

Matxoclock: sistema de cronometratge per a una cursa de muntanya

Enginyeria Tècnica en Informàtica de Sistemes

Estudiant

Marc Font Codina

Consultor

Jordi Bécares Ferrés

Data de lliurament

20/06/2015

Resum

La mesura precisa del temps en esports de velocitat esdevé un factor crític. El propòsit del projecte és crear una solució de cronometratge en temps real per a curses de muntanya de llarga distancia amb accessos i comunicacions difícils. Similar a les que s'ofereixen en els sistemes de cronometratge comercials però utilitzant només components de baix cost.

Per verificar-ne la viabilitat es selecciona una cursa de llarga distancia per fer les proves de camp: Pels camins del matxos, 63Km amb 11 punts de control i una durada màxima de 15h. Aquesta cursa presenta uns requeriments ideals en termes de: autonomia, mobilitat, dificultat de comunicacions i volum de dades.

Per donar solució a tots els requeriments s'han dissenyat i fabricat dispositius de captura basats en diferents tecnologies: RFID, Arduino, Raspberry Pi o GPRS. També s'han creat artefactes de programari per gestionar, emmagatzemar i presentar les dades capturades.

El sistema de cronometratge dissenyat permet identificar inequívocament als participants, contenir els seus temps parcials en un sistema de persistència i, si la infraestructura de comunicacions ho permet, publicar les dades en temps real. Disposar de les dades a mesura que es capturen és una funcionalitat que dóna molt valor afegit des del punt de vista de l'organització, participants i públic.

La implementació del projecte ha proporcionat una solució totalment operativa i ha permès establir les bases de cara a futures implementacions amb noves funcionalitats.

Paraules clau: cronometratge, *datalogger*, temps real, Arduino, Raspberry Pi

Àrea del TFC: Sistemes encastats

Índex de continguts

1.	Introducció.....	1
1.1.	Justificació	1
1.2.	Descripció del projecte	2
1.3.	Objectius del TFC	4
1.4.	Marc del projecte	5
1.5.	Enfocament i mètode seguit	6
1.6.	Planificació del projecte	7
1.7.	Recursos emprats	8
1.8.	Productes obtinguts	8
1.9.	Breu descripció dels propers capítols.....	9
2.	Antecedents.....	10
2.1.	Estat de l'art.....	10
2.2.	Referències d'empreses de serveis de cronometratge	11
2.2.1.	Novotiming	11
2.2.2.	Temps x curses.....	11
2.2.3.	Livetrail	12
2.3.	Estudi de mercat.....	12
2.3.1.	Unitats de control.....	13
3.	Descripció funcional	16
3.1.	Introducció.....	16
3.2.	Matxoclock: sistema de cronometratge per a una cursa de muntanya.....	16
3.3.	Funcionalitat de captura.....	17
3.4.	Funcionalitat d'emmagatzematge.....	18
3.5.	Funcionalitat de consulta	19
3.6.	Resum funcional	20
4.	Descripció de la implementació	22
4.1.	Introducció.....	22
4.2.	Funcionalitat de captura.....	22
4.2.1.	Introducció.....	22
4.2.2.	Decisions de disseny perennes.....	22
4.2.3.	Arquitectura de programari.....	23
4.2.4.	El primer prototip, ArduinoDLv1	25
4.2.5.	Datalogger amb funcionalitats de xarxa, ArduinoRT.....	29
4.2.6.	ArduinoDLv2, un ArduinoDLv1 amb pantalla	32
4.2.7.	rPi-eth, una nova plataforma de treball.....	33

4.2.8.	rPi-GPRS, el dispositiu final.....	38
4.3.	Funcionalitat d'emmagatzematge.....	40
4.3.1.	Interfície d'entrada.....	41
4.3.2.	Càrrega massiva.....	42
4.3.3.	Contenedor.....	42
4.3.4.	Publicació.....	42
4.3.5.	Validació de la implementació.....	43
4.4.	Funcionalitat de consulta.....	43
5.	Configuració de la prova de camp.....	45
6.	Viabilitat tècnica.....	47
6.1.	Viabilitat tècnica dels dispositius de control.....	47
6.2.	Viabilitat tècnica del web en temps real.....	48
7.	Valoració econòmica.....	50
7.1.	Costs del projecte.....	50
7.2.	Viabilitat econòmica.....	51
8.	Conclusions.....	53
8.1.	Assoliment d'objectius.....	53
8.2.	Incidències.....	54
8.3.	Proposta de millores.....	55
8.4.	Autoavaluació.....	58
9.	Glossari.....	59
10.	Bibliografia.....	60
10.1.	Referències.....	60
10.2.	Comunitats.....	61

Índex de figures

Figura 1.1 - Localització geogràfica del recorregut de la cursa.....	3
Figura 1.2 - Esquema de punts de control.....	4
Figura 1.3 - Primera planificació presentada.....	7
Figura 1.4 - Planificació final.....	7
Figura 2.1 - Comparativa de característiques tècniques d'unitats de control.....	14
Figura 3.1 – Funcionalitat bàsica del sistema.....	16
Figura 3.2 - Diferenciació temporal a l'hora d'emmagatzemar les dades.....	16
Figura 3.3 - Diagrama funcional de la solució.....	17
Figura 3.4 - Flux basic de la funcionalitat de captura.....	17

Figura 3.5 - Funcionalitat de captura.....	18
Figura 3.6 - Flux bàsic de la funcionalitat d'emmagatzematge	18
Figura 3.7 - Funcionalitat d'emmagatzematge.....	19
Figura 3.8 - Flux bàsic de la funció de consulta	20
Figura 3.9 - Funcionalitat de consulta	20
Figura 3.10 - Diagrama funcional detallat de la solució	21
Figura 4.1 – Procés iteratiu de millora contínua	22
Figura 4.2 - Diagrama d'execució a alt nivell	24
Figura 4.3 - Cas d'us de la posada en funcionament	24
Figura 4.4 - Cas d'us de captura de dades	25
Figura 4.5 - Funcionalitats de captura parcials del model ArduinoDLv1	26
Figura 4.6 - Comparativa entre bateries de plom i d'ions de liti	27
Figura 4.7 - Càlcul de potència elèctrica dels components del ArduinoDLv1	27
Figura 4.8 - Consum de l'ArduinoDLv1	27
Figura 4.9 Diagrama modular del ArduinoDLv1	28
Figura 4.10 - Arduino amb un Real time + SD shield	28
Figura 4.11 – Diagrama modular de l'ArduinoRT	30
Figura 4.12 - Càlcul teòric de l'autonomia del ArduinoRT.....	31
Figura 4.13 - Càlcul teòric de l'autonomia del ArduinoDLv2.....	32
Figura 4.14 - Funcionalitat de captura.....	34
Figura 4.15 - Disposició modular del rPi-eth	35
Figura 4.16 - Càlcul teòric de l'autonomia del rPi-eth.....	35
Figura 4.17 - Càlcul teòric de l'autonomia del rPi-eth.....	39
Figura 4.18 - Càlcul teòric de l'autonomia del rPi-eth amb dues bateries.....	39
Figura 4.19 - Disposició modular del rPi-GPRS	40
Figura 4.20 - Funcionalitat d'emmagatzematge.....	41
Figura 4.21 - Funcionalitat de consulta	43
Figura 5.1 - Mapa de distribució de dispositius de control	46
Figura 6.1 - Pista cremada d'un ArduinoRT	48
Figura 7.1- Preus dels components dels dispositius de control	50
Figura 7.2 - Cost total de la configuració emprada a la prova de camp.....	51
Figura 7.3 - Preus de serveis de cronometratge.....	51
Figura 8.1 - Estadístiques del servei Pusher del dia de la prova de camp.....	56

1. Introducció

1.1. Justificació

Des de temps immemorials la humanitat ha inventat i practicat esports. Aquests es poden separar en dues categories ben diferenciades:

- Esports basats en objectius: fer entrar una pilota a una porteria (futbol), encistellar una pilota (bàsquet), disparar amb precisió sobre un objectiu (tir olímpic)...
- Esports basats en el temps emprat per complir un objectiu: anar del punt A al punt B amb el menor temps possible, cobrir una distància concreta el més ràpid que es pugui... En aquesta categoria s'hi troben esports com l'esquí, el ciclisme, l'automobilisme...

Als esports basats en la velocitat de complir un objectiu la mesura del temps esdevé un factor crític. S'ha de fer d'una manera justa per tots els participants i amb unes regles que normalment venen determinades per la normativa de l'esport o competició concreta.

La dificultat tècnica del cronometratge en les diferents modalitats serà creixent. En l'atletisme en pista els participants donen voltes a un circuit de 400 metres. Es tracta d'un lloc relativament petit en el que presumiblement la dificultat d'aconseguir, per exemple, electricitat o accés a Internet serà relativament baixa. Aquesta dificultat tècnica creixerà en l'atletisme en circuits urbans sobretot per l'ampliació de l'àmbit geogràfic. En curses de muntanya que la competició es desenvolupa en entorns remots la dificultat tècnica esdevé encara més alta.

Aquest projecte pretén dotar a una cursa de muntanya amb una solució de cronometratge. La cursa triada és Pels Camins dels Matxos [1]. Es tracta d'una cursa de muntanya amb sortida i arribada al poble de Torelló [2]. És una cursa circular en la que els participants, segons la seva preferència, han de cobrir una distància de 49Km, 53Km, 56Km o 63Km.

L'elecció d'aquesta cursa rau en el fet de que l'alumne fa molts anys que està involucrat amb l'organització de l'esdeveniment. Conèixer bé des de dins com s'organitza i es desenvolupa la cursa és un gran avantatge a l'hora de plantejar aquest repte.

Fins ara el cronometratge de la cursa es feia mitjançant codis de barres impresos als dorsals dels participants. Quan els participants passaven pels diferents punts de control una persona de l'organització feia una lectura del codi de barres. Totes les lectures quedaven emmagatzemades al dispositiu. Quan el punt de control es tancava, la persona responsable portava el dispositiu al centre de control de cursa i allà es processaven. Aquest sistema plantejava algunes limitacions a l'organització:

- Els dispositius lectors de codis de barres eren propietat de la Federació d'Entitats Excursionistes de Catalunya (FEEC) [3]. Al ser de propietat d'un tercer l'organització no podia desenvolupar aquesta solució per dotar-la de més funcionalitats.
- El problema d'aquesta solució és el temps d'espera entre que passa el primer participant pel control i quan les dades arriben al centre de control. Això fa que, per exemple, l'organització no pugui saber mai quanta gent hi ha corrent a cada tram del recorregut.

Amb aquesta situació l'alumne, d'acord amb l'organització, va decidir dissenyar una solució pròpia que permetés gestionar el cronometratge de l'esdeveniment. Aquesta solució havia de poder solucionar els dos problemes del sistema anterior: ser propietat de l'organització i dotar a alguns punts de control amb tramesa de dades a temps real.

S'ha de tenir en compte però, que la nova solució no pot obviar les bones funcionalitats del sistema anterior: la funcionalitat de lectura i emmagatzematge de les dades, que els dispositius siguin reutilitzables any rere any i, finalment, que els dispositius siguin portàtils i autònoms.

1.2. Descripció del projecte

Aquest projecte pretén donar una solució integral de cronometratge per una cursa de muntanya. La cursa de muntanya seleccionada, Pels Camins dels Matxos, té unes característiques orogràfiques i organitzatives molt concretes però aquesta solució seria aplicable a un ampli ventall de curses de muntanya.

En una cursa de muntanya els participants han de seguir el recorregut marcat per l'organització. Per assegurar que els participants fan aquest recorregut i no un altre, l'organització estableix estratègicament una sèrie de punts de control. Segons la normativa de la cursa cada participant ha d'haver passat per tots els punts de control perquè l'organització pugui considerar el seu recorregut com a vàlid.

El recorregut de la cursa es desenvolupa al nord d'Osona. Els participants de seguida abandonen la zona urbana i es dirigeixen cap a les serres de Bellmunt, Curull i Cabrera. Aquest és un entorn muntanyós i de difícil accés on alguns dels punts de control només són accessibles a peu. Així doncs, la solució de cronometratge haurà de constar de dispositius d'adquisició de dades autònoms i mòbils.

Tal i com marca la llei, l'organització de Pels Camins dels Matxos disposa d'un pla de seguretat aprovat per tots els Ajuntaments dels termes municipals per on es desenvolupa la cursa. En aquest document s'identifiquen tots els riscos i es fa un pla per reduir-los al màxim. Els apartats que tenen a veure amb la seguretat dels participants tenen una incidència en la solució de cronometratge. L'exemple més clar és el fet de que no es pot tancar un control si no s'està del tot segur de que hi han passat el mateix nombre de corredors que al control anterior.

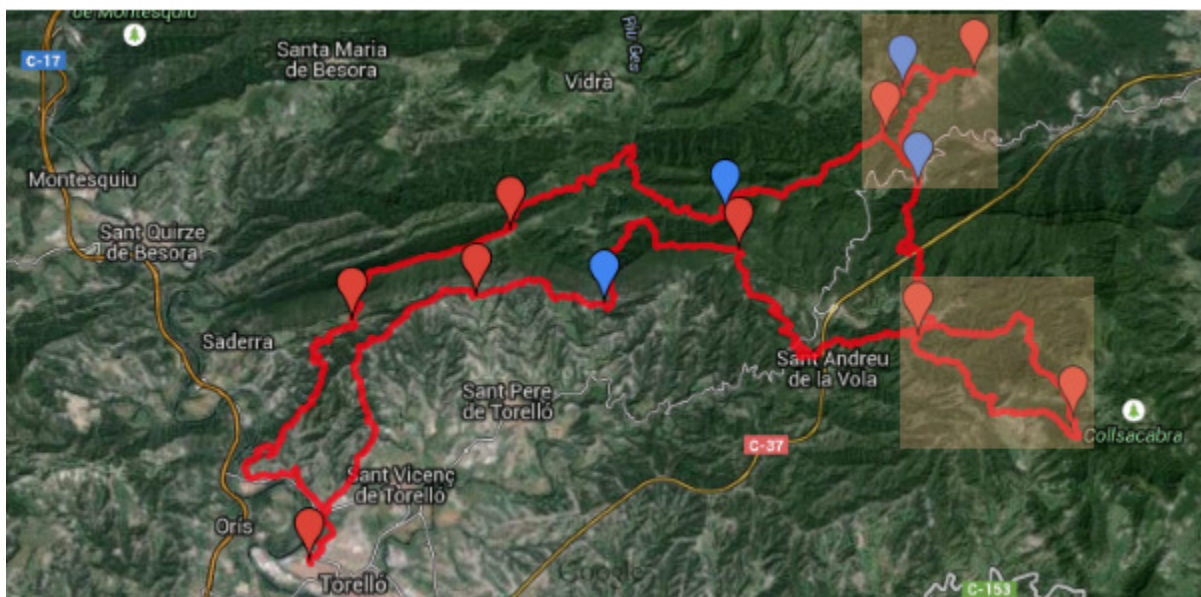


Figura 1.1 - Localització geogràfica del recorregut de la cursa

Una altra peculiaritat de la cursa és el fet de tenir quatre recorreguts disponibles i que els participants poden decidir quin faran mentre van vançant. Això s'aconsegueix amb un disseny de recorregut circular que inclou dos sub-recorreguts: un al Puigsacalm i un a Cabrera. La Figura 1.1 representa la localització geogràfica del recorregut i destaca els dos sub-recorreguts.

Quan els participants aconseguen arribar al final de la cursa l'organització els imprimeix un diploma. En aquell moment l'organització ha de saber quin recorregut ha realitzat cada corredor. Per tant, alguns dels controls hauran de ser capaços d'enviar dades en temps real a la base de dades de la cursa. Val a dir que fins ara l'organització ha fet servir un sistema manual per aconseguir aquesta fita. Es tracta de dues caselles al dorsal que estan etiquetades amb els noms de Puigsacalm i Cabrera. Quan els participants arriben a aquests punts els responsables del control fan una marca al dorsal del participant.

La Figura 1.2 mostra l'estructura de la cursa amb els seus diferents recorreguts i punts de control.

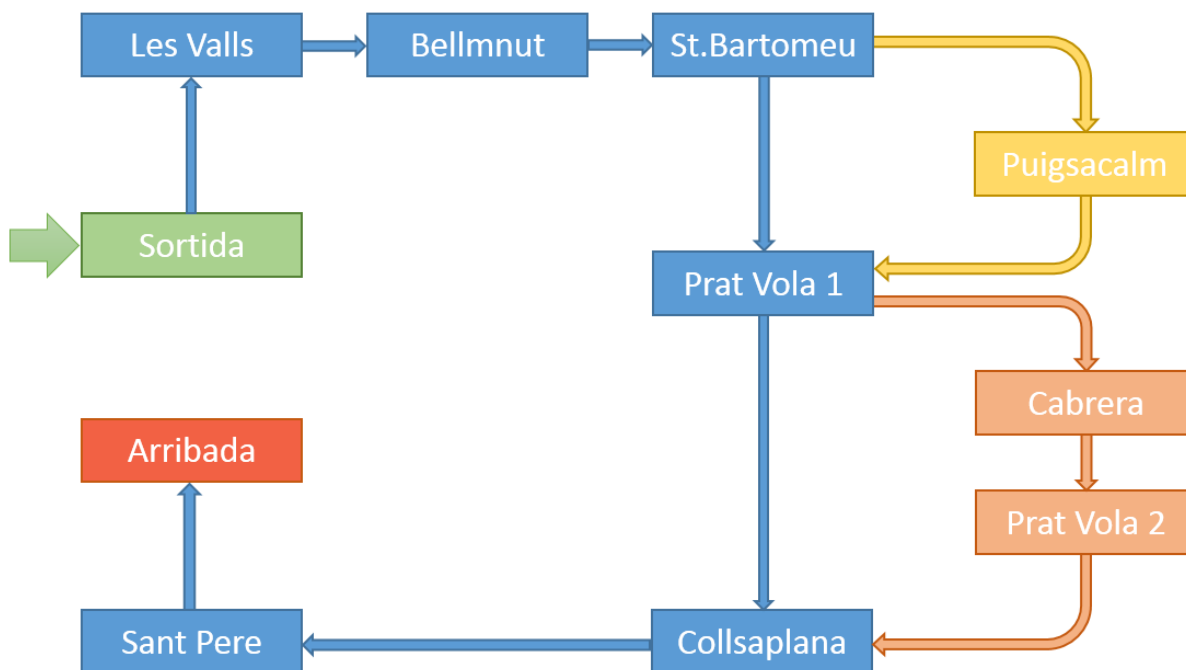


Figura 1.2 - Esquema de recorregut i punts de control

En resum, els principals requisits funcionals del sistema de cronometratge són els següents:

- Identificació inequívoca dels participants
- Dispositius de control amb les següents característiques:
 - Autònoms i portàtils per la captura de dades de pas dels participants en llocs remots
 - Propietat de l'organització i reutilitzables a cada edició de la cursa
 - Visualització del recompte de participants
 - Sistema de persistència de dades
 - Alguns dispositius hauran de ser capaços d'enviar dades a través d'Internet
- El sistema ha de ser capaç de determinar quin recorregut ha efectuat cada participant
- Les dades que envien els dispositius cap a Internet han de ser presentades en una pàgina web per tal de que la gent pugui seguir les evolucions dels participants a la cursa

1.3. Objectius del TFC

L'objectiu principal és construir una solució integral per a gestionar el cronometratge de la cursa Pels Camins dels Matxos que es celebra el dia 19 d'Abril de 2015.

A cada un dels 11 punts de control que l'organització ha establert s'haurà de poder capturar l'hora que hi ha arribat cada participant. Aquestes dades hauran de ser emmagatzemades en algun tipus de suport que sigui fàcil de processar a posteriori. L'emmagatzematge de totes les captures de dades és la part més crítica de la solució perquè ha de permetre a l'organització saber a quina hora ha passat cada participant per cada punt de control.

Partint dels requeriments funcionals definits en l'apartat anterior es defineixen una sèrie d'objectius per al present TFC que es passen a detallar a continuació.

Identificació inequívoca dels participants.

Dispositius de control autònoms i portàtils. Restriccions importants de disseny: han de tenir una autonomia de 15 hores i han de tenir un volum i pes que permetin ser transportats per una persona en una motxilla.

Dispositius propietat de l'organització i reutilitzables.

Visualització del recompte de participants. Objectiu derivat del pla de seguretat de la cursa. El sistema de cronometratge ha d'ajudar l'organització a decidir si es pot tancar un punt de control. Això implica que els punts de control han de mostrar el recompte de participants.

Sistema de persistència de dades. Tots els dispositius de control hauran de tenir la possibilitat de guardar de forma persistent les dades que van adquirint.

Accés a Internet per a alguns dels dispositius. La transmissió en temps real de les dades que van capturant els dispositius té infinitat d'avantatges per a la gestió de la cursa i per al públic que la segueix. És un objectiu secundari degut a la complexa orografia del terreny i la baixa quantitat de senyal que proporcionen els diferents operadors.

El sistema ha de ser capaç de determinar quin recorregut ha efectuat cada participant.

Les dades en temps real han de ser visibles des d'una plana web. No n'hi ha prou en presentar-les en una pàgina web clàssica que requereixi a l'usuari actualitzar-la per veure les noves dades sinó que la pàgina ha de tenir un mecanisme per presentar les dades noves tal com arriben.

1.4. Marc del projecte

Aquest projecte va arrencar a principis de l'any 2014 quan l'alumne va decidir junt amb l'organització de la cursa fer una prova pilot d'un sistema de cronometratge dissenyat per l'alumne. La idea inicial era dissenyar alguns prototips i fer-los servir com a sistema de cronometratge paral·lel a la cursa Pels Camins dels Matxos del dia 13 d'abril de 2014. Els prototips havien de servir com a estudi de viabilitat per a un futur sistema de gestió integral del cronometratge de la cursa.

Aquest estudi previ fel l'any 2014, tot i no haver donat uns resultats perfectes, va fer veure tant a l'organització com a l'alumne que era possible gestionar la cursa amb aquest tipus de solució. Es va acordar assumir la responsabilitat plena del cronometratge de l'edició del dia 19 d'abril de 2015.

Els primers prototips es basaven amb unitats de control Arduino [4]. La selecció d'aquesta tecnologia va venir donada pels següents factors: baix cost de maquinari, comunitat de desenvolupadors molt gran i activa,

moltes possibilitats de comunicació amb altres dispositius mitjançant interfícies de diferents tecnologies, llenguatge de programació C que és perfectament familiar per l'alumne.

Es van dissenyar dos models diferents de prototips amb tecnologia Arduino:

- **ArduinoDL1.** Aquest model implementa les funcionalitats bàsiques d'un *datalogger* [5]: rellotge intern, lectura i emmagatzematge de dades en una targeta SD. El resultat d'aquest model va ser més que satisfactori fet pel qual s'ha tornat a fer servir a l'edició de 2015 amb algunes petites modificacions.
- **ArduinoRT.** Aquest model implementa les mateixes funcionalitats que l'ArduinoDL1 més la capacitat de connexió a Internet. En la fase d'implementació es va detectar una de les principals limitacions de l'Arduino, la quantitat de memòria disponible que és al voltant dels 28KB. El resultat no va ser satisfactori ja que la limitació de memòria feia que no es pogués fer un control d'errors rigorós.

De la prova pilot a l'edició 2014 se'n van extreure moltes experiències. En primer lloc, es va arribar a la conclusió de que la plataforma Arduino era vàlida per la tasca de *datalogger* però no per afegir-hi capacitats de xarxa. En segon lloc, es van fer reunions amb l'organització per definir de manera formal els requeriments funcionals de la solució.

Quan els requeriments funcionals van estar formalment definits es va procedir a seleccionar la nova plataforma de treball que va ser Raspberry Pi [6]. Aquesta selecció es va basar en els mateixos criteris de selecció que la plataforma Arduino. Essencialment, la Raspberry Pi és un concepte similar però amb molta més capacitat de procés, molta més memòria i menys capacitats d'expansió, tot i que suficients per les necessitats del projecte.

1.5. Enfocament i mètode seguit

El disseny i fabricació dels primers prototips es comença seguint un mètode de prova i error. En el moment de començar el projecte els requeriments funcionals no estaven formalment definits i no es podia seguir una metodologia formal de treball.

Amb els primers prototips validats i els requeriments funcionals formalment definits es passa a utilitzar una metodologia de treball *Agile* [7]. Amb aquest mètode, la prioritat és tenir sempre un producte operatiu i anar-hi afegint noves funcionalitats a cada iteració.

Poques setmanes abans de la cursa es va procedir a fer proves de camp. Les proves tenien l'objectiu de provar a fons la funcionalitat d'enviament de dades a Internet a temps real. Amb els resultats d'aquestes proves es van seleccionar els operadors que oferien millor servei a cada un dels tres punts necessaris i es va donar la configuració per acabada.

1.6. Planificació del projecte

El treball de final de carrera objecte de la present documentació va començar a principis de març de 2015. La primera presa de contacte va consistir en una sèrie de correspondència entre el tutor del projecte i l'alumne en la que l'alumne suggeria una proposta de projecte alternatiu. Fruit d'aquestes converses es va dissenyar la primera planificació que es mostra a la Figura 1.3.

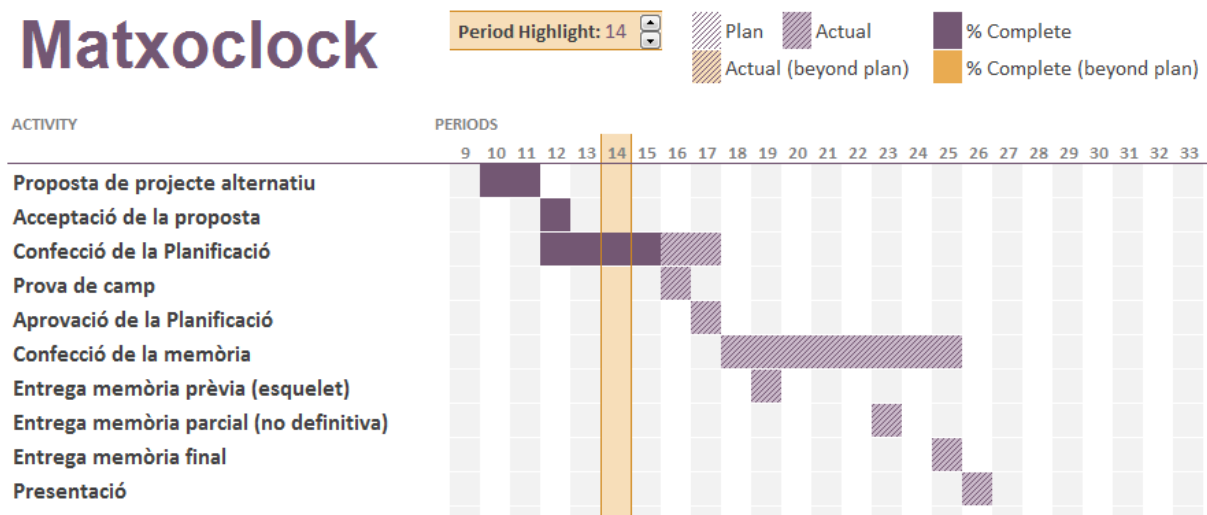


Figura 1.3 - Primera planificació presentada

La planificació inicial es va dissenyar tenint en compte les dates de la UOC juntament amb les dates de la prova de camp que era un *deadline* important i inamovible.

A la Figura 1.4 següent es pot veure com va quedar la planificació final. S'ha de tenir en compte que l'alumne va estar de baixa mèdica durant 8 dies entre les setmanes 17 i 18. Aquest fet va fer endarrerir l'aprovació de la planificació i l'inici de la confecció de la memòria.

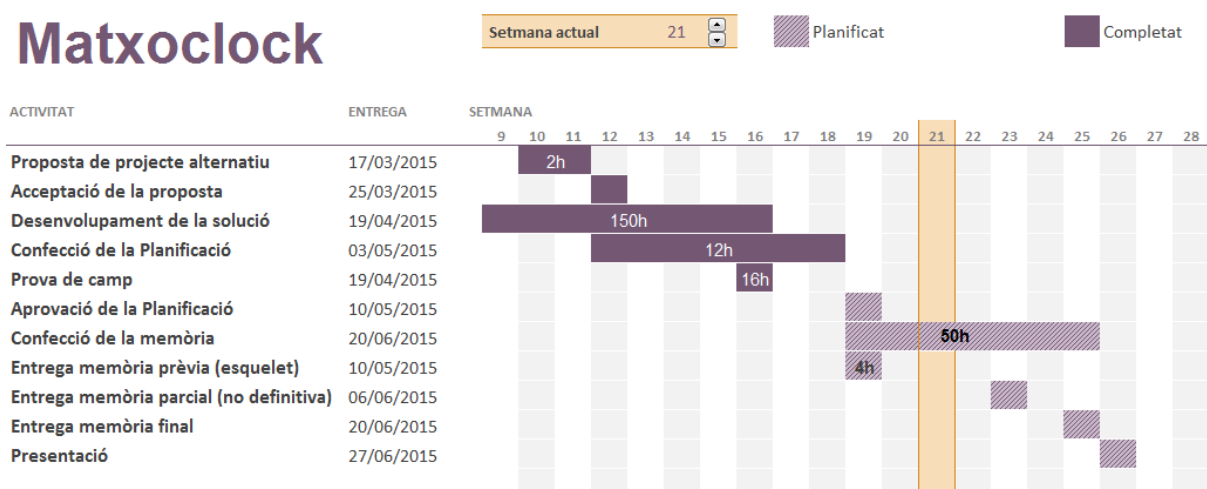


Figura 1.4 - Planificació final

Pel que fa a la precisió entre les temporitzacions teòriques i reals no hi ha hagut massa diferència. Una excepció a aquesta afirmació podria ser la redacció de la present documentació que, tot i no haver-se mesurat de forma precisa, dóna la sensació d'haver resultat ser una planificació massa optimista.

1.7. Recursos emprats

Les eines de les que ha disposat l'alumne es poden classificar segons cada vessant de la solució en les següents categories:

- **Maquinari.** Tenint en compte que el projecte té una component d'electrònica s'han emprat eines habituals d'aquesta ciència: cablejat de diferents diàmetres i longituds, terminals, interruptors i components electrònics, bateries, caixes estanques, jocs de tornavisos, jocs de tenalles, multímetre, soldador d'estany, font d'alimentació variable i diverses fonts d'alimentació a tensions fixes.
- **Programari.** L'ordinador amb sistema operatiu Windows 7 ha estat l'eina essencial per totes les tasques i per a aquesta en particular. La gestió documental i les còpies de seguretat s'han fet sobre Dropbox. La programació dels Arduino s'ha fet amb C utilitzant l'entorn de desenvolupament (*IDE*) oficial: Arduino IDE 1.6.4. La programació de les Raspberry Pi s'ha fet amb Python sense l'ajuda de cap *IDE*; s'ha hagut de fer alguna petita programació amb Shell Script per automatitzar la posada en marxa. Per a la programació de l'entorn web tampoc s'ha utilitzat cap *IDE*; els llenguatges de programació emprats són HTML, Javascript, PHP, Java i SQL. Per fer la presentació de dades a temps real s'ha utilitzat el servei comercial Pusher [8]. Per editar el codi font s'ha utilitzat l'editor Notepad++ i el seu *plugin* FTP per distribuir-lo. La documentació del projecte s'ha fet amb Office 2013.
- **Comunicacions.** Alguns models de dispositius de control porten un *router* del fabricant Mikrotik [9]. La configuració d'aquests dispositius s'ha fet amb RouterOS que és el sistema operatiu natiu del fabricant.

1.8. Productes obtinguts

El present projecte ha acabat generant una important quantitat de dispositius de maquinari i d'artefactes de programari. El sistema central que aglutina totes les dades és un servidor amb servei web sobre Apache i un sistema gestor de base de dades sobre MySQL. Aquests serveis ja estaven operatius abans del començament del projecte. L'alumne només ha hagut de fer petits canvis a la base de dades i algunes tasques d'optimització de recursos.

Al servidor hi resideixen els artefactes de programari que s'han creat per guardar les captures de dades i per consumir-les. Paral·lelament s'ha creat una petita aplicació en Java per fer càrregues massives de captures de dades partint d'un fitxer de text.

Els dispositius de control són el gran gruix dels productes obtinguts des del punt de vista de maquinari. S'ha de tenir en compte que s'han arribat a produir fins a 13 dispositius de control essent tots ells operatius. Aquests dispositius es poden dividir en 5 models diferents que s'analitzaran detalladament al Capítol 4: ArduinoDLv1, ArduinoRT, ArduinoDLv2, rPi-eth, rPi-GPRS.

1.9. Breu descripció dels propers capítols

2- Antecedents. Estat de l'art, empreses que ofereixen serveis similars i estudi de mercat.

3- Descripció funcional. Descripció funcional del sistema com a solució amb les seves diferents etapes: captura, emmagatzematge i consulta.

4- Descripció de la implementació. Decisions de disseny, arquitectura de programari i descripció de la implementació de cada funcionalitat.

5- Configuració de la prova de camp. Disposició dels dispositius de control a la prova de camp.

6- Viabilitat tècnica. Estudi de viabilitat tècnica de cada dispositiu de control.

7- Valoració econòmica. Escandall de costos del projecte.

8- Conclusions. Grau d'assoliment d'objectius, propostes de millora i autoavaluació.

2. Antecedents

2.1. Estat de l'art

Hom pot fer mesures aproximades de distància (pams, passes...) però es fa molt difícil fer mesures de temps més petites que un dia. Al llarg de la història la humanitat ha anat desenvolupant tecnologies per mesurar el temps com ara el rellotge de sol o rellotges mecànics. Amb l'arribada de les diferents tecnologies per mesurar el temps es va poder plantejar solucionar problemes complexos de cronometratge.

Avui en dia, la problemàtica del cronometratge està íntimament lligada a les ciències electrònica i informàtica. Aquestes ciències proveeixen de gran quantitat d'eines per mesurar el temps que es poden aplicar a àmbits molt diversos. Un dels àmbits, entre d'altres, als que el cronometratge és un factor crític és als esdeveniments esportius.

Depenent de l'esport i del nivell de professionalitat, les exigències per al sistema de cronometratge seran totalment diferents. Un exemple de sistema de cronometratge molt exigent podria ser el de les curses automobilístiques de Fórmula 1. En aquest esport professional hi ha requeriments molt exigents com ara mesures de temps amb una precisió de mil·lèsima de segon o sistemes redundants d'adquisició de dades. A l'altre extrem s'hi podria col·locar una cursa atlètica amateur en la que una persona engega un cronòmetre de ma quan comença la cursa. Aquesta mateixa persona podria anar anotant els temps dels participants a mesura que passin per la línia d'arribada.

Seguidament, es passen a enumerar les últimes novetats funcionals i tecnològiques en el mercat dels serveis de cronometratge de curses de muntanya.

- **Comunicacions.** Amb l'ús de la tecnologia actual, la societat està molt interconnectada i té l'hàbit de rebre informació de forma immediata. Aquest concepte d'immediatesa es tradueix en enviar les captures de dades de pas immediatament a la xarxa perquè estiguin disponibles als consumidors amb el mínim retard possible. En emplaçaments de dispositius d'adquisició remots s'utilitzen tecnologies de telefonia mòbil (GPRS, UMTS, LTE...) o comunicacions via satèl·lit.
- **Xips bidireccionals.** Els desplegaments de tecnologia punta no estan a l'abast dels pressuposts de totes les curses de muntanya. Per tal d'abaratir costos, una de les tendències actuals és emprar xips bidireccionals. Aquests tipus de dispositius a part de transmetre el seu identificador recullen el temps i l'emmagatzemen. A l'arribada s'hi emplaça un dispositiu lector que recull les dades emmagatzemades al xip i les fa arribar al contenidor de dades corresponent. Aquesta tecnologia va en contra de la tendència de immediatesa però aconsegueix baixar els costos a base de no incloure sistemes de comunicacions a cada control intermedi.

- **Dispositius de llarga autonomia.** L'atletisme ha viscut en els últims anys una tendència que consisteix en allargar les curses. Es dona el cas de curses en que els l'organització dona dies per finalitzar la cursa [10]. Això implica que la tendència actual als dispositius de captura de dades és la minimització del consum, l'adopció de les últimes tecnologies aplicades a bateries i sistemes híbrids amb energies renovables.
- **API's d'integració.** Les curses de més repercussió i pressupost tenen els seus propis recursos tecnològics per gestionar la cursa. Aquests tipus d'organitzacions solen treballar amb un proveïdor tecnològic a qui encomanen la gestió del cronometratge. Aquest sistema de cronometratge haurà de ser capaç d'intercanviar dades amb els sistemes de l'organització de la cursa.
- **Previsió de temps de pas.** El següent pas a la tendència de la immediatesa és saber les coses abans de que passin. A partir de l'anàlisi dels resultats d'altres anys i mitjançant eines de correlació estadística, hi ha proveïdors que ofereixen estimacions de temps de pas dels participants. Així, quan un corredor ha passat pel primer control de temps intermedi el sistema estima els temps de pas dels següents controls fins l'arribada.

2.2. Referències d'empreses de serveis de cronometratge

2.2.1. Novotiming

Novotiming és un proveïdor innovador que ofereix un sistema de cronometratge (Smartchrono) basat en codis QR impresos als dorsals. Els operadors dels controls utilitzen una aplicació de mòbil per fer les lectures. Aquest sistema representa un cost quasi nul en adquisició de maquinari perquè s'utilitzen els telèfons intel·ligents dels operadors.

- **Punts forts.**
 - Al no haver-hi maquinari el preu final és molt competitiu.
 - Diferenciació respecte altres competidors del mercat de cronometratge.
 - Si els punts de control tenen bona cobertura tots els punts de control són a temps real.
- **Punts febles.**
 - L'autonomia dels dispositius és una responsabilitat dels operadors.
 - El temps necessari per fer les captures de temps és relativament alt i això pot ser problemàtic en punts de control amb gran afluència concurrent de participants.
 - Inscripcions i resultats en lloc web extern al de la cursa.
 - Baixa precisió per ser un sistema que requereix parar al corredor.

2.2.2. Temps x curses

Aquest proveïdor és un dels que ofereix el servei de cronometratge que les curses majoritàriament utilitzen. Es tracta d'un sistema de cronometratge que utilitza xips RFID. Tot i oferir altres possibilitats, la configuració

típica consisteix en una mesura manual del començament de la cursa i un sol lector de xips RFID en forma de catifa a l'arribada.

- **Punts forts.**
 - Desplegament de la infraestructura molt senzill perquè només requereix d'un element de maquinari a l'arribada.
 - Sistema d'alta precisió.
 - Cost del servei competitiu i assumible per la majoria de curses.
- **Punts febles.**
 - Impossibilitat de controlar autònomament diversos recorreguts.
 - Els resultats es presenten una vegada s'ha acabat la cursa.

2.2.3. Livetrail

Livetrail és un dels proveïdors més internacionals i amb més renom del mercat. Ofereix múltiples solucions de cronometratge que van des d'una configuració minimalista com la que ofereix Temps x curses fins a seguiment en temps real amb previsió de temps de pas. A l'extrem superior dels seus serveis s'hi situa un servei de coorganització que delega les funcions d'inscripció, cronometratge i seguiment en temps real al proveïdor. En aquest cas, els seus serveis són integrats al lloc web de la organització per mantenir la imatge corporativa.

- **Punts forts.**
 - Tècnicament aquesta empresa pot solucionar qualsevol requeriment de cronometratge de curses de muntanya.
- **Punts febles.**
 - Preus molt alts només a l'abast de curses professionals i/o de gran ressò mediàtic.

2.3. Estudi de mercat

El més habitual és que la organització dels esdeveniments contracti el cronometratge a una empresa especialitzada. Al mercat hi ha multitud d'empreses que ofereixen aquests serveis. Aquests serveis de cronometratge es poden classificar en dues categories des del punt del vista de l'operador:

- **Actius.** El participant porta algun tipus d'identificador que l'operador de cronometratge ha de llegir de forma activa. Exemples d'aquesta categoria: lectura de codis de barra o codis QR impresos als dorsals.
- **Passius.** L'organització proporciona un dispositiu identificador al participant. El punt de control disposa d'algun tipus de lector que captura els dades de pas dels participants. L'operador del control

només ha de fer que els participants passin per la zona designada: caminin sobre una manta, passin per un arc o acostin el seu dispositiu identificador a un dispositiu lector determinat.

La immensa majoria de proveïdors que ofereixen aquests serveis opten per un sistema passiu. Aquests solen consistir de xips amb tecnologia RFID que els participants han de portar a la bamba, integrats al dorsal o al canell. Als punts de control s'hi despleguen antenes que poden prendre diferents formes: al nivell del terra (normalment protegides amb una manta), un petit arc pel que només pot passar una persona a l'hora o una caixa a la que el participant ha d'acostar el seu xip RFID.

Tant les solucions actives com passives poden complir els requeriments de la majoria de curses de muntanya. Una gran diferència entre elles és el preu. Els serveis actius són més barats que els passius. Això es deu a que els dispositius són més barats però sobretot al fet de que requereixen molta menys implicació per part del proveïdor; això s'acaba traduint en temps i diners. En resum, els costos del servei de cronometratge vindran donats pel preu del lloguer dels equips més el temps que el proveïdor hagi de treballar per l'organització de la cursa.

El ventall de possibilitats que ofereixen les empreses de cronometratge és molt ampli. Es pot optar per una solució molt senzilla en la que l'empresa només ofereix dispositius lectors. En aquest cas, és responsabilitat de l'organització de la cursa preparar els dispositius identificadors, associar-los a cada participant, portar el control de temps dels participants, generar classificacions... A l'altre extrem hi ha empreses que ofereixen serveis de gestió integral de la cursa. Això no només inclou la solució de cronometratge sinó que també inclou: gestió d'inscripcions, serveis de seguiment de participants en temps real, ma d'obra per operar els punts de control, confecció i presentació de resultats...

Pel que fa a les configuracions del sistema de cronometratge que ofereixen els proveïdors també és molt ampli. La configuració típica, amb uns costos molt competitius, consisteix amb una assignació de temps de sortida manual i un sol dispositiu de captura a l'arribada. Aquesta configuració no es apta per a curses de llarga distància on poden haver-hi múltiples recorreguts i la necessitat de controlar temps entremitjos. Els proveïdors tenen solucions amb múltiples punts de captura de dades i transmissió a temps real de resultats però a preus prohibitius per la majoria d'organitzacions.

Amb aquestes premisses, es divisa un nínxol de mercat per a serveis de cronometratge de curses de muntanya de llarga distància amb preus per sota els de mercat. Aquest nínxol de mercat és al que va dirigit aquest projecte.

2.3.1. Unitats de control

Fer l'adquisició de dades amb dispositius autònoms implica que hauran de tenir unes funcionalitats pre-programades i una sèrie de perifèrics. Així doncs, els dispositius de control hauran d'estar governats per una

unitat de control. Una opció seria partir d'un microcontrolador i programar a baix nivell totes les funcionalitats i interfícies amb els perifèrics. Aquesta opció implica un temps de desenvolupament i prototipatge molt alt. Com a segona opció, el mercat proporciona d'una sèrie de plataformes especialment orientades al prototipatge. Aquesta segona opció s'ha considerat molt més vàlida en termes de temps de desenvolupament i cost.

El mercat ofereix infinitat d'alternatives però les que es s'han considerat són les següents: Arduino Uno R3 [11], Raspberry Pi B+ [12] i Beaglebone Black [13]. Hi ha desenes de paràmetres tècnics per comparar però s'ha considerat que els que es mostren a la Figura 2.1 són els més representatius per a aquest projecte.

Definició dels diferents paràmetres tècnics:

- **Preu (€).** S'ha fet una mitjana de preus perquè hi ha molts proveïdors per cada dispositiu. No s'hi ha inclòs altres despeses com ara el transport. S'ha de tenir en compte que els preus varien d'una forma considerable amb espais de mesos. La raó és que aquest mercat està en constant evolució i quan es presenten models nous els models anteriors tendeixen a baixar de preu.
- **Consum (mA).** En aquest apartat s'ha intentar determinar un rang de consums. Aquests tipus de dispositius tenen sistemes d'estalvi d'energia que permeten baixar el consum elèctric depenent de la càrrega computacional. Els valors aquí presentats són els que es consideren nominals. L'Arduino no disposa de cap sistema d'estalvi d'energia més enllà d'un mode de repòs. En aquest mode de repòs l'Arduino consumeix uns 10mA i es pot controlar mitjançant interrupcions.
- **Expansió (1-5).** Degut a la complexitat de les possibilitats d'expansió, aquest punt podria ben merèixer una comparativa aïllada. El ventall de punts de vista des dels que analitzar aquesta característica és molt ampli: nombre d'entrades/sortides analògiques i/o digitals, temporitzadors, ports de modulació per amplada de pols (*PWM*), ports de comunicació sèrie, possibilitat d'ampliació mitjançant mòduls pre-dissenyats... L'alumne s'ha pres la llicència de puntuar entre 1 i 5 aquestes capacitats d'acord amb les necessitats del projecte.
- **Memòria (MB).** En el cas de l'Arduino s'ha computat la suma dels 32 KB de memòria Flash, els 2KB de memòria SRAM i 1KB de memòria EEPROM. En els altres dos casos simplement s'ha computat la memòria RAM que porten instal·lada.
- **Comunitat (1-5).** Altra vegada, l'estudiant s'ha pres la llibertat de puntuar el volum, la quantitat i la qualitat de la informació disponible a les respectives comunitats de desenvolupadors.

	Preu (€)	Consum (mA)	Expansió (1-5)	Memòria (MB)	Comunitat (1-5)
Arduino Uno R3	23 €	80-100	5	0,035	5
Raspberry Pi B+	26 €	500-1000	2	512,000	5
Beaglebone Black	54 €	800-1300	5	512,000	3

Figura 2.1 - Comparativa de característiques tècniques d'unitats de control

A la Figura 2.1 s'hi han tabulat els valors per cada dispositiu. Aquestes són les conclusions de la comparativa de característiques tècniques:

- **Preu (€).** L'Arduino i la Raspberri Pi estan en un rang de preu molt similar al voltant dels 25€. El Beaglebone té un preu que duplica el dels altres dos candidats.
- **Consum (mA).** En aquest apartat el clar guanyador és l'Arduino que està ordres de magnitud per sota els seus contrincants.
- **Expansió (1-5).** L'Arduino i el Beaglebone han rebut la màxima puntuació per raons diferents. En el cas de l'Arduino l'alumne ha considerat molt positiu el concepte dels *shields* i la gran quantitat que n'hi ha al mercat. Pel que fa al Beaglebone, el nombre de ports de diferents tecnologies que porta de fàbrica és clarament superior al dels seus rivals.
- **Memòria (MB).** Mentre que la Raspberry Pi i el Beaglebone tenen un valor idèntic, l'Arduino té un valor que és molts ordres de magnitud inferior.
- **Comunitat (1-5).** Segons l'opinió de l'estudiant, la informació que es pot trobar a les comunitats de desenvolupadors d'Arduino i Raspberry Pi són clarament superiors a la de Beaglebone. S'ha de tenir en compte que, per principis, les plataformes Arduino i Raspberry Pi estan orientades a la formació de nous desenvolupadors. Aquest fet dóna a les comunitats d'Arduino i Raspberry Pi un caire molt més pedagògic que en el cas de Beaglebone.

Analitzant els resultats, el Beaglebone és el primer descartat. Els factors més determinants són l'elevat consum i la menor orientació pedagògica de la seva comunitat de desenvolupadors. Pel que fa als dos finalistes, la Raspberry Pi té un consum molt més elevat i unes capacitats menors d'expansió. A l'altre cantó de la balança hi ha l'Arduino que té millors (o iguals) puntuacions a totes les característiques amb excepció de la quantitat de memòria que és extremadament menor.

En conclusió, l'Arduino Uno R3 és la millor opció si les prioritats són el baix consum i la capacitat d'expansió. Tot això, tenint en compte que podrà ser una opció real si els requeriments funcionals són reduïts. Aquest fet bé donat per la petita quantitat de memòria que porta, fet que farà que no s'hi pugui emmagatzemar codi font gaire llarg ni acumular gran quantitat de dades en memòria.

Si els requeriments computacionals i de memòria són superiors, s'hauria de seleccionar la Raspberry Pi B+ o el Beaglebone Black segons els requeriments de consum i expansió; sempre tenint en compte les diferències en les seves respectives comunitats d'usuaris.

3. Descripció funcional

3.1. Introducció

En aquest capítol es fa una definició formal de les funcionalitats que ha de tenir el sistema de cronometratge. Al mateix temps de ser una definició precisa i que no porti a dobles interpretacions, ha de ser oberta en el sentit que no forci a una implementació concreta. Amb tota certesa hi ha diferents formes vàlides d'implementar aquesta descripció funcional.

3.2. Matxoclock: sistema de cronometratge per a una cursa de muntanya

Aquest projecte dóna una solució al problema de la gestió de cronometratge de curses de muntanya. Analitzant la problemàtica, es pot segmentar en tres grans blocs funcionals representats a la figura següent.



Figura 3.1 – Funcionalitat bàsica del sistema

Primerament, el sistema ha de tenir un mecanisme per capturar les dades. Aquestes dades que s'han capturat s'hauran de poder emmagatzemar en un contenidor. Les dades s'han d'emmagatzemar amb una estructura i una forma que permeti fer-hi consultes.

Analitzant els requeriments funcionals es detecta que hi ha dos processos temporals diferenciats per emmagatzemar les dades al contenidor comú: transmissió de dades a temps real i a temps diferit. Això implica que hi haurà dues vies diferenciades d'enviament de dades com es pot veure a la figura següent.



Figura 3.2 - Diferenciació temporal a l'hora d'emmagatzemar les dades

Del fet de que en alguns casos les dades s'han de poder emmagatzemar en temps real se'n deriva una nova funcionalitat: la consulta a temps real. Prenent la consideració que la funció de consulta és un consumidor de dades implica que la funcionalitat d'emmagatzematge haurà de ser capaç d'enviar dades en temps en temps real a la funcionalitat de consulta.

A diferència de la funcionalitat d'emmagatzematge que és comú per a tota la solució, les funcionalitats de captura i consulta han de poder suportar múltiples instàncies. La funcionalitat de captura ha de ser capaç de capturar i transmetre dades al mateix temps i des de localitzacions geogràfiques diferents. Al mateix temps,

la funcionalitat de consulta ha de ser multi-usuari; totes les instàncies de la funcionalitat de consulta han de poder rebre les dades tal i com arriben a la funcionalitat d'emmagatzematge.

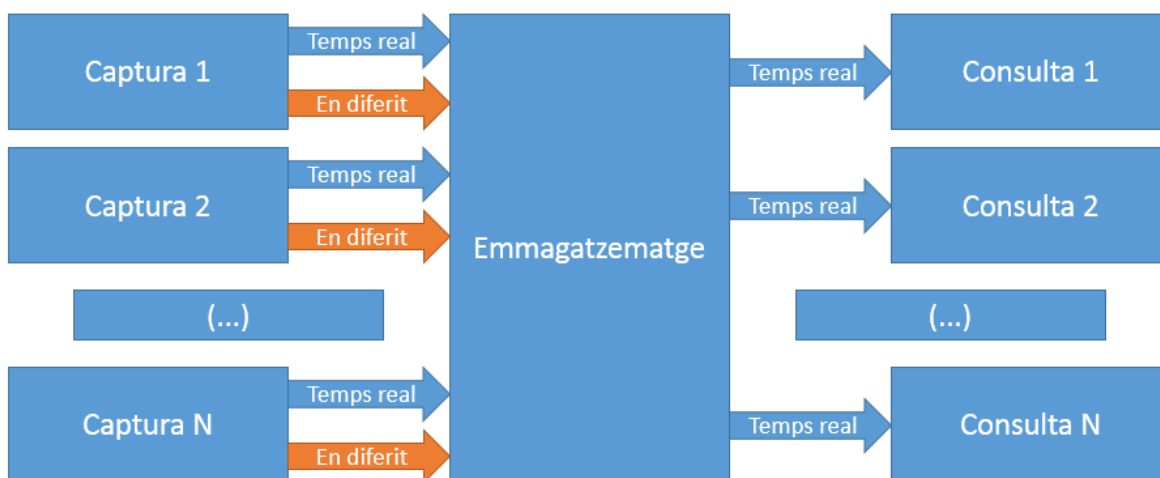


Figura 3.3 - Diagrama funcional de la solució

3.3. Funcionalitat de captura

La funcionalitat de captura és la primera que s'executa i és la que origina les dades que s'aniran propagant per la solució. Aquestes dades s'hauran de captar en diferents localitzacions geogràfiques remotes sense xarxa elèctrica i amb una xarxa de telefonia mòbil sense garanties.

En essència, aquesta funcionalitat es pot segmentar en tres operacions bàsiques:

- **Identificació.** Es tracta d'un procés d'entrada/sortida que interactua amb els participants de la cursa. En aquesta fase, el sistema ha de ser capaç d'identificar inequívocament el participant i propagar l'identificador a la fase següent.
- **Consulta horària.** El sistema ha de tenir un rellotge intern sincronitzat amb el de la cursa. Quan aquest procés s'instancii haurà de fer una consulta al rellotge i guardar el valor.
- **Persistència.** A aquest procés li arriba la parella de dades formada per l'identificador del participant i l'hora de pas. Aquesta parella de dades s'haurà de guardar en un sistema de memòria no volàtil.



Figura 3.4 - Flux bàsic de la funcionalitat de captura

Al final de la fase d'identificació hi ha d'haver un control que detecti si aquell participant ja ha estat registrat anteriorment. En cas de que es detecti aquesta situació el sistema ignorarà la lectura. Aquest s'ha representat de color vermell a la Figura 3.5.

La funcionalitat de captura ha de ser capaç de presentar el recompte de participants en un dispositiu de sortida. A continuació de la funcionalitat de persistència hi ha una funcionalitat optativa. Es tracta de l'enviament de dades en temps real cap a la funcionalitat d'emmagatzematge.

Finalment es passa a la fase de senyalització. El sistema ha de permetre senyalar a l'usuari que el procés de captura ha finalitzat correctament.

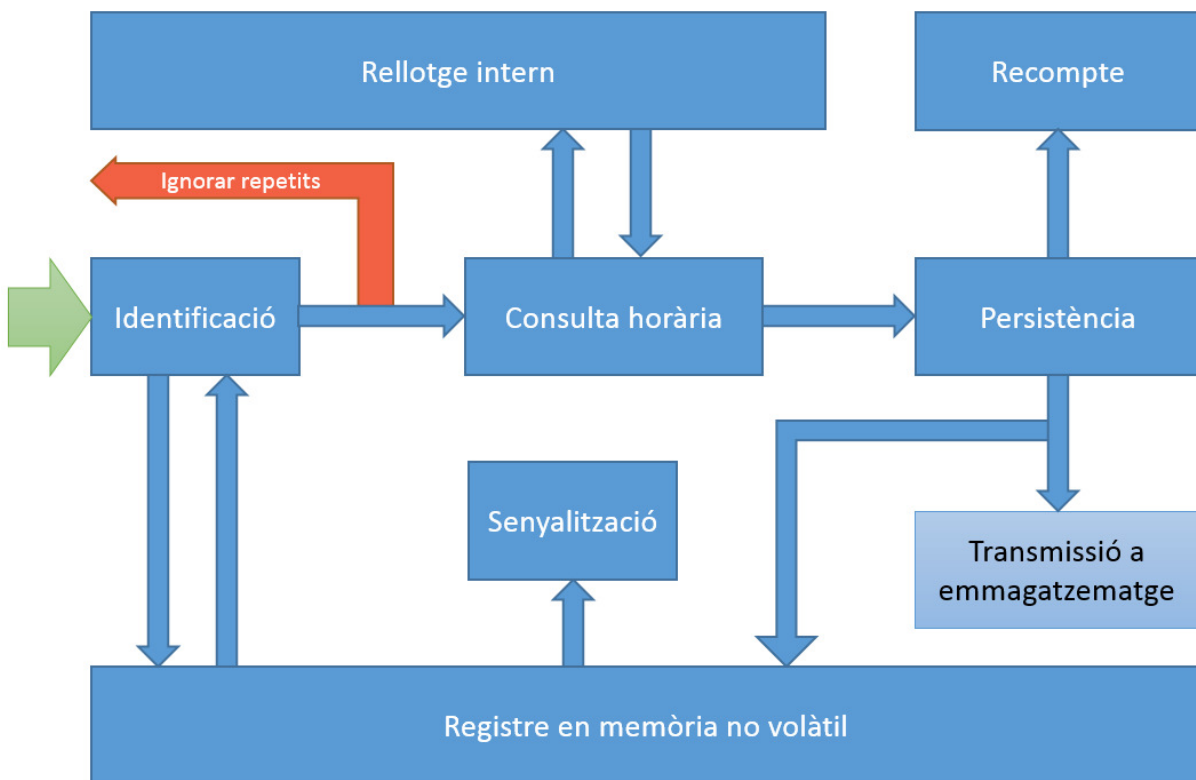


Figura 3.5 - Funcionalitat de captura

3.4. Funcionalitat d'emmagatzematge

El punt d'entrada a aquesta funcionalitat és una interfície que ha de permetre acceptar comunicacions de la funcionalitat de captura. Una vegada s'hagin rebut les dades, la funcionalitat d'emmagatzematge les desarà a un contenidor de dades. Aquesta funcionalitat haurà de dotar al sistema d'un servei per a la publicació de dades.



Figura 3.6 - Flux bàsic de la funcionalitat d'emmagatzematge

Considerant que la funcionalitat de captura serà operada per múltiples instàncies, la interfície d'entrada a la funcionalitat d'emmagatzematge ha de permetre múltiples crides concurrents. Això implica que el

contenedor ha de ser capaç d'atendre peticions concurrents d'emmagatzematge de dades. Al mateix temps, el mòdul de publicació ha de suportar múltiples peticions concurrents de publicació de dades. En resum, això implica que el contenidor s'haurà de dissenyar de manera que pugui atendre peticions concurrents d'entrada i sortida.

No totes les implementacions de la funcionalitat de captura disposaran de la funcionalitat de la transmissió online. Això vol dir que hi haurà d'haver una funcionalitat auxiliar per la càrrega massiva de dades provinents del registre de memòria no volàtil. Per fer la funcionalitat més compacte, el procés de càrrega massiva atacarà la interfície d'entrada.

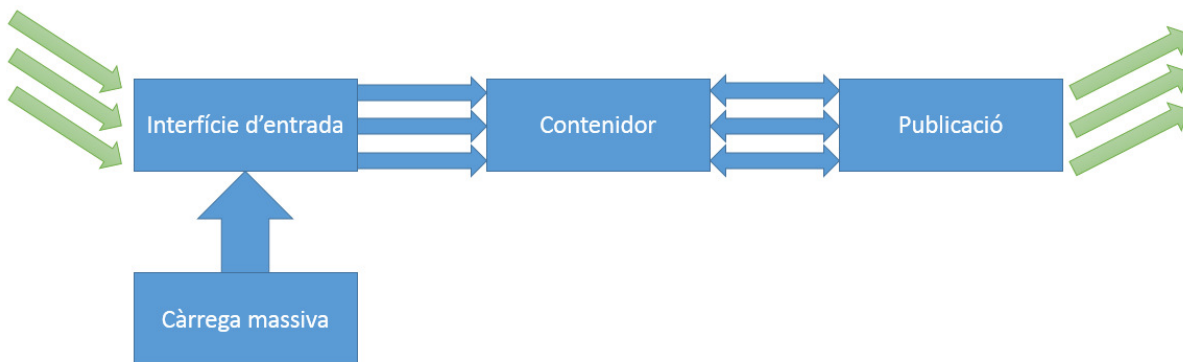


Figura 3.7 - Funcionalitat d'emmagatzematge

3.5. Funcionalitat de consulta

La funcionalitat de consulta serà la interfície amb el món exterior. Aquesta funcionalitat ha de permetre que agents externs a la solució puguin consultar les dades que s'estan generant al sistema mentre s'estan generant. Diferents agents han de poder instanciar aquesta funcionalitat concurrentment. La funcionalitat de consulta haurà de proporcionar les dades a cada agent que les sol·liciti.

A la Figura 3.8 es representa com la funcionalitat de consulta atén als agents externs. Aquest model respondria al model de representació estàtica d'informació: per cada petició d'un agent el sistema li retorna la informació. Però hi ha un requeriment de presentació de dades amb flux continu.

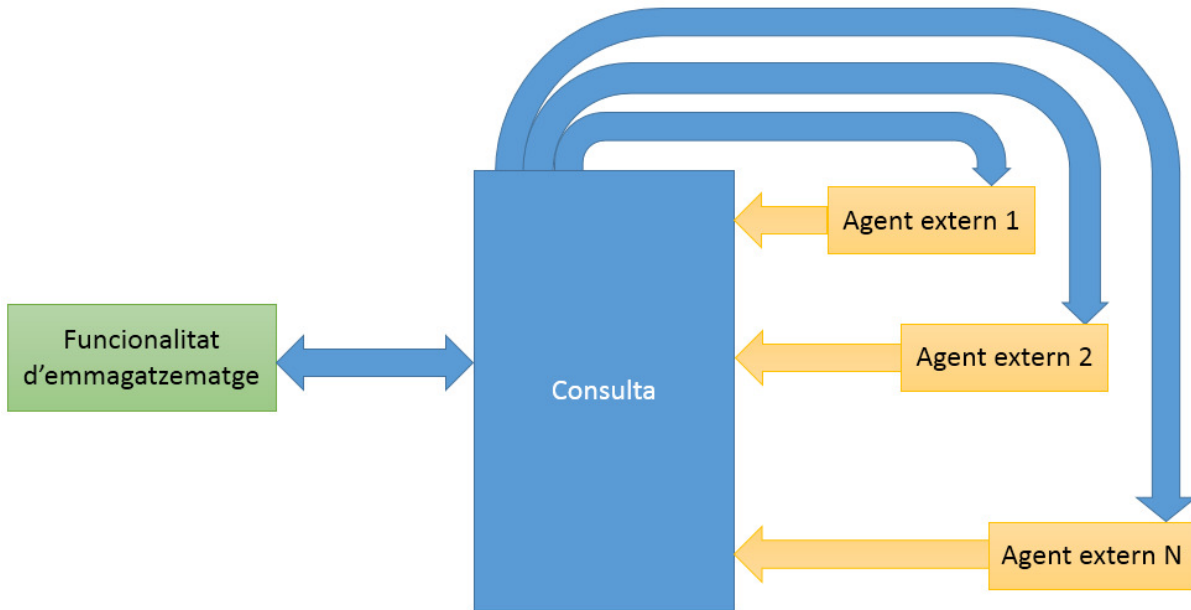


Figura 3.8 - Flux bàsic de la funció de consulta

Per cada agent actiu la funcionalitat de consulta haurà d'establir un enllaç de manera que quan hi hagi una actualització a la funcionalitat d'emmagatzematge aquesta es notifiqui a l'agent extern. La Figura 3.9 representa aquest procés amb un sol agent per simplificar el diagrama.

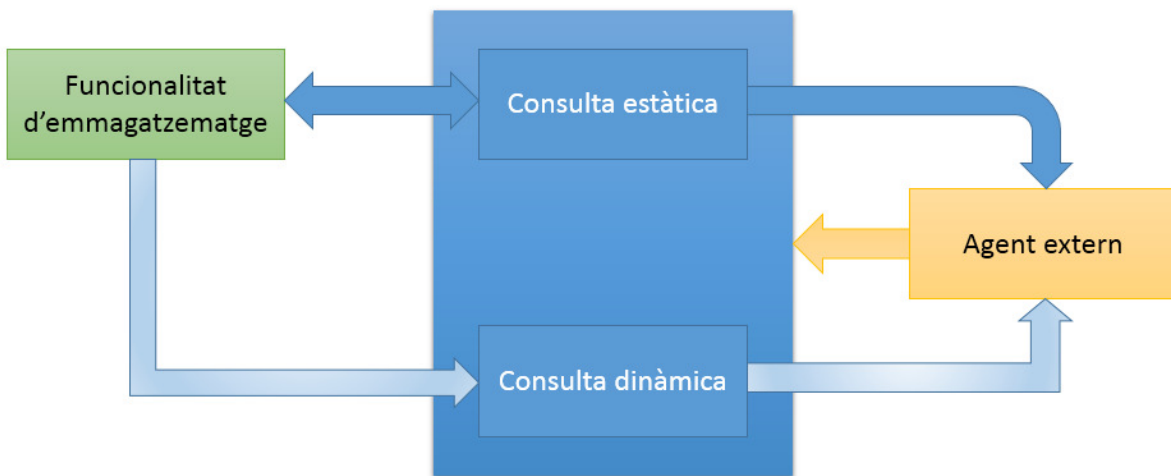


Figura 3.9 - Funcionalitat de consulta

3.6. Resum funcional

A la Figura 3.10 es representa el diagrama funcional detallat de tota la funcionalitat. Per conveniència només es representa una instància de la funcionalitat de captura i consulta.

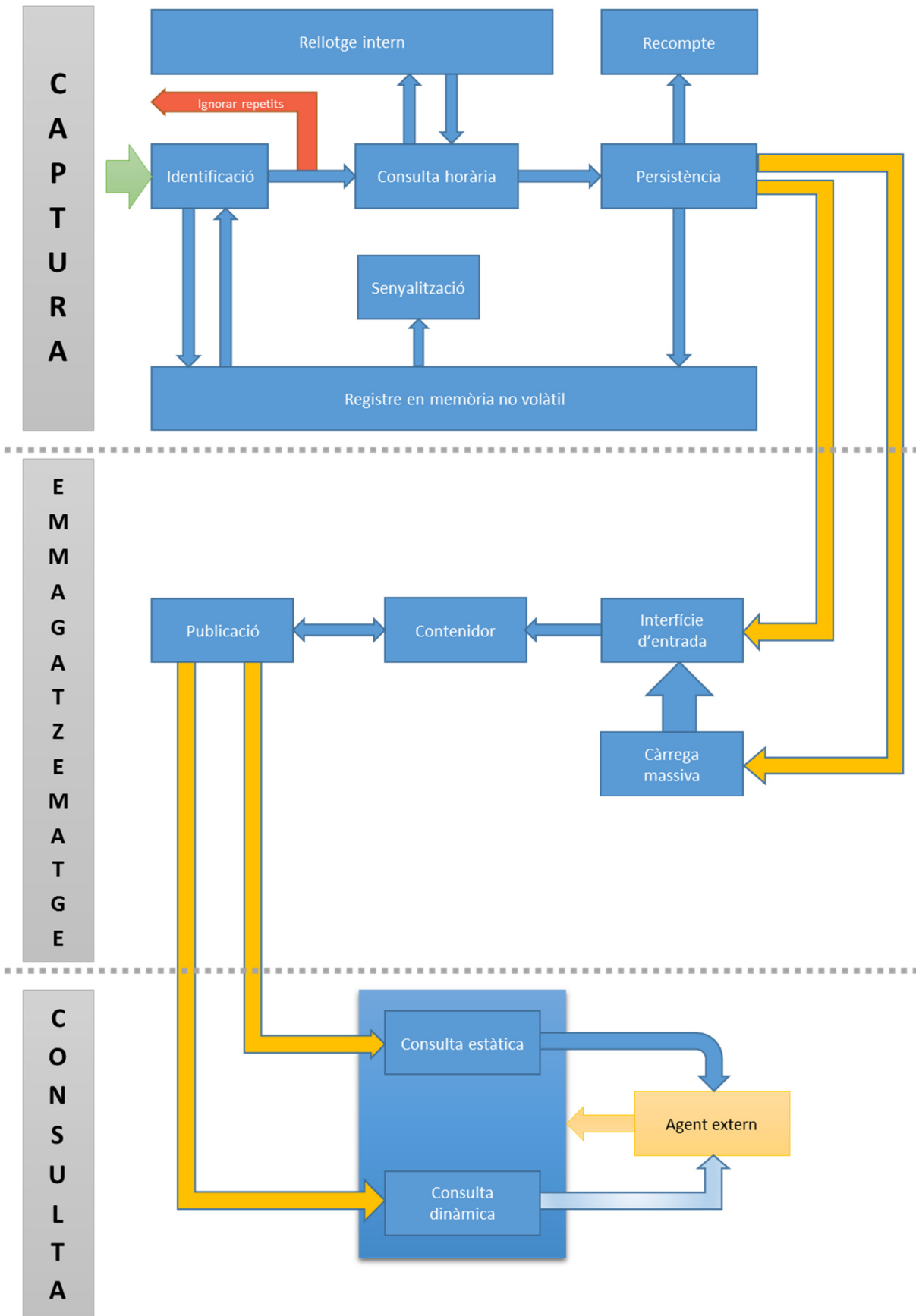


Figura 3.10 - Diagrama funcional detallat de la solució

4. Descripció de la implementació

4.1. Introducció

En aquest capítol s'explica com s'ha passat de la descripció funcional del capítol anterior a una implementació real. La descripció funcional deixa moltes possibilitats per a la implementació de manera que en aquest capítol també és important donar una justificació de les eleccions preses a la fase d'implementació.

El contingut s'ha estructurat seguint la mateixa diferenciació de blocs funcionals definits al capítol anterior: funcionalitat de captura, funcionalitat d'emmagatzematge i funcionalitat de consulta.

4.2. Funcionalitat de captura

4.2.1. Introducció

El desenvolupament de la implementació d'aquesta funcionalitat s'explica sobre el concepte de millora contínua. Aquest concepte consisteix múltiples iteracions de: fer dissenys i implementacions en base als requeriments funcionals coneguts, provar-los i refinar requeriments funcionals.

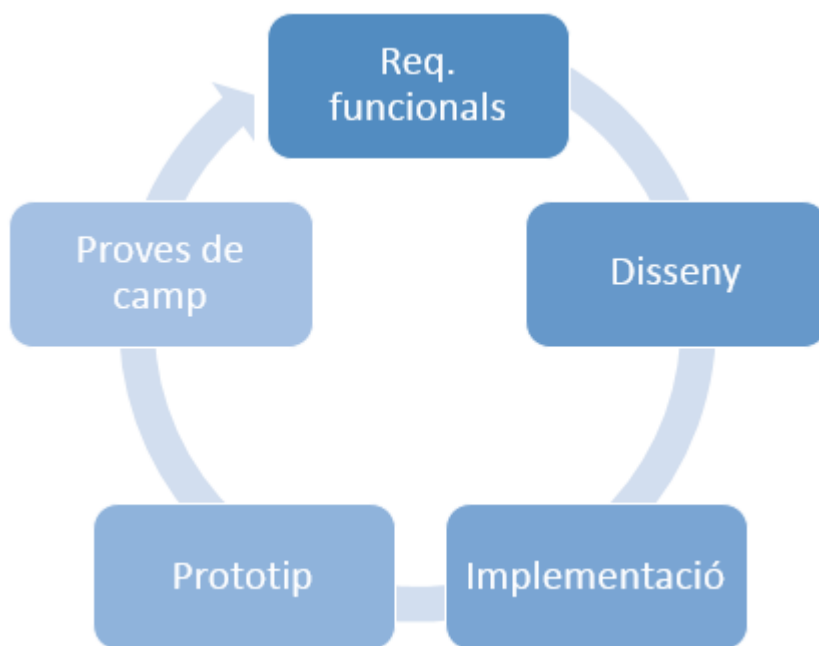


Figura 4.1 – Procés iteratiu de millora contínua

4.2.2. Decisions de disseny perennes

No tots els requeriments funcionals han canviat durant la vida del projecte. En aquest apartat es presenten les decisions de disseny que s'han mantingut invariables de principi a fi del projecte.

Aquesta funcionalitat s'ha d'executar en entorns remots sense xarxa elèctrica. La primera decisió de disseny és implementar aquesta funcionalitat amb dispositius autònoms. Aquests dispositius es fabricaran amb components de maquinari ja existents al mercat i que tinguin les següents característiques: baix cost, una

comunitat de desenvolupadors activa i possibilitats d'expansió. Al haver de treballar a la intempèrie els dispositius s'hauran de confinar en una caixa IP56.

Aquests dispositius s'han de poder transportar per una persona, per exemple, en una motxilla. Aquest fet acota uns certs límits de volum i pes. Aquest fet determinarà que en l'elecció dels components es doni especial importància al consum. La raó és que les bateries són força voluminoses i pesades.

La segona decisió de disseny clau que es pren i que condiciona tota la resta de decisions és utilitzar xips passius RFID [14] per identificar els participants. Aquesta tecnologia és present d'una forma molt àmplia al mercat i és relativament barata. Els xips tenen un volum i pes molt baixos de manera que no es probable que es generi resistència a portar-los per part dels participants. Pel que fa als lectors, el mercat ofereix un gran ventall d'opcions de característiques tècniques: volum, pes, consum, abast... Una altra opció hauria estat la tecnologia NFC que es va descartar pels següents motius:

- Es una tecnologia més recent i per tant amb molta menys penetració al mercat. La disponibilitat de models de maquinari és molt més limitada que en el cas de la tecnologia RFID.
- Relacionat amb el primer punt, la comunitat de desenvolupadors d'aquesta tecnologia és molt inferior.
- L'abast dels lectors és molt inferior. El rang màxim és d'uns 20 centímetres. Això no és un factor limitant per la present implementació però impossibilitaria fer futurs desenvolupaments a una distància superior.

4.2.3. Arquitectura de programari

En aquest apartat es fa un descripció de l'arquitectura del programari dels dispositius de control. L'arquitectura i els casos d'us són comuns a tots els models de prototips presentats als següents capítols.

En primer lloc es presenta el diagrama d'execució a alt nivell a la Figura 4.2. Aquest consta de tres fases:

- **Inicialització.** En aquesta fase es prepara el sistema. Es carreguen les llibreries externes necessàries, es defineixen identificadors pels perifèrics i es defineixen constants.
- **Configuració.** Aquí el sistema s'autoconfigura. A continuació es comprova la presència dels perifèrics i se'n programa el seu comportament.
- **Execució.** En aquesta fase el dispositiu ja està llest per operar la funcionalitat de captura. El dispositiu entra en un estat de repetició infinita que no s'aturarà a no ser que se li talli l'alimentació.



Figura 4.2 - Diagrama d'execució a alt nivell

Des del punt de vista dels casos d'us, els dispositius de control tenen en tenen dos diferents segons l'actor que interactua amb el sistema. L'actor pot ser l'operador de cronometratge o el participant de la cursa. El primer cas d'us és la posada en funcionament del dispositiu per part de l'operador. Seguint les instruccions rebudes, l'operador ha d'alimentar el dispositiu i aquest senyalitza la correcta o incorrecta inicialització. En cas d'inicialització incorrecta l'usuari s'ha de posar en contacte amb el grup de suport.

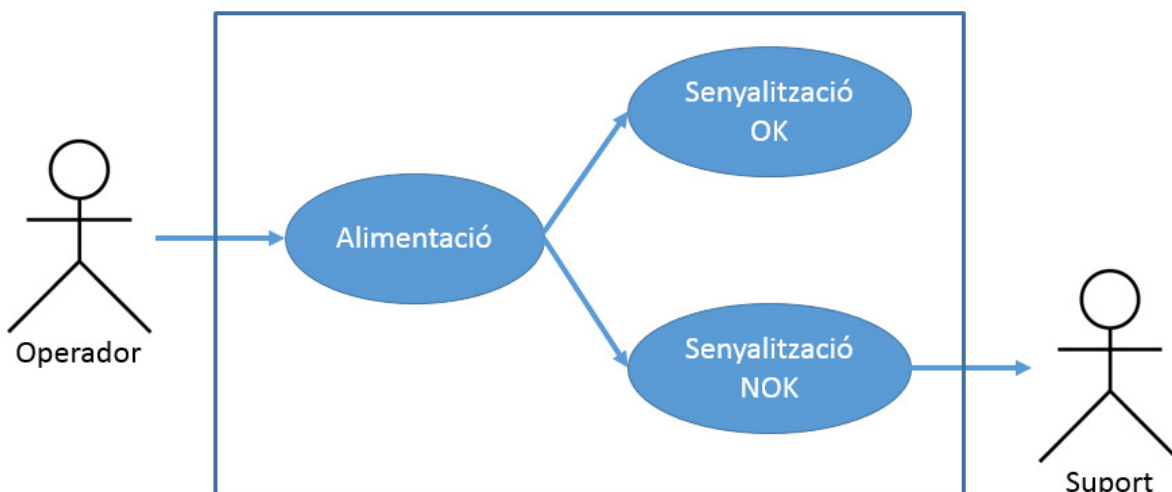


Figura 4.3 - Cas d'us de la posada en funcionament

El segon cas d'us correspon al participant actuant amb el sistema mitjançant la funcionalitat de captura. En tot moment l'operador fa una tasca de supervisió que queda fora del sistema. L'operador és responsable de contactar amb el grup de suport si hi ha qualsevol incidència.

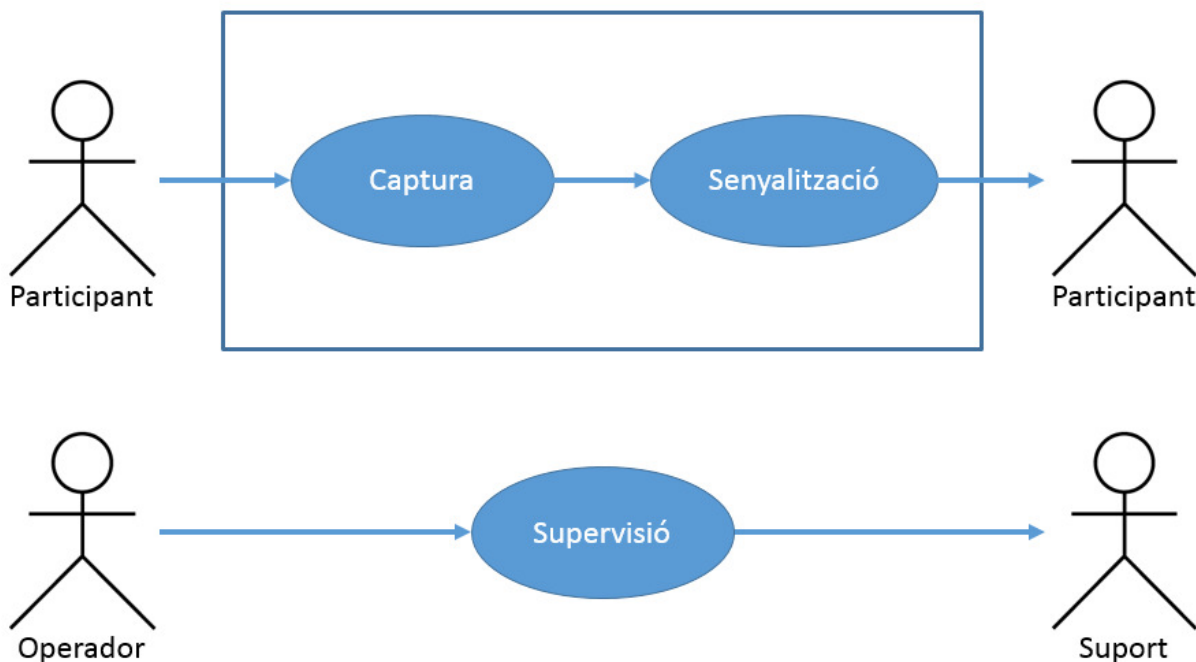


Figura 4.4 - Cas d'us de captura de dades

4.2.4. El primer prototip, ArduinoDLv1

4.2.4.1. Justificació de les decisions de disseny

Els primer prototip es va començar a dissenyar i fabricar sense tenir tots els requeriments funcionals definits. Les funcionalitats no definides, que es mostren a la figura següent, eren: el control de captures repetides, la transmissió de dades a Internet i la visualització del recompte de captures. Aquest primer prototip va rebre el nom de ArduinoDLv1 perquè essencialment és un *datalogger*.

Amb aquests requeriments funcionals limitats disponibles es va decidir apostar per dissenyar el primer prototip de dispositiu de control amb tecnologia Arduino, concretament amb el model Arduino Uno.

L'elecció d'aquesta plataforma va venir donada pel fet de que són dispositius barats, de baix consum i amb unes excel·lents capacitats d'expansió. En contrapartida, aquest model té una capacitat de càlcul i quantitat de memòria força limitades. Al capítol 2.3.1 s'hi pot trobar una comparativa detallada amb altres dispositius.

L'Arduino Uno no disposa de pila. Per tant, quan perd l'alimentació es deixa d'alimentar el rellotge intern i a la següent posada en marxa s'inicialitza amb una hora arbitrària. Per solucionar aquesta mancança s'afegeix una expansió (RTC *shield*) del fabricant Adafruit [15]. Aquesta expansió està formada per una pila i un rellotge.

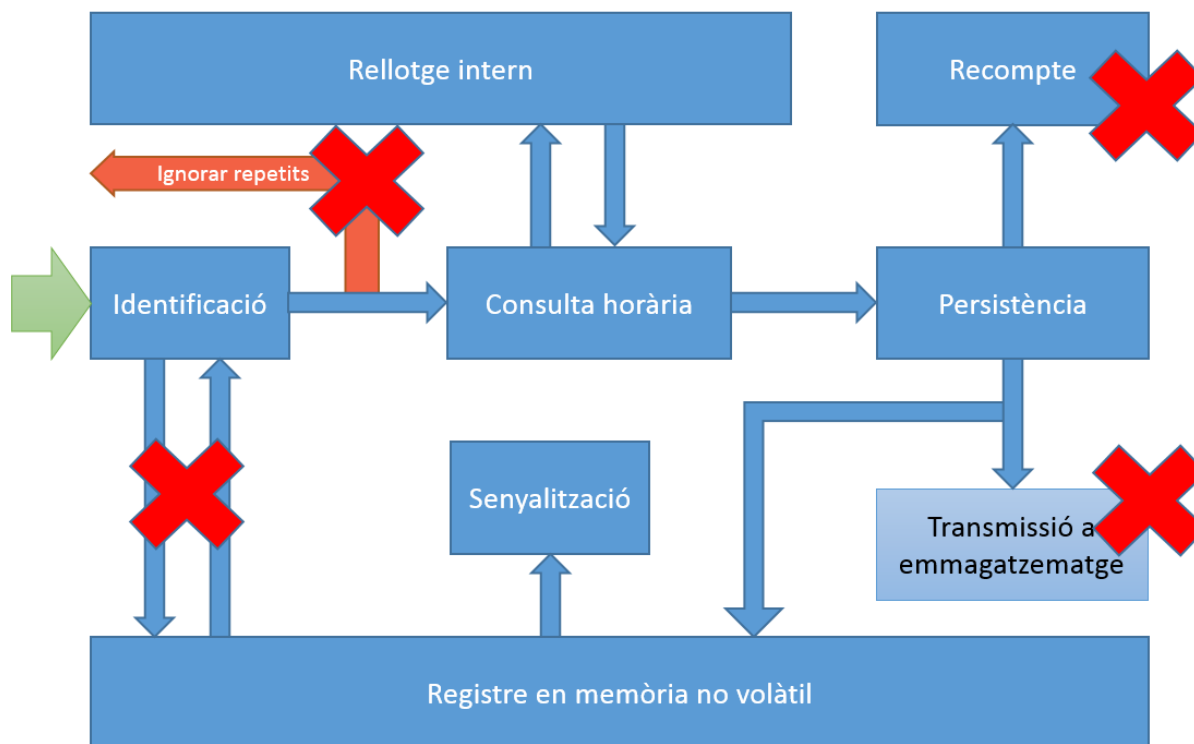


Figura 4.5 - Funcionalitats de captura parcial del model ArduinoDLv1

L'expansió RTC *shield* a part de solucionar la problemàtica temporal porta una interfície de lectura/escriptura de targetes SD. Aquests tipus de suports es consideren suficientment fiables per implementar la funcionalitat de registre en memòria no volàtil.

El següent grup de components que s'havia d'escollir era la parella de xips i lectors de tecnologia RFID. El proveïdor seleccionat va ser un proveïdor local, AERF de Vic [16]. La decisió es va veure motivada pel fet de que tenien uns preus competitiu i un lector de baix consum amb abast suficient. La proximitat del proveïdor també va ser un fet molt apreciat a l'hora de prendre aquesta decisió. Un últim punt a favor d'aquest producte va ser que porta un avisador acústic; aquest dispositiu es va valorar positivament perquè proveïa d'un sistema de senyalització de les captures als usuaris.

4.2.4.2. Anàlisi de l'autonomia del dispositiu

Abans de seleccionar el tipus de bateria calia fer una anàlisi de consum per tal de fer una elecció que garantís complir amb els requeriments: pes i volum raonables i una autonomia de tot el sistema de fins a 15 hores.

El mercat ofereix essencialment dues tecnologies de bateries, les de plom i les d'ions de liti [17]. Les bateries de plom tenen un preu molt més baix que les d'ions de liti. Per exemple, una bateria de 12V i 7Ah de plom es pot trobar el mercat per uns 13€ mentre que una de liti val uns 110€. El sobre cost de les bateries de liti ve acompanyat d'una sèrie d'avantatges com que tenen molt menys efecte memòria i una auto-descàrrega molt més baixa. A la Figura 4.6 es pot veure una comparativa entre ambdues tecnologies.

Tipus	Preu	Memòria	Auto-descàrrega
Plom	Baix	Sí	Sí
Ions de liti	Alt	No	No

Figura 4.6 - Comparativa entre bateries de plom i d'ions de liti

Com que una de les premisses del projecte era el baix cost es va optar per adquirir bateries de 12V i 7Ah de la marca EnergiVM [18]. Aquestes bateries són les que s'han fet servir a tots els models de dispositius de control.

Per tal de fer una validació teòrica de l'autonomia, primer s'han de calcular la potència de cada component tenint en compte la seva tensió i consum com es mostra a la Figura 4.7.

Component	Tensió (V)	Consum (mA/h)	Potència (mW)
Arduino	5	100	500
RTC SD Shield	3,3	20	66
Lector RFID	12	100	1200

Figura 4.7 - Càlcul de potència elèctrica dels components del ArduinoDLv1

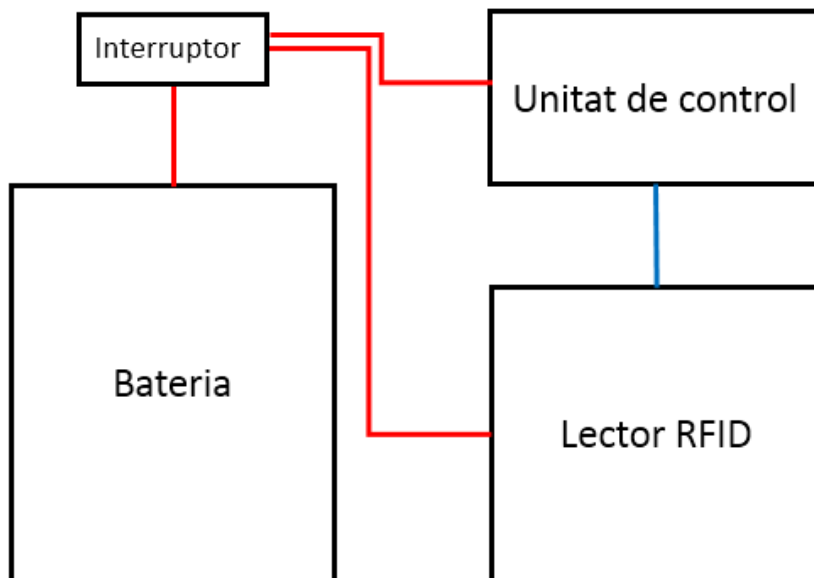
Una vegada s'han calculat les potències es pot procedir a calcular els requeriments d'entrega de corrent per una bateria de 12V tal i com es mostra a la Figura 4.8. Segons els requeriments del projecte els dispositius de control han de tenir una autonomia de 15 hores. El resultat final és que aquesta configuració requereix una bateria de 2,2Ah, per tant, la bateria seleccionada de 7Ah compleix els requeriments.

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Arduino	500			
RTC SD Shield	66			
Lector RFID	1200			
	1766	15	26.490	2,2

Figura 4.8 - Consum de l'ArduinoDLv1

4.2.4.3. Disposició dels components

Els components que conformen l'ArduinoDLv1 estan confinats en una caixa amb un nivell d'estanqueïtat IP56. Aquesta caixa estanca té les següents cotes: 225x175x95mm. La disposició a la caixa segueix el patró definit a la Figura 4.6. La bateria alimenta la unitat de control i el lector RFID. L'alimentació està controlada per un interruptor que està situat a l'exterior de la caixa estanca.



Llegenda:

- Vermell: cables d'alimentació
- Blau: cables de comunicació sèrie

Figura 4.9 Diagrama modular del ArduinoDLv1

L'Arduino està dissenyat de manera que s'hi puguin annexar extensions (*shields*). Aquestes extensions s'apilen a sobre l'Arduino. A la Figura 4.7 es pot veure un Arduino Uno R3 amb un *Real Time Clock (RTC) + SD shield* a sobre.

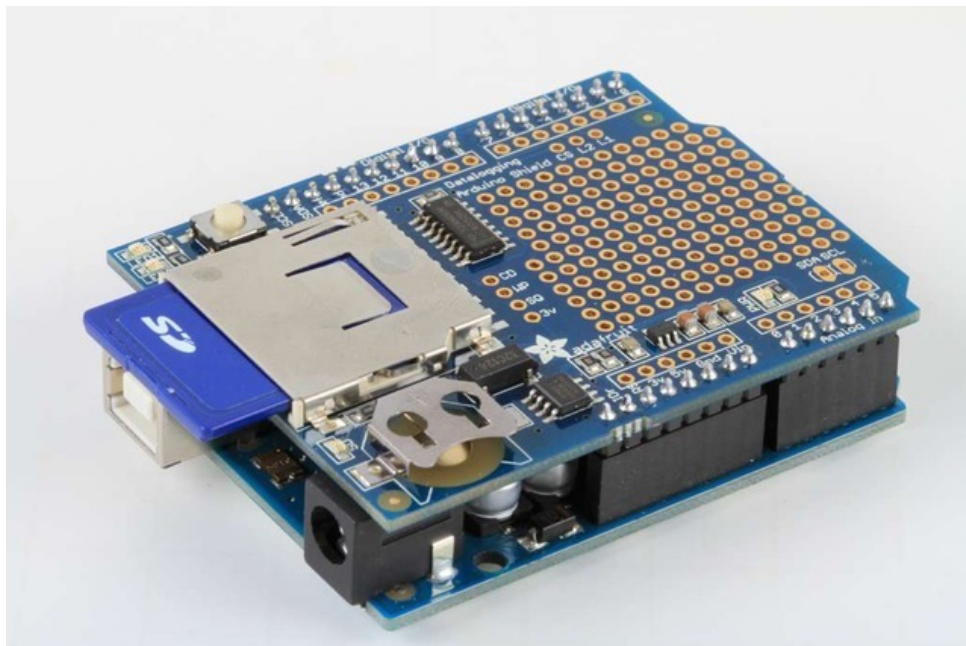


Figura 4.10 - Arduino amb un Real time + SD shield

4.2.4.4. Descripció de les operacions a nivell de programari

En aquest apartat es passa a fer una descripció de les operacions que es fan a cada una de les fases definides l'apartat 4.2.3: inicialització, configuració i execució. La fase d'inicialització no implementa cap operació, només s'encarrega de carregar les llibreries externes necessàries i defineix paràmetres de configuració en forma de constants.

- **Configuració:**
 - Configuració dels ports de comunicacions i inicialització dels perifèrics
 - Control de presència de la targeta SD. Si no es detecta es senyalitza mitjançant el LED vermell en mode intermitent. El procés es queda en aquest mode de senyalització i no progressa a cap fase posterior.
 - S'interroga el selector d'identificador de control i es guarda el valor.
 - S'obre, o es crea si no existeix, el fitxer de persistència de la SD.
 - Configuració acabada correctament. Es senyalitza aquest fet amb tres intermitències del LED verd.
- **Execució:**
 - Escoltar indefinidament el port de comunicacions del lector RFID fins que arribin dades. Quan arribin dades passar a fer les operacions següents.
 - Descodificar el número de dorsal.
 - Interrogar l'hora actual del RTC *shield*.
 - Escriure les dades al fitxer de persistència.
 - Senyalitzar la lectura correcta amb dues intermitències del LED verd.
 - Tornar al primer punt.

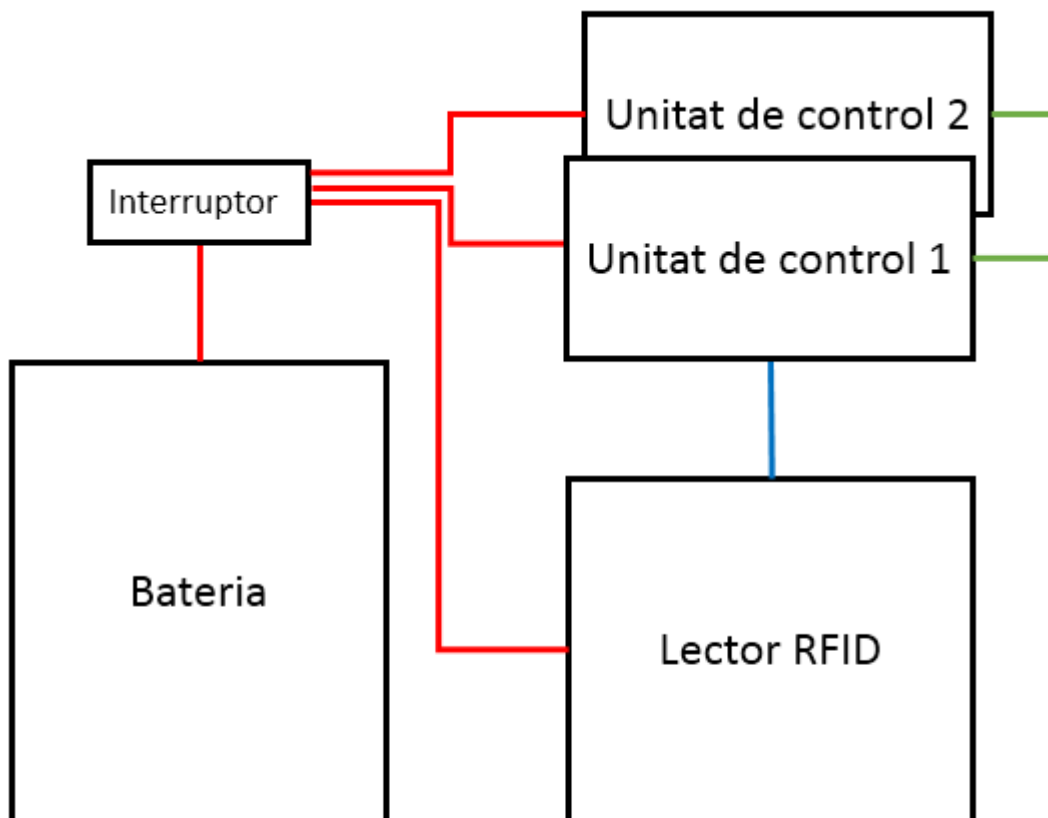
4.2.5. Datalogger amb funcionalitats de xarxa, ArduinoRT

L'ArduinoRT és una evolució del ArduinoDLv1. El repte que intenta resoldre és enviar les dades a Internet tal i com es llegeixen. Aquesta funcionalitat es pot identificar a la Figura 4.6 amb el nom "Transmissió a emmagatzematge".

4.2.5.1. Evolucions del maquinari

La novetat pel que fa a maquinari d'aquest model és el GSM Shield [19]. Es tracta d'una expansió que aporta un mòdem GPRS que es comunica amb l'Arduino mitjançant *software* serial. Seguint la filosofia de l'Arduino d'anar apilant expansions primerament es va posar a sobre del RTC + SD Shield. Aquesta solució es va haver de descartar perquè de seguida s'omplia la memòria de dades SRAM de 2KB provocant *stack overflows* [20].

Es va optar per solucionar-ho afegint un segon Arduino amb el GSM Shield (ArduinoGSM a partir d'ara). L'ArduinoGSM es comunica amb l'Arduino principal a mitjançant un bus i2C. Per tal de senyalitzar quan l'ArduinoGSM està fora de línia se li afegeix un LED vermell a fora de la caixa.



Llegenda:

- Vermell: cables d'alimentació
- Blau: cables de comunicació sèrie
- Verd: cables de comunicació i2C

Figura 4.11 – Diagrama modular de l'ArduinoRT

L'espai disponible a la caixa estanca era molt reduït. Tal i com representa la figura 4.11, la solució que es va adoptar va ser posar les dues unitats de control amb els seus respectius *shields* un a sobre l'altre. Anant de sota cap a sobre, l'ordre dels components és el següent: Arduino, GSM Shield, capa protectora i aïllant, Arduino i RTC + SD Shield.

4.2.5.2. Anàlisi de l'autonomia del dispositiu

Segons els canvis de maquinari que s'han definit a l'apartat anterior, es procedeix a fer el càlcul teòric de l'autonomia del dispositiu tal i com es mostra a la Figura 4.12.

Component	Tensió (V)	Consum (mA/h)	Potència (mW)
Arduino	5	100	500
RTC SD Shield	3,3	20	66
Lector RFID	12	100	1.200
Arduino	5	100	500
GSM Shield	5	700	3.500

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Arduino	500			
RTC SD Shield	66			
Lector RFID	1200			
Arduino	500			
GSM Shield	3500			
	5766	15	86.490	7,2

Figura 4.12 - Càlcul teòric de l'autonomia del ArduinoRT

Aquest dispositiu requeriria d'una bateria de 12V i 7,2mA. Així doncs, la bateria de 12V i 7mA no seria suficient per complir el requeriment de 15 hores d'autonomia i es quedaria amb una autonomia teòrica de 14 hores i mitja.

4.2.5.3. Canvis a nivell de programari

L'Arduino principal només es veu afectat per unes petites modificacions en les fases de configuració i execució:

- **Configuració:**
 - Inicialització de la llibreria de comunicacions pel bus i2C.
- **Execució:**
 - Quan s'ha acabat de processar una lectura i s'ha guardat a persistència es procedeix a enviar a l'Arduino GSM una cadena de caràcters que inclou l'identificador del punt de control, el dorsal i l'hora de pas.

L'ArduinoGSM per la seva banda implementa noves operacions en les següents fases:

- **Configuració:**
 - Inicialització de la llibreria de comunicacions pel bus i2C.
 - Registre d'una rutina que es cridarà quan es rebin comunicacions pel bus i2C.
 - Inicialització de la llibreria del GSM Shield.
 - Donar corrent al LED vermell
- **Execució.**
 - Quan la rutina de servei al i2C rep comunicacions emmagatzema el missatge i aixeca el *flag* de missatge pendent.

- Si no hi ha connexió executar les instruccions per connectar el mòdem. Si acaben correctament apagar el LED vermell.
- Si hi ha connexió i el *flag* de missatge pendent està aixecat enviar el missatge.
- Tornar al segon punt.

Amb aquesta implementació final un missatge només tenia una oportunitat per ser enviat. Es va intentar implementar controls més robusts basats en cues de missatges pendents d'enviar però va ser impossible degut a les limitacions de memòria de dades de l'Arduino.

4.2.6. ArduinoDLv2, un ArduinoDLv1 amb pantalla

L'ArduinoDLv2 és purament una iteració del ArduinoDLv1 amb algunes millores. Implementa les noves funcionalitats: ignorar repetits i recompte de lectures. La funcionalitat de recompte de lectures ve com una imposició derivada del pla de seguretat de la cursa.

Afegint una pantalla com a interfície de sortida s'elimina la necessitat de tenir dispositius LED per fer les diferents senyalitzacions;

4.2.6.1. Evolucions del maquinari

Aquesta iteració té una novetat important en forma d'Arduino Shield. Es tracta d'un LCD Shield que incorpora cinc polsadors, un regulador mecànic de contrast i una pantalla LCD de 16x2 caràcters [21]. Aquesta expansió es comunica amb la unitat de control, l'Arduino, mitjançant el bus i2C.

4.2.6.2. Anàlisi de l'autonomia del dispositiu

Segons els canvis de maquinari que s'han definit a l'apartat anterior, es procedeix a fer el càlcul teòric de l'autonomia del dispositiu tal i com es mostra a la Figura 4.12.

Component	Tensió (V)	Consum (mA/h)	Potència (mW)
Arduino	5	100	500
RTC SD Shield	3,3	20	66
Lector RFID	12	100	1200
LCD Shield	5	40	200

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Arduino	500			
RTC SD Shield	66			
Lector RFID	1200			
LCD Shield	200			
	1966	15	29.490	2,5

Figura 4.13 - Càlcul teòric de l'autonomia del ArduinoDLv2

Aquest dispositiu requeriria d'una bateria de 12V i 2,5mA. Tenint en compte que la bateria proporcionada és de 12V i 7mA, es pot afirmar que seria més que suficient.

4.2.6.3. *Canvis a nivell de programari*

Es parteix de les operacions implementades al ArduinoDLv1 i s'hi afegeixen les següents modificacions:

- **Configuració.**
 - Inicialització de la llibreria del LCD Shield.
- **Execució.**
 - A mesura que es van llegint dorsals, es guarden els tres últims.
 - Si la lectura actual no és cap de les tres anteriors en continua processant i es senyalitza per la pantalla LCD. Sinó s'ignora i el sistema es deixa a punt per la següent lectura.
 - Es manté un comptador de lectures.
 - Constantment la pantalla LCD mostra el nombre de lectures correctes i l'hora de l'última.

4.2.7. *rPi-eth, una nova plataforma de treball*

El dispositiu de control rPi-eth és un concepte completament nou d'unitat de control. El canvi d'Arduino cap a Raspberri Pi bé donat per les limitacions de memòria detectades a l'ArduinoRT. La funcionalitat d'enviar dades a Internet, tot i no ser crítica, és un punts que realment afegeixen valor afegit a la solució de cronometratge. Al no tenir la possibilitat de fer un control d'errors robust, la funcionalitat de temps real implementada a l'ArduinoRT estava molt lluny de cobrir les expectatives.

4.2.7.1. *Justificació de les decisions de disseny*

De cara a l'edició 2015 l'organització decideix dotar dos punts de control amb Internet a través d'una interfície Ethernet. El desplegament de les solucions de xarxa per part de l'organització queden fora de l'abast d'aquest projecte.

L'elecció de la Raspberry Pi B+ es va fer tenint en compte els criteris presentats al capítol 2.3.1. En resum, amb un cost similar s'aconsegueix una capacitat de procés i memòria molts ordres de magnitud superiors.

Des del punt de vista de la funcionalitat de captura aquest és el primer model que l'implementa completament (Figura 4.8). Per la funció d'identificació es continua fent servir el mateix mòdul lector RFID. Es decideix solucionar la problemàtica del rellotge fent us de la connectivitat a Internet i s'implementa a nivell de programari amb un client NTP [22]. Pel que fa a la senyalització s'opta per una pantalla LCD *shield* [23] especialment dissenyada per la Raspberry Pi. Es tracta d'una pantalla de 16x2 caràcters, cinc polsadors i un regulador mecànic de contrast. La Raspberry Pi requereix d'una targeta SD per tal d'emmagatzemar-hi el sistema operatiu i dotar a aquest de funcions de disc dur. Aquesta targeta SD serà utilitzada per fer les operacions de persistència en memòria no volàtil.

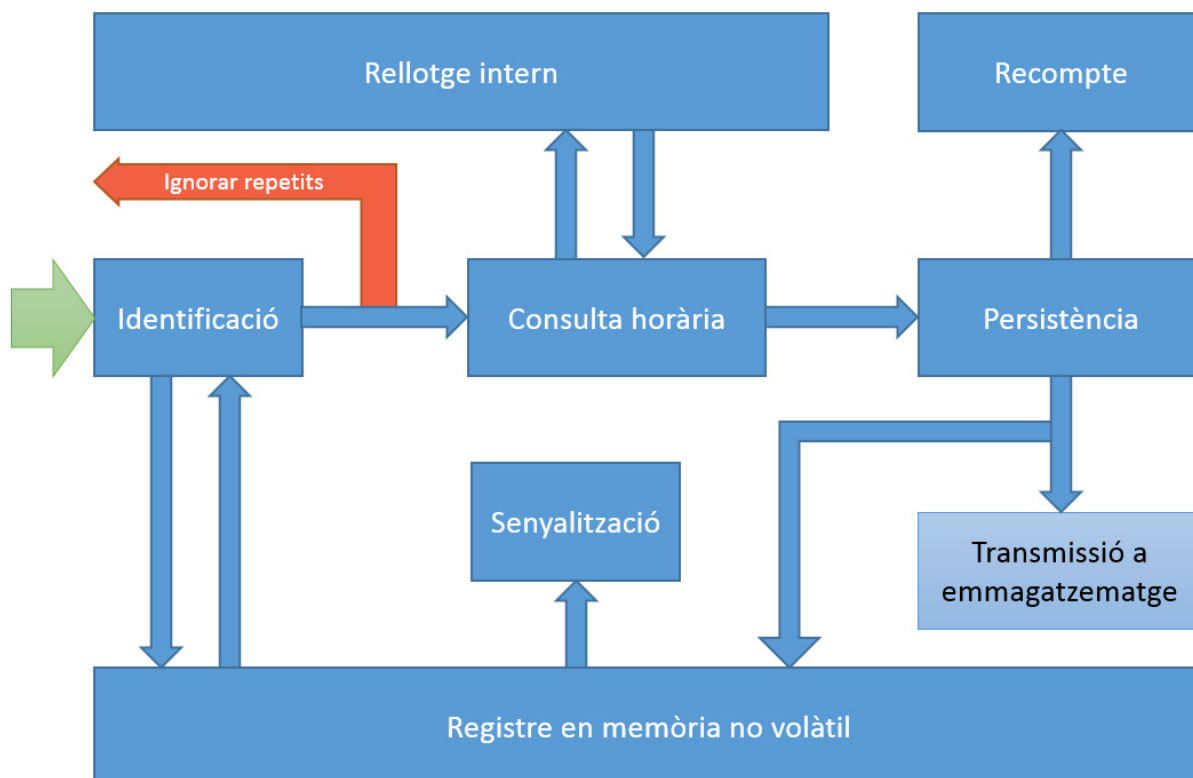
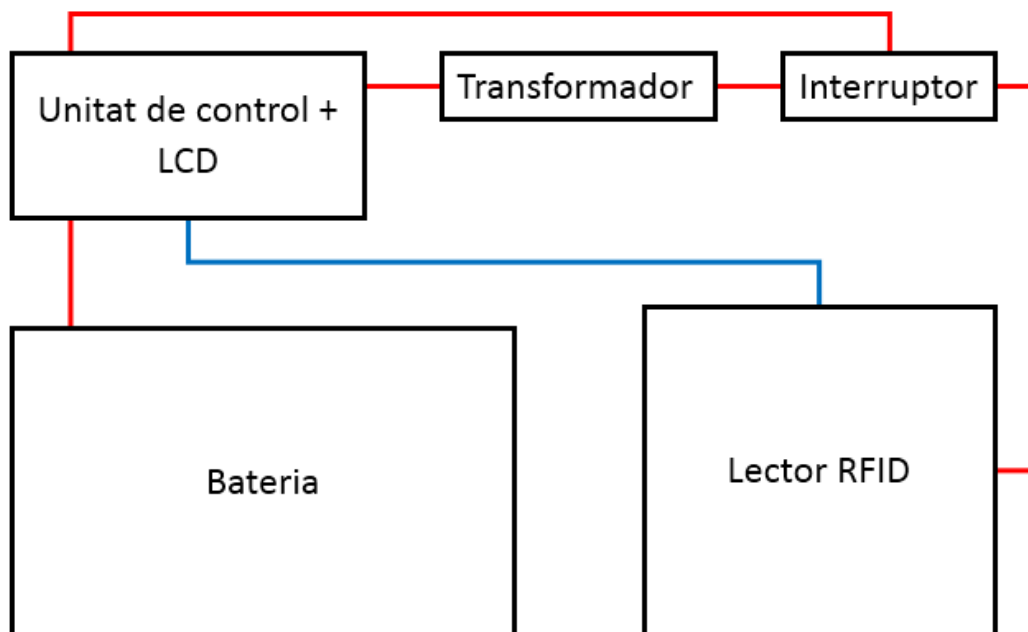


Figura 4.14 - Funcionalitat de captura

4.2.7.2. Disposició dels components

La Raspberry Pi ocupa un volum sensiblement més gran que l'Arduino. En conseqüència, el seu LCD Shield també és més voluminós. Si s'hi suma que es va decidir encapsular la unitat de control i el LCD Shield dins d'un estoig protector de metacrilat resulta que la caixa estanca que s'havia fet fins ara no tenia un volum suficient i es va haver de buscar una alternativa. La nova caixa estanca és del fabricant Himel [24] i té les següents cotes: 300x220x230mm. El volum d'aquesta caixa és clarament superior a les necessitats d'aquest model però es va seleccionar així anticipant futurs desenvolupaments. A l'igual que el model que s'havia fet servir pels models basats en Arduino, té un coeficient d'estanqueïtat IP56.

El voltatge d'entrada que admet la Raspberry Pi és d'entre 4,75V i 5,25V. La bateria és de 12V per tant, s'ha d'incorporar un transformador de 12V a 5V. A la Figura 4.15 s'ha representat la disposició a la caixa estanca dels components que conformen aquest model.



Llegenda:

- Vermell: cables d'alimentació
- Blau: cables de comunicació sèrie

Figura 4.15 - Disposició modular del rPi-eth

4.2.7.3. Anàlisi de l'autonomia del dispositiu

Segons la configuració de maquinari que s'ha definit a l'apartat anterior, es procedeix a fer el càlcul teòric de l'autonomia del dispositiu tal i com es mostra a la Figura 4.16.

Component	Tensió (V)	Consum (mA/h)	Potència (mW)
Raspberry Pi	5	700	3500
LCD Shield	5	40	200
Lector RFID	12	100	1200
Transformador 12V-5V	12	10	120

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Raspberry Pi	3500			
LCD Shield	200			
Lector RFID	1200			
Transformador	120			
	5020	15	75.300	6,3

Figura 4.16 - Càlcul teòric de l'autonomia del rPi-eth

Aquest dispositiu requeriria d'una bateria de 12V i 6,3mA. Tenint en compte que la bateria proporcionada és de 12V i 7mA, es pot afirmar que seria suficient.

4.2.7.4. *Descripció de les operacions a nivell de programari*

A diferència de l'Arduino, la Raspberry Pi necessita un sistema operatiu. Tot i haver-hi força opcions s'ha seleccionat Raspbian [25] per ser un dels més utilitzats per la comunitat de desenvolupadors. S'ha utilitzat la mateixa arquitectura de programari definida al apartat 4.2.3: inicialització, configuració i execució. A diferència de la plataforma Arduino que el dispositiu arrenca automàticament en mode de configuració, a la Raspberry Pi s'ha de programar aquesta operació.

Una novetat d'aquest model és que implementa un registre intern d'operacions en forma de fitxer de text. Aquesta funció no persegueix cap funcionalitat final sinó que facilita el desenvolupament i l'anàlisi de les operacions realitzades durant el funcionament del dispositiu de control.

També s'implementen operacions noves a la funcionalitat de senyalització accessibles a través dels botons que incorpora el LCD Shield:

- Pantalla per defecte: mostra el nom del control, l'identificador, l'hora de l'última lectura correcta i el nombre de lectures.
- Visualització dels vuit últims dorsals llegits correctament.
- Visualització de la direcció IP del dispositiu.

Amb la voluntat de que el codi font sigui totalment reutilitzable es crea un fitxer de configuració que emmagatzemarà l'identificador del punt de control, el nom del control i el temps d'il·luminació de la pantalla.

Per tal d'estalviar energia es crea una funció que para la il·luminació de la pantalla LCD cada X segons. X és un paràmetre que es llegeix del fitxer de configuració. Cada vegada que es presenten dades noves a la pantalla aquesta s'il·lumina i es dispara un temporitzador que pararà la il·luminació passats X segons.

Segons els requeriments del projecte la funcionalitat de lectura i persistència de dorsals és crítica mentre que la funcionalitat d'enviament en temps real és optativa. Per assegurar la no interferència entre les dues funcionalitats es decideix implementar-les de manera que s'executin com a fils diferents. Per tant, aquesta implementació s'haurà de veure des de dos punts de vista diferents: fil principal i fil de temps real. Quan el principal llança el fil de temps real ho fa sense cap mena de control: el llança i se n'oblida per tornar de seguida a la tasca crítica. Serà el fil de temps real que implementarà el control d'errors d'enviament de dades a Internet.

A continuació es passen a detallar les operacions de cada fase del fil principal.

- **Inicialització.**
 - Quan la càrrega del sistema operatiu ha acabat de carregar cal automatitzar l'execució del fil principal que implementa les fases de configuració i execució.

- **Configuració.**

- Carregar les llibreries i inicialitzar els perifèrics.
- Llegir el fitxer de configuració que controla els paràmetres: identificador del control, nom del control i temps d'il·luminació de pantalla.
- Crear (si no existeixen) un fitxer de registre i el fitxer de persistència de dades.
- Inicialització de la llista de les últimes vuit lectures.
- Esperar que hi hagi connexió a Internet i actualitzar l'hora actual mitjançant una crida al client NTP. Si no s'aconsegueix connectar a Internet el dispositiu entra en un estat d'excepció que es notifica mitjançant la pantalla LCD.
- Inicialitzar el temporitzador que desactiva la il·luminació de la pantalla.
- Presentar la pantalla per defecte.

- **Execució.**

- Escoltar indefinidament el port de comunicacions del lector RFID fins que arribin dades. Quan arribin dades passar a fer les operacions següents:
- Descodificar el número de dorsal.
- Interrogar l'hora actual del sistema operatiu.
- Escriure les dades al fitxer de persistència.
- Llançar un nou fil de temps real.
- Presentar la pantalla per defecte.
- Tornar al primer punt.

En la fase de configuració, a mesura que es van fent les operacions ens van escrivint missatges de progrés per pantalla. L'objectiu d'aquests missatges és informar a l'operador del progrés de l'arrancada del dispositiu. Abans de presentar la pantalla per defecte informa de que la fase d'arrencada ha acabat i el dispositiu està a punt per treballar.

A continuació es passen a detallar les operacions del fil de temps real. En aquest cas només hi ha operacions d'execució perquè la configuració la fa el fil principal.

- **Execució.**

- Aquest fil rep per paràmetre l'identificador de control, el dorsal i el temps. A partir d'ara "les dades".
- Es comprova si hi ha connexió a Internet. Si no n'hi ha:
 - Es guarden les dades a un fitxer de pendents d'enviar.
- En cas de que hi hagi connexió a Internet:
 - S'intenten enviar les dades.
 - Si falla s'afegeixen les dades al fitxer de pendents d'enviar.

- Si ha acabat correctament, es comprova si hi ha pendents d'enviar. Si n'hi ha se n'intenta enviar un.

4.2.8. rPi-GPRS, el dispositiu final

El model rPi-eth implementa totes les funcionalitats però assumeix que disposarà de connectivitat a Internet mitjançant una interfície Ethernet. Aquesta assumpció no és vàlida per totes les localitzacions dels punts de control. El model rPi-GPRS és essencialment un rPi-eth amb capacitat per connectar-se a internet mitjançant un mòdem que interactuarà amb la xarxa de telefonia mòbil.

4.2.8.1. Evolucions del maquinari

La primera opció que es va provar per dotar de connexió a Internet a aquest dispositiu va ser connectant un mòdem USB a la Raspberry Pi a través d'un *hub* USB. S'havia de posar aquest *hub* entremig perquè la Raspberry Pi no és capaç de donar els 500mA que necessita el mòdem a través del port USB. Aquesta opció es va acabar descartant perquè acabava donant un comportament molt erràtic a tot el sistema.

La opció definitiva va venir incorporant un *router* Mikrotik mAP2n [bibilio]. Es tracta d'un dispositiu de molt baix consum, molt petit, lleuger, barat i totalment configurable. No es va trobar cap alternativa al mercat que tingués unes característiques similars. Aquest *router* proveeix de dues interfícies Ethernet, una interfície sense fils i un port mini-USB per connectar el mòdem. El mòdem seleccionat és el Huawei E-303 [26]. Al mercat hi ha una infinitat de mòdems USB amb característiques similars. Es va triar aquest perquè el seu firmware crea una porta d'enllaç que és molt fàcil d'enllaçar amb un Mikrotik.

4.2.8.2. Anàlisi de l'autonomia del dispositiu

Aquesta nova configuració requereix dos dispositius nous: el mòdem USB i el *router* Mikrotik:

Component	Tensió (V)	Consum (mA/h)	Potència (mW)
Raspberry Pi	5	700	3500
Transformador 12V-5V	12	10	120
LCD Shield	5	40	200
Lector RFID	12	100	1200
Mikrotik	12	170	2040
Mòdem USB	5	500	2500

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Raspberry Pi	3500			
Transformador	120			
LCD Shield	200			
Lector RFID	1200			
Mikrotic	2040			
Mòdem USB	2500			
	9560	15	143.400	12,0

Figura 4.17 - Càlcul teòric de l'autonomia del rPi-eth

Com es veu a la Figura 4.17, aquest dispositiu requeriria d'una bateria de 12V i 12mA. Tenint en compte que la bateria proporcionada és de 12V i 7mA, està clar que és insuficient. Es decideix segmentar els components amb dos grups alimentats per dues bateries. Una altra opció hauria estat comprar una bateria de 12V i 12mA però es va decidir reutilitzar les bateries ja adquirides per fer prototips anteriors.

rPi-GPRS - Bateria 1

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Raspberry Pi	3500			
LCD Shield	200			
Lector RFID	1200			
Transformador	120			
	5020	15	75.300	6,3

rPi-GPRS - Bateria 2

Component	Potència (mW)	Autonomia (h)	Potència total (mW x h)	Bateria 12V (A/h)
Mikrotic	2040			
Mòdem USB	2500			
	4540	15	68.100	5,7

Figura 4.18 - Càlcul teòric de l'autonomia del rPi-eth amb dues bateries

La nova configuració amb dues bateries compleix perfectament els requeriments d'autonomia com es pot veure a la Figura 4.18. La disposició final de components per al rPi-GPRS queda definida segons el diagrama de la Figura 4.19.

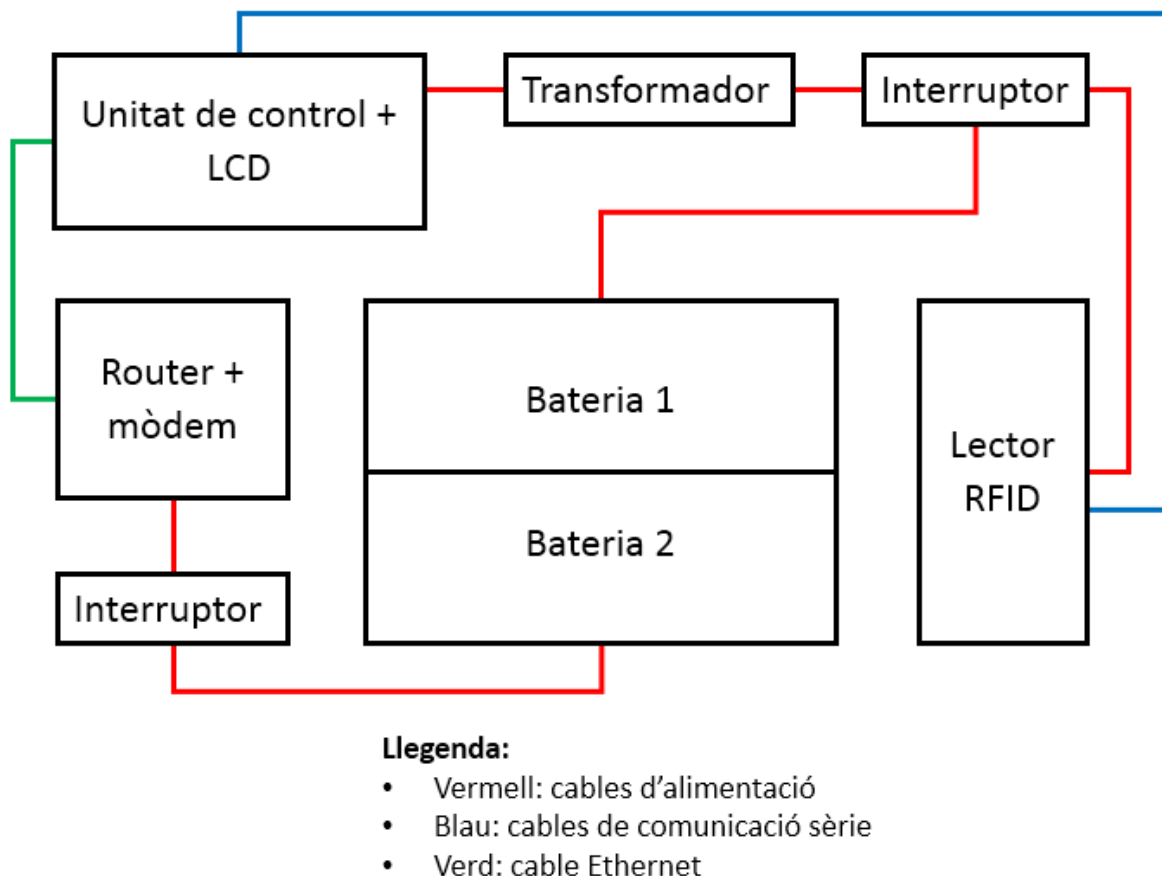


Figura 4.19 - Disposició modular del rPi-GPRS

4.2.8.3. Descripció de les operacions a nivell de programari

Comparat amb el model rPi-eth, des del punt de vista de la unitat de control no hi ha absolutament cap canvi a nivell de programari. La Raspberry Pi interactua amb la xarxa exactament de la mateixa manera, a través d'una interfície Ethernet que li dóna connexió a Internet. Això s'aconsegueix delegant totes les funcionalitats de xarxa al Mikrotik.

La configuració del Mikrotik consisteix en les següents operacions:

- Configurar el port USB de manera que escolti la porta d'enllaç que publica el mòdem USB
- Configurar els dos ports Ethernet de manera que donin DHCP.
- Configurar el port sense fils com a *access point* també amb DHCP. Aquest port sense fils és molt útil per fer verificacions a la unitat de control sense haver d'accedir-hi físicament.

4.3. Funcionalitat d'emmagatzematge

La funcionalitat d'emmagatzematge definida a l'apartat 3.4 que es pot veure a la Figura 4.20 s'ha implementat sobre un servidor remot. Aquest és el servidor que té contractat la organització de cursa per allotjar la seva web. Més que d'un servidor com a tal, es tracta d'un servei de *hosting*. Això és una limitació

important perquè vol dir que la organització no té la possibilitat d'instal·lar-hi altres serveis o modificar la configuració base dels existents.

A continuació es detallen sobre quins serveis s'ha implementat cada una de les funcionalitats:

- **Interfície d'entrada.** Es tracta d'una plana web escrita amb PHP.
- **Contenidor.** El contenidor de dades és una base de dades MySQL que l'organització utilitza des de 2009.
- **Publicació.** El servei emprat per publicar dades és Apache.
- **Càrrega massiva.** És un programa escrit amb Java.

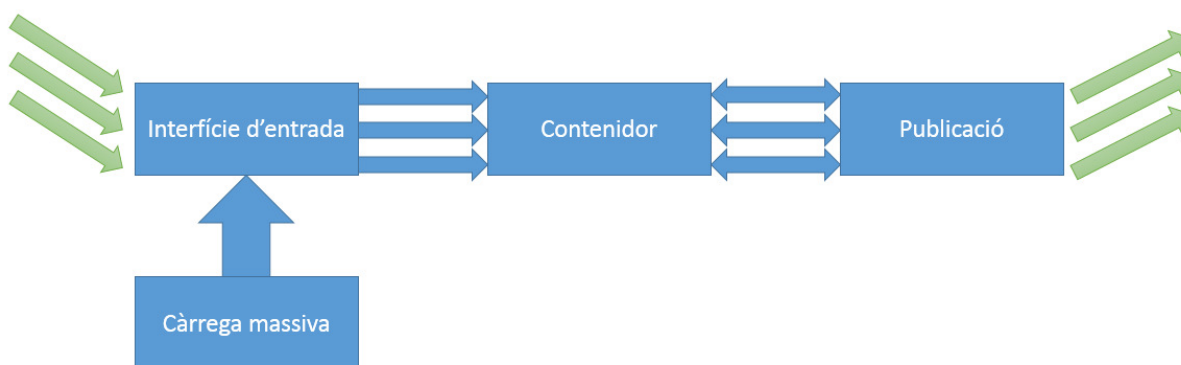


Figura 4.20 - Funcionalitat d'emmagatzematge

4.3.1. Interfície d'entrada

Segons els requeriments, la interfície ha de ser accessible remotament per la funcionalitat de captura. Disposant d'un servei d'Apache publicat a Internet es decideix implementar la interfície amb PHP. Segurament hauria estat més elegant implementar-ho amb un servei web però la impossibilitat de canviar la configuració base del servidor va fer descartar aquesta opció.

La interfície implementa el mètode *GET* mitjançant una *URL* parametritzada. Així mateix, s'hi ha implementat un mecanisme de seguretat per evitar atacs de d'injecció d'SQL. Aquesta interfície rep tres paràmetres: l'identificador de control, el número de dorsal i l'hora de pas. Internament té dues constants per definir l'edició de la cursa i el nom de la taula amb la que s'ha d'interactuar.

Per tal d'alliberar la funcionalitat de captura d'haver de fer control d'errors s'ha implementat aquesta interfície amb un patró *fire and forget* [27]. Es dona aquest nom als patrons d'intercanvi de missatges en els que l'emissor envia un missatge sense esperar cap resposta.

Tot i que en teoria la funcionalitat d'emmagatzematge no hauria d'enviar dades duplicades s'implementa un control de duplicats. Si un participant ja té un temps assignat per a un control, s'ignoren les successives peticions sobre aquest binomi participant, control.

Després d'actualitzar correctament el temps, es comprova si l'usuari acaba d'arribar al darrer punt de control. Si el participant acaba d'arribar a l'arribada s'ha de determinar quin recorregut ha fet i guardar-ho al contenidor. Cal tenir en compte que la funcionalitat de temps real de la funcionalitat de captura és optativa de manera que aquesta operació no està garantida. Ha de ser supervisada pels operadors de l'arribada. Aquesta comprovació és una revisió visual de les marques que hauran fet els operadors dels controls de Puigscalm i Cabrera al dorsal del participant. En cas de reclamació del participant, aquesta es podrà resoldre una vegada el contenidor tingui totes les dades de la cursa.

4.3.2. Càrrega massiva

Hi ha dispositius de control que no tenen funcionalitats de xarxa i els que en tenen no està garantit que hagin pogut enviar totes les dades a temps real. Per tant, de seguida que arriba un dispositiu de control al centre de control de cursa s'ha de recuperar la targeta SD i processar el fitxer de captures. Aquest fet justifica l'existència d'una aplicació de càrrega massiva.

El procés de càrrega massiva és un procés molt senzill escrit amb Java que llegeix tots els registres del fitxer de captura. Per cada registre fa una crida a la interfície d'entrada i, seguint el patró *fire and forget*, passa a processar el següent registre.

4.3.3. Contenedor

El contenidor és una taula MySQL que la organització ja tenia dissenyada per gestionar el cronometratge d'edicions anteriors. Es tracta d'una sola taula amb les següents característiques:

- La clau principal està definida pel binomi número de dorsal i any de l'edició. Això implica que la taula tindrà un registre per cada participant de cada edició.
- Algunes dades personals com ara el nom i cognoms.
- Hi ha 11 camps que serviran per emmagatzemar el temps de pas del participant per cada un dels controls.
- Un camp de text per emmagatzemar el nom del recorregut.

L'alumne no ha participat en el disseny d'aquesta estructura de dades. Simplement s'ha validat que el disseny compleixi els requeriments funcionals de la solució.

4.3.4. Publicació

La fase de publicació dota de visibilitat a les dades del contenidor cap al món exterior. Aquestes dades s'han de publicar de manera que múltiples consultes es puguin servir al mateix temps. Aquesta fase s'implementa sobre el servei Apache. Tant el servei de MySQL que implementa la funcionalitat de contenidor com el servei Apache estan dissenyats tenint en compte la concurrència.

Apache és un servei provinent d'un projecte de codi obert que ja estava essent utilitzat per la organització. L'alumne simplement ha validat que el servei compleixi els requeriments funcionals de la solució.

4.3.5. Validació de la implementació

Tenint en compte que s'utilitzen solucions que proporciona el mercat, aquesta funcionalitat requeria d'una validació funcional. En la fase de desenvolupament, el procés de càrrega massiva va ser utilitzat per validar tota la funcionalitat d'emmagatzematge. Es van preparar dos fitxers d'entrada amb 20.000 registres de dades sintètiques cada un. Es van executar al mateix temps dues instàncies del procés de càrrega massiva amb aquests dos fitxers d'entrada. Aquesta prova persegueix les següents validacions per cada fase:

- **Interfície d'entrada.**
 - Segons els requeriments funcionals la interfície ha de ser capaç de processar múltiples peticions simultànies. El servei Apache no ha de ser un factor limitant en aquest sentit.
- **Contenidor.**
 - Totes les peticions que rep el contenidor s'han d'atendre sense que se'n perdi cap. A més, s'han de processar prou de pressa perquè no representin un coll d'ampolla.
- **Publicació.**
 - A mesura que el contenidor va actualitzant les dades, aquestes han de ser consultables pel mòdul de publicació al mateix temps que es generen. A més, s'han de poder consultar per múltiples clients al mateix temps.

4.4. Funcionalitat de consulta

La funcionalitat de consulta ha de permetre que múltiples usuaris finals tinguin informació actualitzada sobre el desenvolupament de la cursa. Segons el document de requeriments funcionals, la consulta d'informació consta d'una fase estàtica i una fase dinàmica, com es mostra a la figura 4.21.

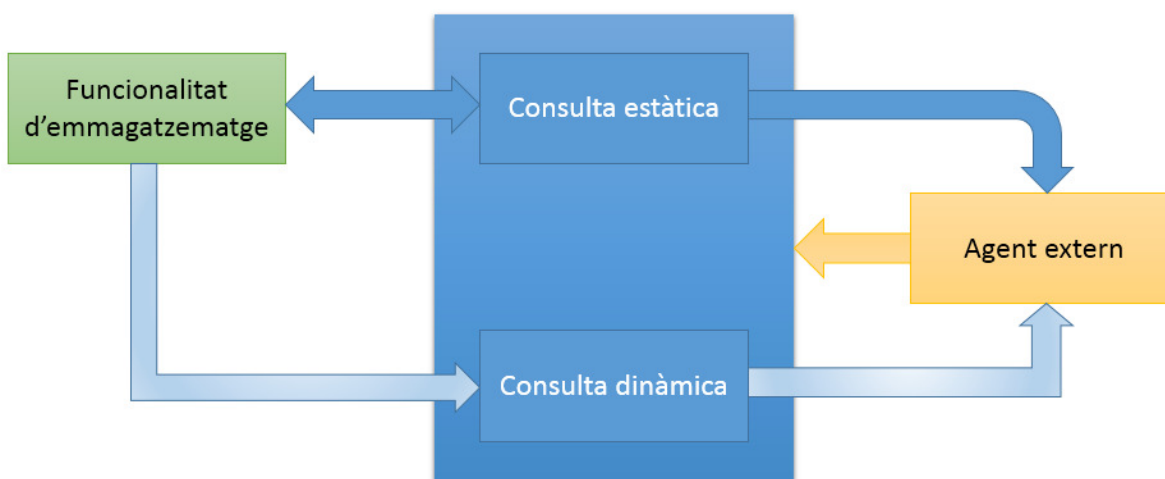


Figura 4.21 - Funcionalitat de consulta

Aquesta funcionalitat s'ha implementat sobre una plana web que està allotjada al mòdul de publicació de la funcionalitat d'emmagatzematge. La fase estàtica de la consulta està implementada amb PHP/HTML i fa les següents operacions:

- Consulta al contenidor les dades emmagatzemades en aquell moment per tots els controls.
- Presentació tabular de les dades.

Després de l'execució de la consulta estàtica l'usuari tindria un representació de l'estat de la cursa en el moment de sol·licitat la informació. Qualsevol esdeveniment futur no apareixeria per pantalla a no ser que l'usuari refresqués les dades.

Per solucionar aquesta limitació s'implementa una consulta dinàmica. Tenint en compte que l'alumne no tenia accés a modificar la configuració del servidor es va optar per utilitzar el servei comercial Pusher [28]. Aquest servei implementa *websockets* per tal d'enllaçar esdeveniments entre usuaris finals o generats pel servidor. Pusher dota d'un canal de comunicacions. Tots els usuaris subscrits a un canal reben els missatges que s'envien al canal.

La fase dinàmica de la consulta està implementada amb Javascript i implementa les següents operacions:

- Subscriure el client al canal de Pusher.
- Configurar una funció d'atenció a l'arribada de nous missatges pel canal.
- Aquesta funció afegeix la informació a la plana web de seguida que s'ha rebut el missatge.

L'origen de les dades de la consulta dinàmica es situen a la interfície d'entrada. En altres paraules, la interfície d'entrada a part de guardar les dades al contenidor envia un missatge al canal de Pusher que serà escoltat i presentat per pantalla a tots els clients de la funcionalitat de consulta.

5. Configuració de la prova de camp

El dia 19 d'abril de 2015 es va realitzar una prova de camp amb les solucions plantejades en aquest projecte. Aquest capítol pretén explicar quina va ser la configuració final de models de dispositius de control i justificar-ne l'elecció.

- **Control 1 – Les Valls.** El primer control és força proper a la sortida i els participants arriben molt junts. per evitar cues es va optar per dotar el control de dos dispositius de control del model ArduinoDLv1. En aquest primer control s'assumeix que els participants no abandonaran i quan arribi el responsable que tanca la cursa el control es podrà tancar. En altres paraules, el recompte de participants no és crític en aquest control.
- **Control 2 – Bellmunt.** Aquí els participants encara van força junts i es va optar per posar-hi dos ArduinoDLv2. Segons el pla de seguretat de la cursa aquí ja és obligatori comprovar el nombre de participants i aquest dispositiu ho permet.
- **Punt d'avituallament – Salgueda.** Històricament aquest punt era un punt d'avituallament i no un punt de control però la organització va voler-hi tenir un ArduinoDLv2 per tal de gestionar amb més seguretat la cursa.
- **Control 3 – Coll de Sant Bartomeu.** Primer punt on els participants poden triar recorregut. En aquest punt de control s'hi posa un ArduinoDLv2.
- **Control 4 – Puigsacalm.** Aquest punt de control es situa en un cim amb certa cobertura de xarxa mòbil. S'hi decideix posar un rPi-GPRS per intentar transmetre les dades en temps real.
- **Control 5 – Prat de la Vola 1.** L'organització dota aquesta localització d'accés a Internet a través d'una interfície Ethernet. El dispositiu a tal efecte és el rPi-eth.
- **Control 6 – Cabrera.** A l'igual que al Puigsacalm aquest punt de control està situat en un punt elevat amb certa possibilitat d'accés a la xarxa de telefonia mòbil. S'hi posa un rPi-GPRS.
- **Control 7 – Prat de la Vola 2.** Els dos controls del Prat de la Vola estan separats pocs metres entre ells, per tant en aquest control també s'hi posa un rPi-eth.
- **Control 8 – Collsaplana.** Aquest control està situat en una petita vall envoltada d'arbres amb poca possibilitat de tenir accés a la xarxa de telefonia mòbil. S'hi posa un ArduinoDLv2.
- **Control 9 – Sant Pere.** El control de Sant Pere està a prop de la població amb el mateix nom. Amb aquesta proximitat a la població es decideix posar-hi un rPi-GPRS per intentar enviar dades a temps real.
- **Control 10 – Arribada.** A l'arribada hi ha el centre de control de la cursa i s'hi disposa de xarxa Ethernet. Per tant el dispositiu emprat és el rPi-eth.

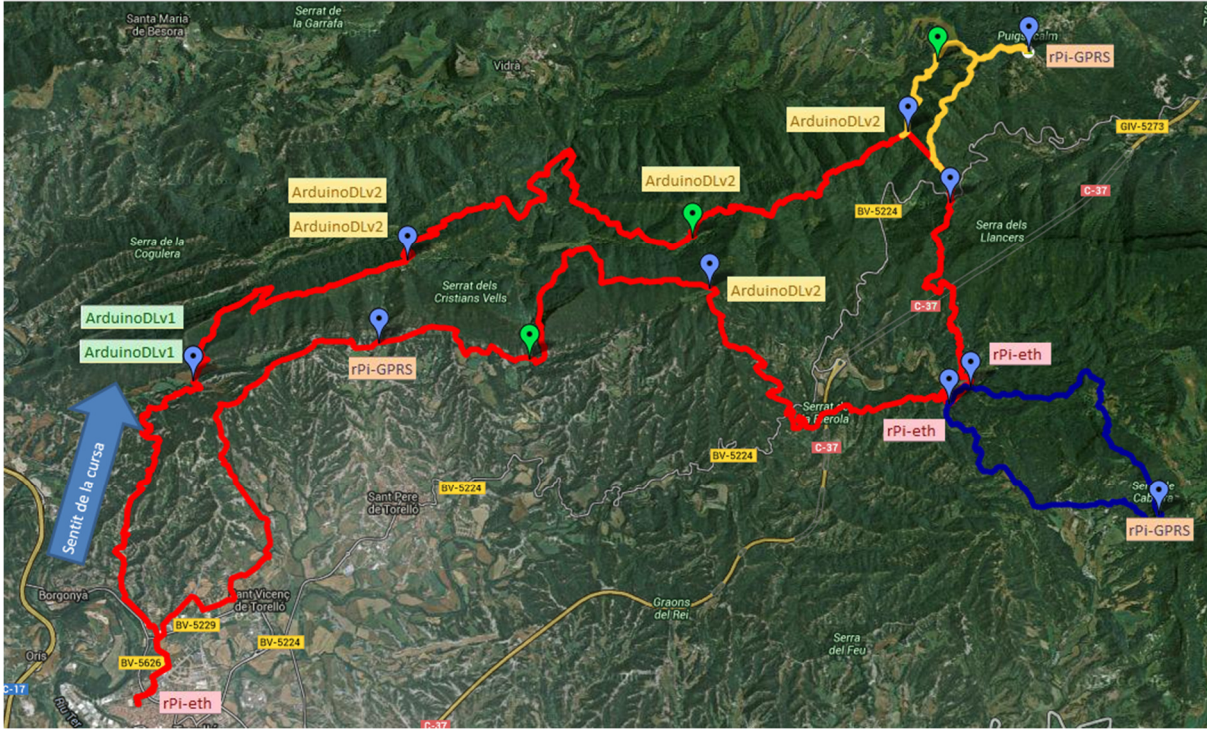


Figura 5.1 - Mapa de distribució de dispositius de control

6. Viabilitat tècnica

El sistema de cronometratge que presenta aquesta documentació es va fer servir com a única solució de cronometratge per la cursa Pels Camins dels Matxos del 14 d'abril de 2015. Per tant, es pot afirmar que globalment la solució és viable des del punt de vista tècnic.

Tot i això si s'aprofundeix en l'anàlisi hi ha alguns punts forts i alguns punts que tenen recorregut de millora o que simplement no són viables.

6.1. Viabilitat tècnica dels dispositius de control

A continuació es passa a detallar la viabilitat tècnica de cada un dels diferents models de dispositiu de control:

- **ArduinoDLv1.** Es tracta d'un dispositiu que compleix perfectament els requeriments funcionals. Comparat amb els altres models, es tracta d'una solució senzilla que implementa la funcionalitat de *datalogger* i a les proves de camp ha complert perfectament els objectius. Com a punt dèbil té la seva limitada funcionalitat. Com a punts forts té la seva fiabilitat i l'alta autonomia que clarament excedeix el requeriment de 15 hores.
- **ArduinoRT.** Aquest dispositiu no va complir els requeriments funcionals a les proves de camp. La comunicació i2c entre els dos Arduinos es va fer amb un cable no apantallat; aquest fet provocava falses comunicacions al induir-se corrent probablement generat per l'antena del lector RFID o el mòdem GPRS. A més, segurament debut a l'alt consum de corrent del mòdem GPRS, a un dels prototips fins i tot se li va fer malbé la pista d'alimentació principal com es pot observar a la Figura 6.1. A aquests problemes de maquinari se l'hi han d'afegir les limitacions de memòria que fan impossible un control d'errors robust. Per aquestes raons es va decidir abandonar la via de desenvolupament oberta per aquest model.
- **ArduinoDLv2.** Aquesta és una evolució del ArduinoDLv1 en la que s'hi afegeix una pantalla per donar informació al operador. Millora la funcionalitat del dispositiu sense penalitzar-ne l'autonomia que continua estant per sobre dels requeriments. Aquest model ja pràcticament consumeix tota la capacitat de memòria de l'Arduino de manera que serà difícil dotar-lo de més funcionalitats; aquest fet es considera un punt feble. Com a punt fort té la fiabilitat ja que a les proves de camp ha donat uns resultats excel·lents.
- **rPi-eth.** Es podria dir que aquest és el primer model de la nova generació. Està basat en Raspberry Pi fet que implica un gran salt quantitatiu en capacitat de càlcul i memòria. Aquest model compleix perfectament tots els requeriments funcionals. Com a punt fort té la gran capacitat encara disponible que permetria implementar-hi moltes més funcionalitats. Com a punt més dèbil té el consum que, tot i complir el requeriment, està a prop del límit. La dependència d'una xarxa Ethernet es considera un factor limitant pel seu desplegament.

- **rPi-GPRS.** Es pot considerar el successor del ArduinoRT. Es tracta d'un disseny molt més compacte al només utilitzar una Raspberry Pi en contrast amb els dos Arduinos. El fet de que a nivell de unitat de control sigui idèntic a un rPi-GPRS es considera un punt fort ja que simplifica la programació i gestió dels dispositius. Aquest fet ve donat pel seu disseny modular en el que la connectivitat sempre es rep per la interfície Ethernet. En aquest model en particular son el *router* i el mòdem USB els encarregats de dotar de connectivitat a la unitat de control. Com a punts dèbils té la dependència d'una infraestructura controlada per tercers com és la xarxa de telefonia mòbil i la necessitat d'una segona bateria n'augmenta el pes final per tal de complir els requeriments d'autonomia.



Figura 6.1 - Pista cremada d'un ArduinoRT

6.2. Viabilitat tècnica del web en temps real

Tot i que la implementació de les funcionalitats que conformen el web en temps real compleixen els requeriments funcionals, a continuació se'n destaquen els seus respectius punts forts i dèbils.

- **Servidor.** El fet d'implementar solucions sobre un servidor de *hosting* estàndard té l'avantatge de no haver de fer-hi operacions de manteniment o tenir cura del maquinari. En contra, no s'hi poden instal·lar nous serveis o canviar el comportament dels ja instal·lats. Aquest fet té una clara limitació a l'hora d'implementar solucions que han de ser compatibles amb la configuració del servidor.
- **Contenidor.** L'estructura de dades que l'organització té definida per gestionar el cronometratge de la cursa és una sola taula. Això és clarament un avantatge a l'hora de fer consultes per què simplifica molt les consultes. En contra, s'augmenta sensiblement el temps necessari per actualitzar temps de pas de participants. En termes de rendiment, una estructura de dades en la que noves captures de dades s'insereixin en comptes d'actualitzar-se seria més ràpida.

- **Pusher.** Utilitzar un servei extern com ara Pusher té l'avantatge de deixar la gestió de la infraestructura a un tercer. Pusher és un servei comercial que ofereix una versió gratuïta amb uns certs límits de clients i missatges transmesos. Analitzar prèviament si la versió gratuïta serà suficient pot ser complicat abans de posar una solució en producció. Tenint en compte els límits imposats pel proveïdor a la versió gratuïta, el servei funciona sense cap mena de problema.

7. Valoració econòmica

7.1. Costs del projecte

Els costs d'aquest projecte estan centrats en l'adquisició dels components que componen els dispositius de control. Aquests costs no inclouen les eines bàsiques de treball amb components electrònics (tester, soldador, joc de tornavisos...) ni els consumibles com ara cable elèctric, terminals o fil d'estany.

ArduinoDLv1		rPi-eth	
Component	Preu (€)	Component	Preu (€)
Bateria	13,00 €	Bateria	13,00 €
Arduino	24,00 €	Raspberry Pi	38,00 €
RTC SD Shield	18,00 €	LCD Shield	23,00 €
Targeta SD	6,00 €	Lector RFID	17,00 €
Lector RFID	17,00 €	Transformador 12V-5V	24,00 €
Caixa estanca petita	12,00 €	Caixa estanca gran	19,00 €
TOTAL	90,00 €	TOTAL	115,00 €

ArduinoRT		rPi-GPRS	
Component	Preu (€)	Component	Preu (€)
Bateria	13,00 €	Bateria (2x)	26,00 €
Arduino	24,00 €	Raspberry Pi	38,00 €
RTC SD Shield	18,00 €	Protector metracrilat	18,00 €
Targeta SD	6,00 €	LCD Shield	23,00 €
Lector RFID	17,00 €	Lector RFID	17,00 €
Arduino	24,00 €	Transformador 12V-5V	24,00 €
GSM Shield	84,00 €	Mikrotik	46,00 €
SIM de dades	15,00 €	Mòdem USB	25,00 €
Caixa estanca petita	12,00 €	SIM de dades	15,00 €
TOTAL	213,00 €	Caixa estanca gran	19,00 €
		TOTAL	251,00 €

ArduinoDLv2	
Component	Preu (€)
Bateria	13,00 €
Arduino	24,00 €
RTC SD Shield	18,00 €
Targeta SD	6,00 €
Lector RFID	17,00 €
LCD Shield	23,00 €
Caixa estanca petita	12,00 €
TOTAL	101,00 €

Figura 7.1- Preus dels components dels dispositius de control

A la Figura 7.1 es pot veure l'escandall de costos per cada model de dispositiu de control. S'ha de tenir en compte que els preus són aproximats perquè el preu de molts d'aquests components oscil·la habitualment i perquè el preu del transport és variable segons la quantitat de components comprats en una sola comanda.

Considerant que aquest maquinari es fa servir 15 hores a l'any, es pot dir que no requereix en absolut de manteniment i es poden obviar els costos relacionats amb el cicle de vida dels components.

Si s'extrapolen els preus de cada dispositiu de control amb la configuració de la cursa descrita al Capítol 5, s'obté el preu total dels dispositius de control de la Figura 7.2. Tot i que hi ha un límit de 600 participants l'organització va decidir comprar 650 xips RFID.

Dispositiu	Quantitat	Cost
ArduinoDLv1	2	180,00 €
ArduinoDLv2	5	505,00 €
rPi-eth	3	345,00 €
rPi-GPRS	3	753,00 €
Xips RFID	650	650,00 €
TOTAL		2.433,00 €

Figura 7.2 - Cost total de la configuració emprada a la prova de camp

Els costos globals del sistema de cronometratge podrien també considerar el cost anual del servei de *hosting*. S'ha decidit desestimar aquest cost tenint en compte que no és directament causat pel sistema de cronometratge; aquest cost ja existia prèviament per donar suport al lloc web de la cursa.

7.2. Viabilitat econòmica

Per tal de veure la viabilitat econòmica del projecte s'ha de comparar amb altres serveis similars que ofereix el mercat. S'han seleccionat solucions dels tres proveïdors ja presentats al capítol 2.2. A la Figura 7.3 es poden veure els preus d'aquests serveis.

Proveïdor	Preu
Smartchrono	390,00 €
Temps x curses	1.500,00 €
Livetrail	3.500,00 €

Figura 7.3 - Preus de serveis de cronometratge

En el cas de Smartchrono es tracta d'un preu tancat que està disponible al seu lloc web. En el cas dels altres dos proveïdors s'ha demanat un pressupost aproximat que dona una cota de preu però en cap cas un preu final real. En el cas de Livetrail s'ha demanat un pressupost aproximat d'una configuració similar a la d'aquest projecte i s'utilitzarà aquest preu per fer comparacions.

Amb les dades de la Figura 7.2, tenint en compte que els xips només s'han de comprar el primer any i assumint que es vol amortitzar el sistema amb 5 anys, es pot dir que el cost anual del sistema és de $2.433,00 / 5 =$ **487,00€**.

A nivell purament numèric aquest projecte és una bona inversió per l'organització de la cursa Pels Camins dels Matxos. S'ha de tenir en compte, però, que aquesta anàlisi de viabilitat està comparant serveis comercials amb una acció de voluntariat sense cap mena de retribució ni costos laborals.

8. Conclusions

8.1. Assoliment d'objectius

Al capítol 1.3 es defineixen una sèrie d'objectius que pretén aconseguir assolir aquest projecte. A continuació se'n detalla el grau d'assoliment de cada un d'ells.

- **Identificació inequívoca dels participants.** La tecnologia RFID emprada per solucionar aquest problema ha resultat ser una bona elecció. Ni en la fase de desenvolupament ni en les proves de camp s'ha donat cap cas de lectures errònies.
- **Dispositius de control autònoms i mòbils.** Els diferents models de dispositius de control han complert les especificacions d'autonomia segons l'anàlisi teòric del capítol 4. A nivell pràctic, durant la prova de camp tots els dispositius van arribar al centre de control de cursa amb bateria. Pel que fa a la mobilitat, el volum i el pes dels dispositius compleixen el requeriment de ser transportables en una motxilla. En les proves de camp s'ha verificat que es poden transportar amb una motxilla d'uns 20 litres de capacitat.
- **Dispositius propietat de l'organització i reutilitzables.** A través d'un acord amb la organització de la cursa es decideix que aquesta finançaria el cost dels components. A canvi, l'alumne es compromet a fer-ne el manteniment i configuració per futures edicions. Els dispositius s'han dissenyat de manera que els costos de manteniment siguin mínims. Al mateix temps, la configuració per a properes edicions es limita a editar un fitxer de configuració a cada dispositiu.
- **Visualització del recompte de participants.** Seguint indicacions de la organització, tots els dispositius de control emplaçats a partir del segon punt de control tenen la funcionalitat de recompte de participants. El recompte no va presentar cap mena d'incidència a les proves de camp i va complir la seva funcionalitat.
- **Sistema de persistència de dades.** Les dues solucions redundants han estat implementades i han funcionat correctament. Pel que fa a la solució sobre fitxer de text tots els dispositius han escrit totes les lectures que s'han realitzat i no se n'ha perdut cap. Des del punt de vista del contenidor, totes les dades generades han estat emmagatzemades correctament ja sigui quan provenien de la interfície d'entrada o del procés de càrrega massiva.
- **Accés a Internet per a alguns dels dispositius.** Aquest és un doble objectiu depenent del canal de comunicacions emprat. L'objectiu primari és transmetre totes les dades capturades a través de la xarxa Ethernet i el secundari és transmetre tantes dades com es pugui a utilitzant la xarxa de telefonia mòbil. Ambdós objectius s'han complert. A la prova de camp tots els dispositius de captura que tenien connectivitat Ethernet van transmetre el 100% de les dades que van capturar. S'ha de tenir en compte que la qualitat de senyal xarxa de telefonia mòbil és fora de l'abast del projecte i l'objectiu secundari s'ha complert quan la xarxa qualitat de senyal era suficient. Les proves en entorns urbans

amb bona qualitat de senyal donen un 100% d'efectivitat. Els resultats finals a la prova de camp són els següents:

- Puigsacalm: 253 captures transmises de 338 → 75%
 - Cabrera: 146 captures transmises de 275 → 53%
 - Sant Pere: 22 captures transmises de 476 → 5%
- **El sistema ha de ser capaç de determinar quin recorregut ha efectuat cada participant.** Els dos controls implicats en la selecció de recorregut són els de Puigsacalm i el del Prat de la Vola 2. Aquest últim va transmetre el 100% de les dades mentre que el del Puigsacalm en va transmetre un 75%. Així doncs, el sistema ha estat capaç de determinar correctament el recorregut del 75% dels participants.
 - **Les dades en temps real han de ser visibles des d'una plana web.** La presentació a temps real de les dades a mesura que es generaven va funcionar correctament amb les limitacions descrites a l'apartat 8.2.

8.2. Incidències

La prova de camp de no va estar lliure d'incidències. Aquestes es passen a detallar a continuació. La categorització d'incidències segueix la següent escala: crítica, greu o lleu.

- **Error al rellotge d'un ArduinoDLv1.** Aquesta incidència va ser causada per un error de programació de l'alumne en un dels dos ArduinoDLv1 del primer punt de control. Per assegurar el correcte funcionament dels rellotges RTC el dia abans es van canviar totes les piles dels dispositius basats en Arduino. A continuació calia assignar l'hora correcta a través del codi i posteriorment comentar aquesta línia per tal de que el dispositiu funcionés normalment. Aquesta línia de codi no es va comentar fet que va provocar que totes les dades capturades per aquest dispositiu tinguessin la mateixa hora assignada i, per tant, no fossin útils per a la organització. L'impacte d'aquesta incidència es categoritza com a crítica i és la pèrdua de dades de pas d'aproximadament la meitat de participants al primer punt de control.
- **Problemes de bloqueig a la base de dades.** Durant el transcurs de la prova de camp es va detectar a l'arribada que algunes captures de dades tardaven molt temps a estar disponibles a la base de dades. Després d'una investigació de la causa de l'error a la prova de camp mateix es va detectar que eren els formularis que s'havia fet la organització amb Microsoft Access per consultar i corregir el recorregut dels participants. Aquests formularis bloquejaven la taula de la base de dades amb exclusivitat causant així *deadlocks*. Com a acció correctiva l'alumne va parar immediatament el Microsoft Access i preparar unes sentències SQL que feien les mateixes funcionalitats sense causar *deadlocks*. Aquesta incidència es va categoritzar com a lleu perquè no es van perdre dades, només arribaven amb retard.

- **Caiguda d'un dispositiu de control.** El dispositiu de control del Prat de la Vola 2 estava situat a sobre d'una taula i per una acció involuntària d'un operador va acabar caient a terra. El resultat de la caiguda va ser que la bateria es desenganxés de la caixa provocant així una estrebada. L'estrebada va tenir prou força per arrencar la soldadura dels cables que connecten el lector RFID amb la Raspberri Pi. Per sort, l'alumne era allà i va poder tornar soldar els tres cables arrencats amb uns minuts. L'impacte van ser 7 captures que es van haver d'anotar i més tard processar-les manualment a la base de dades. Aquesta incidència hauria estat crítica de no haver pogut ser reparada ràpidament però finalment va acabar essent lleu perquè no va haver-hi pèrdua de dades.
- **Servei gratuït de Pusher.** La versió gratuïta d'aquest servei, té dues limitacions: un màxim de 20 clients simultanis i un màxim de 100.000 missatges al dia. En la fase de planificació es van considerar aquestes limitacions com a suficients. Com es pot comprovar a la Figura 8.1 entre les 6:00 i les 18:00 del dia de la cursa es van superar el límit de connexions simultànies. Els usuaris que estaven per sobre el llindar no rebien les actualitzacions automàticament. Aquest fet no afectava la funcionalitat de càrrega a temps diferit. No es van prendre mesures correctives perquè aquesta situació es va detectar aproximadament a les 20h quan la càrrega d'usuaris ja tornava a estar per sota el llindar. Aquesta incidència es categoritza com a greu perquè va afectar a una funcionalitat important de la solució.

8.3. Proposta de millores

La següent llista de proposta de millores està elaborada des dels següents punts de vista:

1. Acabar de complir els objectius que s'han complert parcialment.
 2. Evitar o mitigar les incidències que ja s'han produït.
 3. Implementar millores que puguin evitar futures incidències o dotar al sistema de noves funcionalitats.
 4. Suggestions de l'organització, els operadors i els usuaris.
- **Auto diagnòstic.** Els models basats en Raspberry Pi tenen una senzill eina d'auto-diagnòstic que consisteix en missatges per pantalla que informen l'usuari a mesura que el procés d'inicialització va avançant. Aquesta funcionalitat no està implementada als dispositius basats en Arduino i podria evitar la incidència del rellotge RTC, entre d'altres.
 - **Robustesa.** Considerant que els dispositius de control es mouen i treballen en entorns on els impactes són probables haurien de ser més robusts. Això es tradueix en elements amortidors i fixacions més segures pels diferents components a la caixa estanca.
 - **Homogeneïtzació de dispositius.** A mig termini seria bo minimitzar el nombre de models de dispositius de control. Amb això es faria el desenvolupament de noves funcionalitats i el

manteniment més lleuger. Els dispositius basats en Raspberry Pi són els candidats a ser la plataforma per futures evolucions.

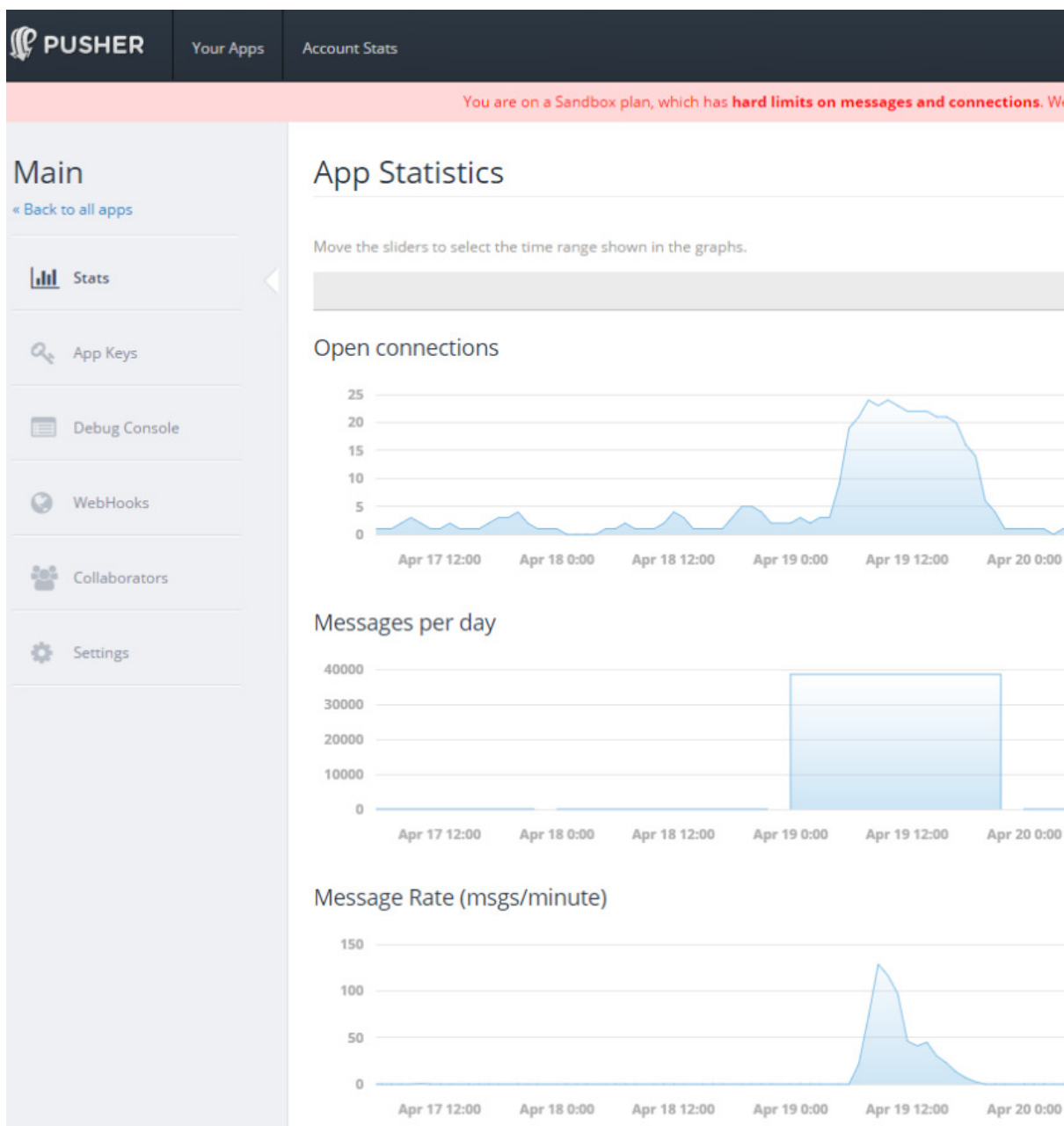


Figura 8.1 - Estadístiques del servei Pusher del dia de la prova de camp

- **M2M.** Els operadors de telefonia mòbil tenen al mercat targetes SIM orientades a serveis M2M (*Machine to Machine*[29]). Aquestes targetes SIM tenen la peculiaritat de poder-se enllaçar amb qualsevol operador que tingui senyal en aquell lloc concret. Aquest fet potser eliminaria la variable proveïdor de la problemàtica de les comunicacions en entorns remots. És clarament una opció a explorar des del punt de vista tècnic i econòmic.

- **Antenes GPRS externes.** El mercat ofereix models de mòdem USB que tenen la possibilitat de connectar-hi antenes externes com ara el Huawei E3131 [30]. Aquesta millora en conjunció amb la del punt anterior hauria d'ajudar a aconseguir unes taxes de tramesa de dades a temps real superior.
- **Senyalització.** Tant els operadors com els participants van reportar no tenir clar si havien completat l'operació de captura correctament. Quan es completa una lectura el lector RFID emet un so però no és suficientment alt i en alguns emplaçaments ni se sent. A més, es mostra un missatge per pantalla. Però tot això no és suficient; s'hauria d'afegir un senyalitzador acústic molt més potent i algun dispositiu visual molt evident com ara una cinta de *leds* que s'il·luminin per indicar el final correcte del procés.
- **Servidor.** Dependre d'un servidor de *hosting* pre-configurat és un factor limitant important, per exemple, per implementar solucions a temps real. La idea consisteix en construir una infraestructura que s'utilitzi única i exclusivament per la solució de cronometratge. Aquesta solució hauria d'estar acompanyada de *APIs* i/o serveis web que permetin integrar-la fàcilment a una plana web. S'hauria de fer un estudi de mercat per veure opcions disponibles.
- **Gestió de la cursa.** La solució de cronometratge hauria de tenir funcionalitats que permetessin fer una gestió integral de la cursa: calcular nombre de persones entre punts de control, gestió d'abandonaments, edició de dades de participants, impressió de dorsals... Aquesta millora evitaria la interferència d'aplicacions externes que poden afectar al rendiment de la solució de cronometratge.
- **Gestió de no enviats.** Els models basats en Raspberry Pi tenen un sistema per emmagatzemar les captures que no s'han pogut transmetre a temps real. Quan es torna a detectar xarxa s'envia l'última captura realitzada i una de les pendents. Aquesta llista de pendents d'enviar està feta sobre un fitxer de text i, tot i funcionar correctament, podria acabar originant problemes. Una taula en un sistema de gestió de bases de dades lleuger podria ser una solució tècnica viable. Enviar només una de les captures pendents és una estratègia molt conservadora que pot derivar en moltes ratxes de connexió que no s'utilitzen. Quan es detecta connectivitat, s'haurien d'enviar tantes captures no enviades com es pugui.
- **Abandonaments.** Quan els participants abandonen informen al punt de control i aquests transmeten la informació pels seus mitjans cap al centre de control. El centre de control és el que actualitza la base de dades. Dotar als dispositius de control de la funcionalitat de registrar abandonaments seria una bona millora funcional.
- **Estimació de temps de pas.** Tenint les sèries històriques d'edicions anteriors no hauria de ser difícil estimar el temps de pas dels participants per cada control. Una vegada un participant passi per cada punt de control es podria anar refinant el càlcul. L'error de l'estimació al primer control serà molt alt

però anirà baixant a mesura que es passin controls. Aquestes dades s'haurien de presentar en una plana web.

8.4. Autoavaluació

Aquesta documentació és fruit del treball d'aproximadament dos anys. Si hagués de descriure l'objectiu principal amb poques paraules seria el següent: implementar una solució viable de cronometratge per la cursa Pels Camins dels Matxos. Aquest objectiu principal el vaig poder complir el passat 19 d'abril de 2015 fent servir única i exclusivament el sistema Matxoclock per cronometrar la cursa.

Globalment estic molt satisfet de com va anar la prova de camp i és un orgull que la solució fos funcional. Al mateix temps, em va saber molt greu cometre l'errada que va causar la pèrdua de dades de la meitat de participants al primer control. Continuant amb els temes per millorar, hi afegiria la tramesa de captures en temps real. Tot i dependre de la xarxa d'uns tercers, aquesta és una funcionalitat que em fa especial il·lusió i m'agradaria arribar a prop del 100% de dades trameses a tots els punts de control.

Penso que el disseny funcional és raonablement bo i dóna possibilitats a seguir diversos models d'implementació. La fase de desenvolupament, tot i no estar exempta d'aproximacions errònies i passes en darrera, va ser força plàcida. Pel que fa a la implementació en sí, un dels factors limitants ha estat haver de treballar amb el servidor de *hosting*; de ben segur que aquest fet canviarà de cara al futur. Crec que la qualitat final dels dispositius de control ha acabat essent prou decent; tot i això, encara tenen massa aparença de prototip. La meva manca d'experiència en disseny de solucions de maquinari es fa evident en el resultat final.

A nivell personal ha estat molt enriquidor haver de prendre tots els rols que eren necessaris per realitzar aquest projecte. A vegades m'ha tocat fer de comercial per vendre confiança a l'organització de la cursa. He hagut de prendre el rol de gestor de projectes durant tota la vida del projecte. Sovint ha convingut agafar el rol de consultor funcional per conceptualitzar les idees, baixar a nivell de programador per fer-les una realitat i, fins i tot, carregar els dispositius en una motxilla per anar a provar-los. Analitzar i aprendre dels errors i encerts comesos en cada un d'aquests rols és el que realment m'emporto d'aquest projecte.

9. Glossari

Arduino. Plataforma de prototipatge de solucions de maquinari de baixes prestacions i baix cost.

Apache. Servidor web HTTP de codi obert.

Captura. Procés pel qual s'identifica un participant i se li assigna un temps en un punt de control determinat.

Dispositiu de control. Implementació en maquinari de la funcionalitat de captura.

GPRS. De l'anglès *General Packet Radio Service*, protocol de comunicacions utilitzat en les xarxes de telefonia mòbil 2G i 3G.

Hosting. Servei orientat a la publicació de llocs web. Els serveis habituals inclouen Apache i MySQL.

MySQL. Sistema gestor de bases de dades de codi obert.

Punt de control. Localització de l'emplaçament dels dispositius de control.

Pusher. Servei que dota de serveis de comunicació asíncrona als llocs web mitjançant websockets.

Raspberry Pi. Plataforma de prototipatge de solucions de maquinari d'altres prestacions i baix cost.

RFID. Identificació per ràdio freqüència. El sistema consta d'un lector actiu que excita un contenidor passiu per recuperar la informació continguda al contenidor.

SD. De l'anglès *Secure Digital*. És un format de targeta de memòria àmpliament utilitzat en electrònica de consum. Els models actuals arriben als centenars de Gigabytes.

Shield. Mòdul d'expansió per a la plataforma Arduino o Raspberry Pi.

Websockets. Tecnologia que proporciona un canal de comunicació bidireccional sobre un sol socket TCP.

10. Bibliografia

10.1. Referències

- [1] Pels camins dels matxos. <http://www.matxos.cat>
- [2] Torelló a la Vikipèdia. <https://ca.wikipedia.org/wiki/Torell%C3%B3>
- [3] Federació d'entitats excursionistes de Catalunya. <https://www.feec.cat/>
- [4] Arduino. <http://www.arduino.cc/>
- [5] Datalogger a la Vikipèdia. https://en.wikipedia.org/wiki/Data_logger
- [6] Raspberry Pi. <https://www.raspberrypi.org/>
- [7] *Agile* a la Wikipèdia. https://en.wikipedia.org/wiki/Agile_software_development
- [8] Pusher. <https://pusher.com/>
- [9] Mikrotik. <http://www.mikrotik.com/>
- [10] Tor des geants. <http://www.tordesgeants.it/>
- [11] Arduino Uno R3. <http://www.arduino.cc/en/Main/ArduinoBoardUno>
- [12] Raspberri Pi B+. <https://www.raspberrypi.org/products/model-b-plus/>
- [13] Beaglebone Black. <http://beagleboard.org/BLACK>
- [14] RFID a la Wikipèdia. https://en.wikipedia.org/wiki/Radio-frequency_identification
- [15] Adafruit RTC + SD shield for Arduino. <http://www.adafruit.com/products/1141>
- [16] AERF. <http://ca.aerf.eu/>
- [17] Comparativa bateries de plom i liti. http://www.altenergymag.com/content.php?post_type=1884
- [18] Bateries de plom de EnergiVM. <http://www.energivm.com/wp-content/uploads/2015/04/CATALOGO-ENERGIVM-PLOMO-MUNDILEC-S.L..pdf>
- [19] Arduino GSM Shield. <http://www.arduino.cc/en/Main/ArduinoGSMShield>
- [20] Documentació sobre stack overflow en els Arduinos. <https://learn.adafruit.com/memories-of-an-arduino/optimizing-sram>
- [21] Adafruit Arduino LCD Shield. <http://www.adafruit.com/products/772>
- [22] Network Time Protocol (NTP) a la Wikipèdia. https://en.wikipedia.org/wiki/Network_Time_Protocol

[23] Adafruit Raspberry Pi LCD Shield. <https://learn.adafruit.com/downloads/pdf/adafruit-16x2-character-lcd-plus-keypad-for-raspberry-pi.pdf>

[24] Himel. <http://www.himel.com/>

[25] Raspbian. <https://www.raspbian.org/>

[26] Huawei E303. <http://consumer.huawei.com/en/mobile-broadband/dongles/features/e303-en.htm>

[27] Fire and forget. http://docs.oracle.com/cd/E17904_01/doc.1111/e17363/chapter05.htm#FPCON248

[28] Pusher. <https://pusher.com/>

[29] Machine to Machine (M2M) a la Wikipèida. https://en.wikipedia.org/wiki/Machine_to_machine

[30] Huawei E3131. <http://consumer.huawei.com/en/mobile-broadband/dongles/tech-specs/e3131-en.htm>

10.2. Comunitats

Durant el desenvolupament del projecte s'han utilitzat documentacions i comunitats d'usuaris de les diferents plataformes de desenvolupament. A continuació s'enumeren les més representatives:

- Arduino. <http://forum.arduino.cc/>
- Raspberry Pi. <https://www.raspberrypi.org/forums/>
- Pusher. <https://pusher.com/docs>
- Python. <https://docs.python.org/2/>
- Javascript. <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>