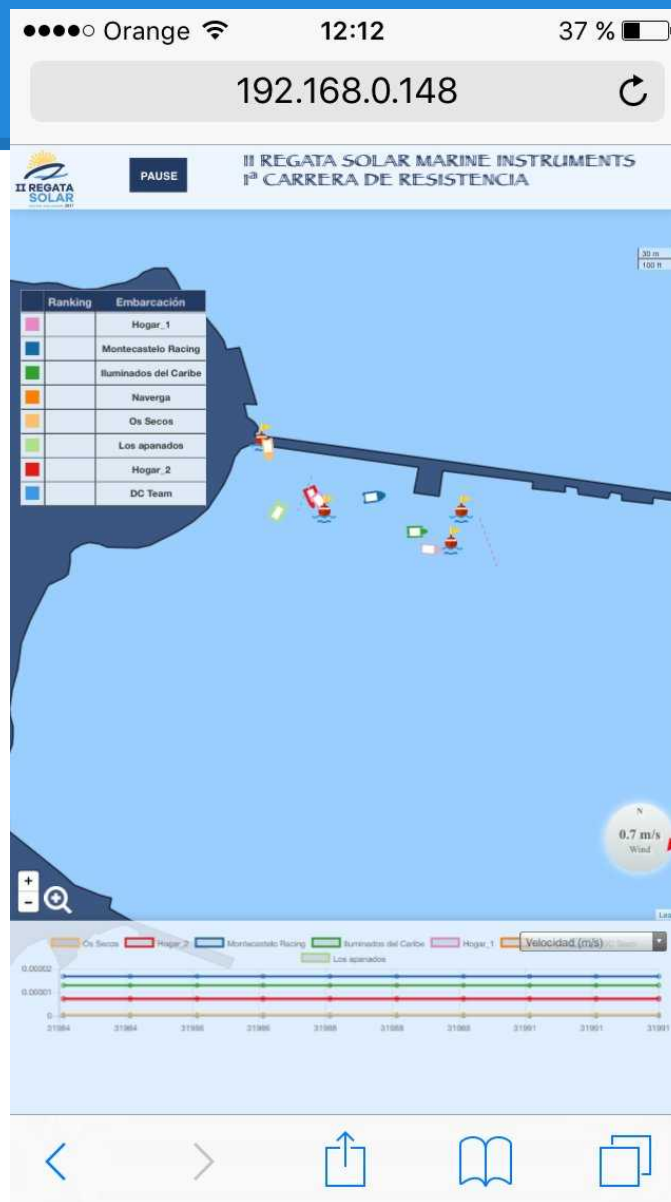
- **Visualización**

- Un sistema que a través de la web puede mostrar la posición de los barcos, así como sus estelas.
- Y que tiene un frame inferior donde se muestra una gráfica de las velocidades, posición en la regata y potencia instantánea.
- Además, se mostrará también el Modulo de viento y su dirección en cada momento.
- En la tabla lateral se mostrará un listado de los barcos, con sus propiedades (nombre, dueño,color): ordenados por posición

ShoaTrack2017 Software



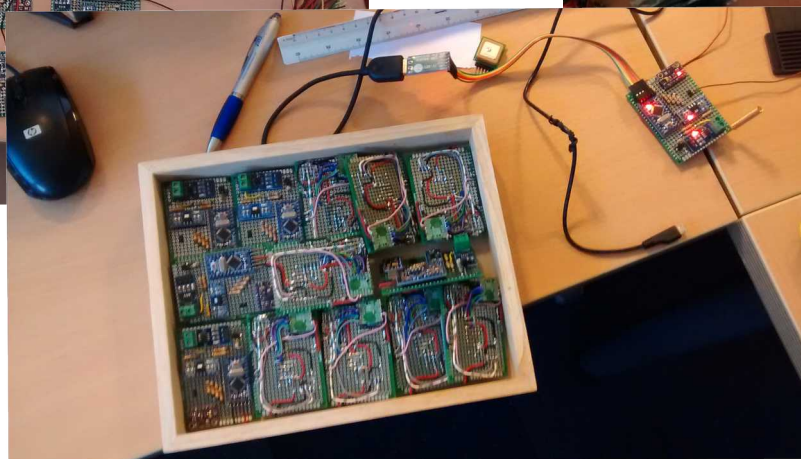
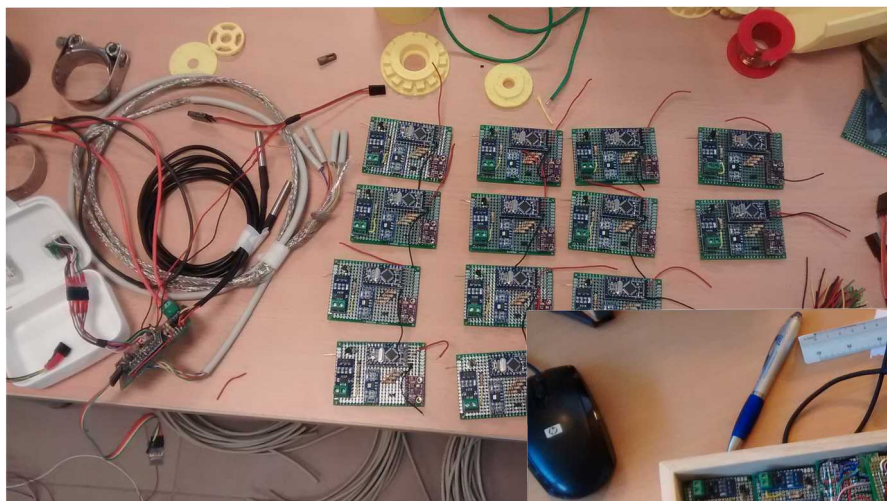
ShoalTrack2017



ShoalTrack2017

Montaje

- Producción de 12+1 unidades
 - Compra de componentes a TxHang (eBay)
 - Cesión de GPS Ublox (MarineInstruments)
 - Soldar, comprobar y cargar Firmware



ShoalTrack2017

Test Final

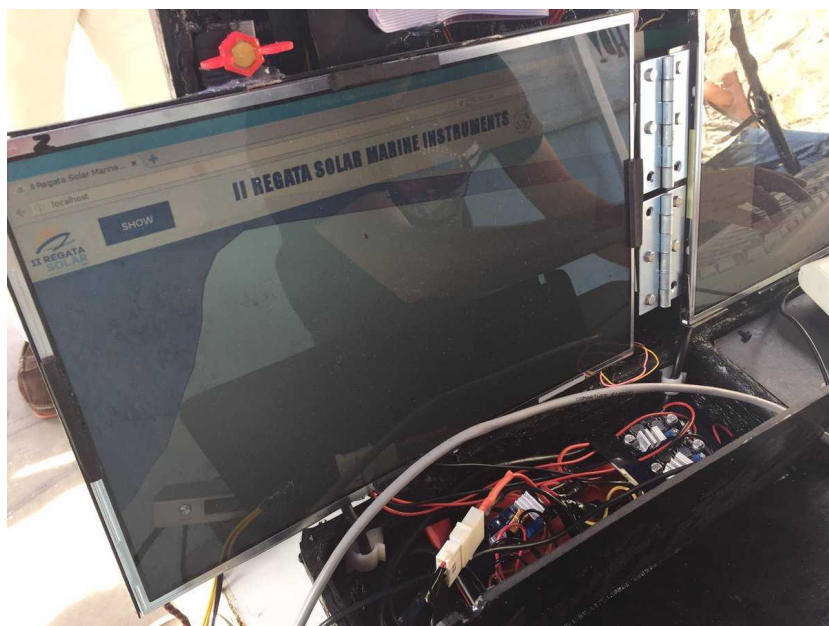
- Prueba real de todos los componentes
 - Prueba de todos los Nodos transmitiendo
 - Prueba de carrera alrededor del parking de Bouzas.
 - Test de alcance



ShoalTrack2017

5 min antes...

- Sistema completo
 - Maleta de Control
 - Trípode con estación meteorológica
 - Punto de acceso para ver datos
 - Visualización de web en RT en televisor



ShoalTrack2017

Durante la carrera...

- Monitorización
 - 1 nodo por embarcación
 - 1 nodo por boya



ShoalTrack2017

Post-Mortem

• Hardware

- El Gateway-master estaba configurado para pedir 14 nodos: de los cuales solo 10 estaban activos. → Eso produjo retardos en la llegada de datos. Fue corregido en la ultima carrera, pero aun así, había una que no estaba activo. [Cada nodo No-Activo produce 200 ms de retardo]
- Los equipos montaron el dispositivo DEBAJO de la placa solar y a veces, este estaba sumergido en el agua (porque tenían poco calado) → No advertimos eso en las instrucciones de instalación, así que tampoco es culpa de los colegios.
- Repasando las librerías RadioHead y el hardware LoRa:
 - Tenia puesto un WAIT de 1000 milisegundos.
 - El Hardware se configura no con los bps, sino con el ancho de banda ocupado y su modulación -> a mas velocidad, menor distancia... (max 300kbps) pero nunca puedes seleccionar la velocidad: se ajusta automáticamente podría ser que una transmisión complicada, durase hasta 2 segundos...
 - Por lo tanto, al ir degradándose las conexiones, estas empezaron a consumir el tiempo: y además, el protocolo exigía ACK, por lo que duplicamos el tiempo de cada petición (hay que devolver) y como existe: peticion-respuesta, con sus respectivos ACK, Cada trasmisión podría consumir hasta 4 segundos...
 - De ahi el lag que hay entre los datos de cada dispositivo: a veces de 20-30 segundos

• Software

- La gestión de estados de una carrera no pudo ser implementada a tiempo. → por falta de tiempo
- En la ultima carrera se ralentizo la visualización → Podría deberse a la alta conexión de dispositivos al servidor y provoco que todo se ralentizase
- El entorno Django Admin de la gestión de la DB, hacia peticiones a la web de openlayers.org.. pero como no tenia internet, quedaba 2 minutos esperando la petición → Hizo inusable la gestión de la web
- No se podía cambiar la posición del Gateway en la web... porque no funcionaba la capa de openlayers → Esto sirve para recalcular las posiciones GPS [DGPS] y hacerlos mas precisos. Al final lo cambie manualmente y las posiciones de los barcos se volvieron correctas..

ShoalTrack2017

Post-Mortem

Para adaptar el código para poder mostrar los datos en el visualizador, se crea una variable que por defecto se inicializa todas las sub-variables a 0...

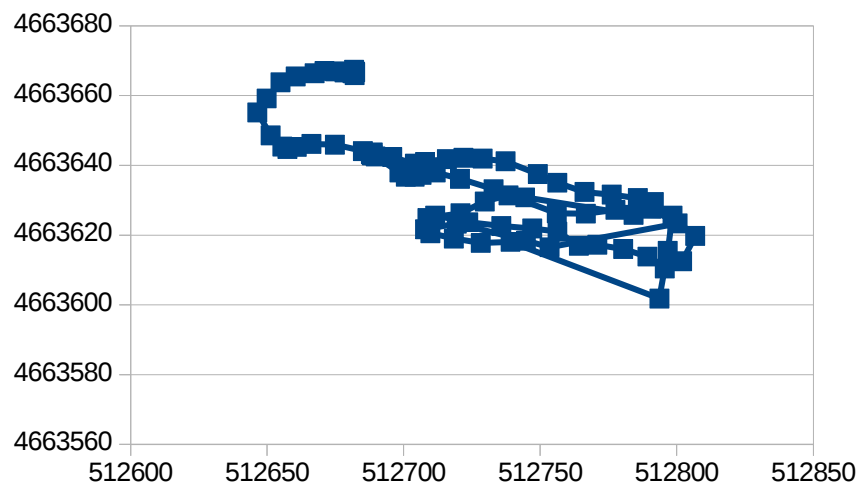
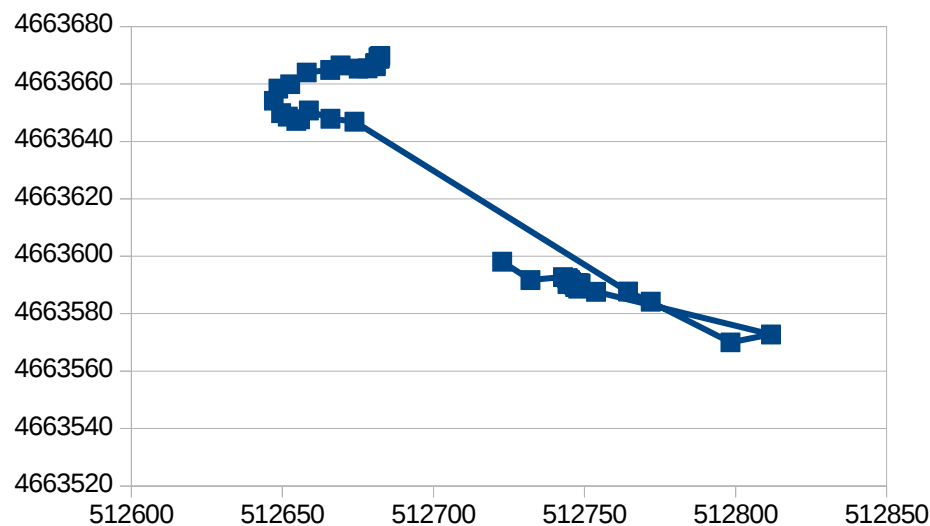
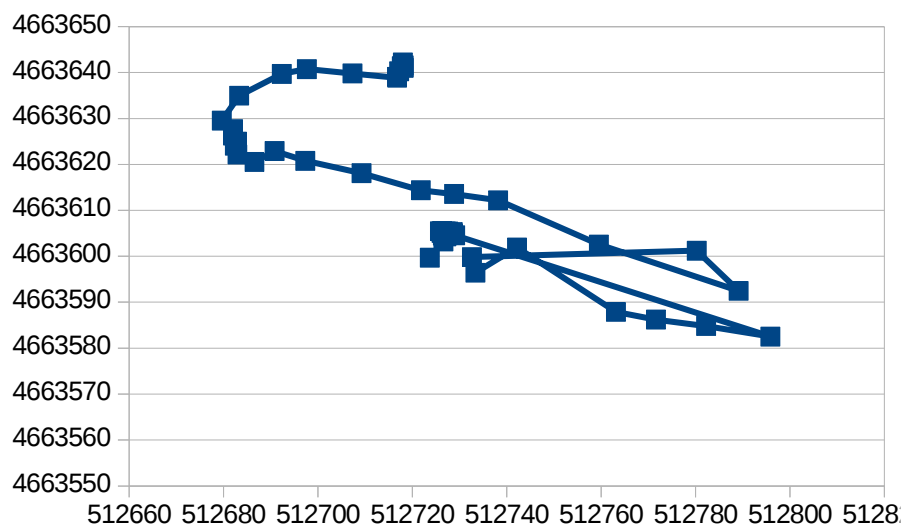
Ahí se graban los datos antes de ser grabados a un fichero y salvados a la DB....

pero se me olvido de copiar los valores RAW de cada sensor.. por lo que los datos que se guardaron en la DB, que son los RAW, quedaron siempre a ZERO.

Por contra, los datos del visualizador si se veían bien.

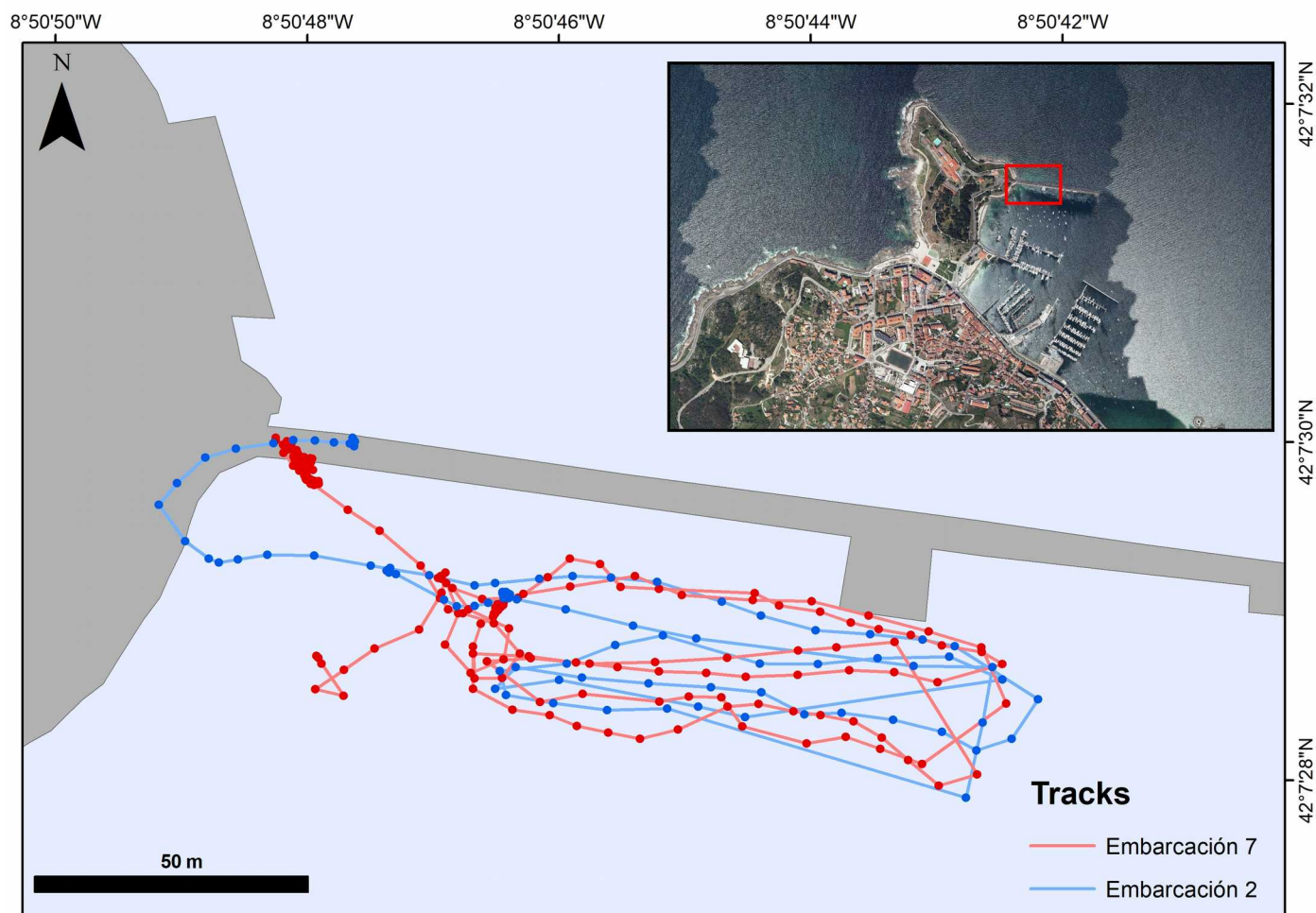
ShoalTrack2017

Post-Mortem



ShoalTrack2017

Post-Mortem



ShoalTrack2017

ShoalTrack2018

Corregir errores

- Visualización
- Trasmisión de datos

Mejorar accesibilidad

- Nueva base de datos
- Tecnología Push/websockets
- Android App

Reconocimiento Imagen

- OpenCV
- Realidad Aumentada

Otras??



Fecha

15 Junio

ShoalTrack2017

ShoalTrack2018

Hardware Nodo (arduino mini):

Dejar solo una adquisición de Luz
meter control de un anillo led RGB [para enviar señales desde el centro de control y así permitir cambiar la luz según el estado en el que este el barco: penalizado, eliminado, primero, segundo...] WS2812 5050 RGB 4 unidades
Añadir un RTC real para la sincronización en arranque (y sin GPS) de tiempo, -> la seguridad del protocolo ante ataques de repetición

Cambiar la frecuencia de adquisición de los parámetros:

IMU: 20Hz (se guarda la media y la desviación estándar por cada segundo)
GPS: 1Hz
Barómetro: 5Hz (se guarda la media y la desviación estándar por cada segundo)
Temperatura: 1Hz
Intensidad Luz: 5Hz (se guarda la media y la desviación estándar por cada segundo)
Voltaje_panel_solar y Amperios panel_solar: 10Hz (se guarda la media y la desviación estándar por cada segundo)

Estructura del Payload del paquete:

Añadir comprobación de autentificación: chacha20 + Poly1305 (16bytes??!!CHECHSUM, esto tengo que ver como reducirlo..)
Intentar ver si se puede comprimir la trama -> crear un nuevo tipo de trama diferente de mavlink para poder comprimir trama y compactar.

Protocolo red:

Estructura de red Mallada a través de la librería RADIOHEAD.
Es necesario primero saber el camino al destino (haciendo broadcast encadenados) y luego el envío de paquetes se envía a través del path

Gateway:

Que no decodifique el paquete: simplemente lo reenvía con la IPv6 real a un servidor central (o al pc conectado).
Lleva un registro de las direcciones conectadas y su relación address/ipv6 [un ruteo NAT de toda la vida..]
Lleva un registro de la estructura mesh bajo su influencia -> esto aun tengo que pensarlo como

Almacenamiento:

Script que obtiene los datos, busca su keysecret, comprueba la autenticidad, y guarda los datos en una tabla TimeSeriesDB de PostgreSQL.
Incluye aquí la gestión de permisos y privacidad.

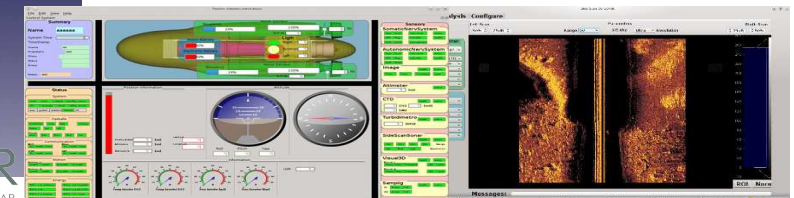
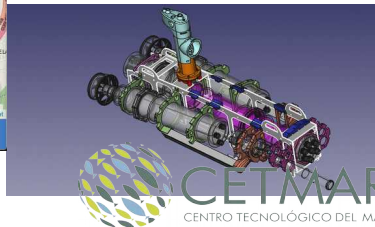
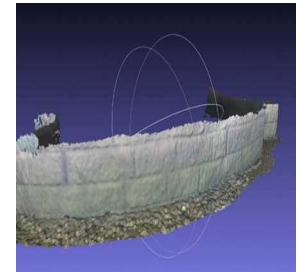
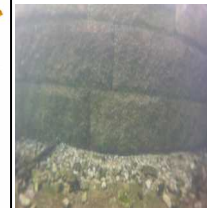
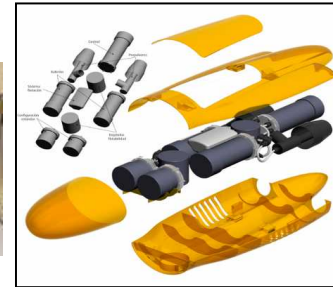
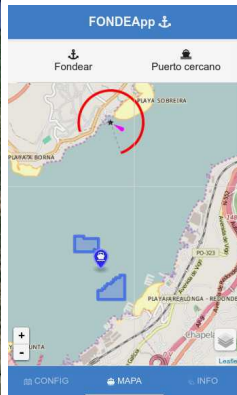
Si no tengo el keysecret, puedo reenviar el paquete a otro sitio (que se especifique en la DB) aumentando la privacidad de los datos (lo envías al sitio del cliente o se pueden almacenar para su decodificación posterior..) -> solo decodifica el paquete aquel que tiene las claves.

Trabajo de master: xDxD

Con este nuevo formato se podrían hacer test de funcionamiento y generar estadísticas de paquetes enviados/recibidos, en relación a la posición GPS, número de nodos activos, tamaños de tramas, problemas de colisiones, latencia cuando hay nodos entrando o saliendo del mesh...

UNIDAD DE TECNOLOGÍAS MARINAS (UTMAR)

- **Ámbitos de especialidad**
 - Diseño, instalación y mantenimiento de Redes de Observación Oceanometeorológicas.
 - Evaluación y desarrollo de nuevas tecnologías de monitorización. Laboratorio de Calibración.
 - Desarrollo de soluciones tecnológicas específicas.
- **Actuaciones**
 - Proyectos de I+D+i de ámbito regional, nacional e internacional.
 - Desarrollo de asistencias técnicas y servicios.



Contacto



Unidad de Tecnologías Marinas

www.cetmar.org



@UTMAR_CETMAR

Silvia Torres

storres@cetmar.org

UTMAR utmar@cetmar.org