

Disseny i implementació d'una aplicació per a mòbils per a la lectura de comptadors d'aigua

LECTCOMP



Alumne: Josep M. Vidal Vidal

Dirigit per Roman Roset Mayals

Projecte final de carrera

Enginyeria d'informàtica

Curs 2012-13, 1r semestre

AGRAÏMENTS

A la Vanessa, la Rut i la Bet, per la paciència i la comprensió en les hores que he dedicat per dur a bon port aquest projecte.

Al Carles Gispert i a la Mireia Ciuraneta per la seva inestimable col·laboració en la correcció de la memòria i per fer-hi la crítica oportuna d'aquell que desconeix la matèria.

A l'Olga Martorell i a l'Elisabet Cros, com a responsables de la gestió de l'aigua de l'Ajuntament de Riudoms per ajudar-me a enfocar, definir i valorar l'aplicatiu desenvolupat.

Índex	Pàgina
1. Context	7
2. Paraules clau	8
3. Descripció del projecte	8
4. Objectiu del projecte	9
Resultats esperats	9
5. Organització	9
Activitats del projecte	9
PAC1	10
PAC2	10
PAC3	10
Lliurament final	10
6. Calendari	11
Calendari general del projecte	11
Calendari detallat del projecte	11
Calendari de la PAC1	12
Calendari de la PAC2	12
Calendari de la PAC3	12
Calendari del lliurament final	12
Fites principals del projecte	12
7. Requeriments de material	13
Requeriments de maquinari	13
Requeriments de programari	13
8. Estudi de costos	15
9. Entorn de treball	14
XAMPP	14
Sencha Touch 2	14
Google Chrome v22 o Safari 5.1	15
7Github	15
Aptana3	16
10. Arquitectura del sistema	16
11. Disseny i anàlisi	17
Estructura dels "Models-Store" i fitxers .json	17
Estructura del controlador	17
Estructura de la vista Info	20
Estructura de la vista Carregar	20
Estructura de la vista Lectures	23
Estructura de la vista Enviar	27
12. Implementació	29
Info	29
Carregar	30

Lectures	31
Enviar	35
Optimització de la visualització de la llista de carrers	35
13. Joc de proves	37
14. Creació dels fitxers .json	38
15. Portafoli cronològic del desenvolupament del projecte	42
Estudi i desenvolupament d'un servidor web amb node.js	42
Estudi i desenvolupament de l'aplicació per a dispositius mòbils	43
Lliurament de la PAC 2	43
Lliurament de la PAC 3	44
Lliurament final	44
16. Conclusions	45
17. Línies de futur	45
18. Bibliografia	45
19. Annex 1 – Servidor node.js	47

Il·lustracions	Pàgina
Il·lustració1: calendari general del projecte.	11
Il·lustració2: calendari del projecte.	11
Il·lustració3: calendari de la PAC1.	12
Il·lustració4: calendari de la PAC2.	12
Il·lustració5: calendari de la PAC3.	12
Il·lustració6: calendari de la PAC4.	12
Il·lustració7: panell de control de XAMPP.	14
Il·lustració8: entorn de treball de Sencha Touch 2 Architect.	15
Il·lustració9: diagrama del Model-Vista-Controlador.	16
Il·lustració10: pantalla inicial de l'aplicació.	20
Il·lustració11: diagrama de flux de la carrega dels fitxers json.	21
Il·lustració12: pantalla per carregar els fitxers json.	22
Il·lustració13: missatge d'error de la càrrega de les dades inicials.	22
Il·lustració14: diagrama de flux per introduir i validar lectures.	23
Il·lustració15: pantalles per seleccionar els comptadors per número de carrer o adreça, o afegir-ne un de nou.	24
Il·lustració16: pantalla que ens mostra la selecció de comptadors feta.	25
Il·lustració17: diagrama de flux d'enviar lectures.	27
Il·lustració18: missatge demanant conformitat per iniciar el procés d'enviar el fitxer.	28
Il·lustració19: missatge d'error del procés d'enviar les dades.	28
Il·lustració20: pantalla inicial de l'aplicació en funció de si entra correctament o si falla la connexió amb els fitxers.	29
Il·lustració21: pantalla per carregar els fitxers json, esborrar les lectures i consultar les lectures introduïdes.	30
Il·lustració22: missatge informant que el codi de seguretat és incorrecte o que no s'ha introduït.	30
Il·lustració23: missatge informant que hi ha hagut un error durant la càrrega.	30
Il·lustració24: missatge demanant la conformitat per executar el procés.	30
Il·lustració25: missatge donant el nombre de lectures fetes.	30
Il·lustració26: missatge informant de quins fitxers falten per carregar.	31
Il·lustració27: pantalla per introduir les lectures.	31
Il·lustració28: pantalla per introduir el número del comptador que volem cercar.	31
Il·lustració29: pantalla per introduir el carrer i el número del comptador que volem cercar.	32
Il·lustració30: missatge d'error al fer la selecció de carrer i número.	32
Il·lustració31: pantalla amb la selecció de comptadors feta.	32

Il·lustració32: diàleg per poder introduir la lectura.	33
Il·lustració33: diàleg d'error a l'introduir la lectura.	33
Il·lustració34: diàleg demanant la confirmació de la lectura entrada.	33
Il·lustració35: diàleg que mostra que la lectura entrada és inferior a la del trimestre anterior.	33
Il·lustració36: diàleg que ens permet guardar la lectura amb incidència o cancel·lar el procés.	33
Il·lustració37: diàleg que mostra que la lectura introduïda té poc consum o massa.	34
Il·lustració38: diàleg informant que s'ha cancel·lat la lectura.	34
Il·lustració39: diàleg informant que s'ha guardat la lectura.	34
Il·lustració40: diàleg demanant que s'introdueixi una incidència.	34
Il·lustració41: diàleg demanant que s'introdueixi una incidència.	34
Il·lustració42: diàleg informant que s'ha guardat la lectura amb una incidència.	34
Il·lustració43: diàleg informant que les dades s'han enviat correctament.	35
Il·lustració44: aplicació per crear els fitxers vialer.json i comptadors.json.	38
Il·lustració45: captura de pantalla un cop executat el servidor i feta la primera crida.	42
Il·lustració46: captura de pantalla després de fer un POST a la pàgina inici.	43

1. Context

Com a responsable del Departament d'Informàtica de l'Ajuntament de Riudoms he de desenvolupar i mantenir el software de gran part del programari.

El programari que porta més feina a l'hora de mantenir és el que realitza la captura de dades i la facturació de l'aigua consumida a Riudoms que subministra el propi Ajuntament.

Dins el tractament de les dades, per tal de recaptar aquest tribut, hi ha dues fases ben diferenciades:

1. Recollida i entrada de les dades de consum i tractament de les incidències.
2. Facturació dels consums de cada usuari.

A nivell de programació, el que comporta més feina és la facturació, ja que intervenen diferents conceptes que poden canviar d'un període a un altre, i s'ha d'anar reprogramant l'aplicatiu.

Però pel que fa a la utilització dels recursos humans, la major part de temps es dedica a la recollida i tractament de dades.

Aquest procés comença amb el treball de camp d'una treballadora de l'Ajuntament que localitza cadascun dels comptadors i n'anota la lectura en un full on apareixen les dades de l'usuari.

Un cop recollides les dades es van introduint al programa i, aquest en fa una anàlisi i busca possibles errors en la lectura. Els possibles errors poden ser dos:

- La lectura actual és inferior a la del trimestre anterior:
 - Lectura errònia.
 - El comptador ha arribat al final de la numeració i torna a començar de 0.
- El consum que surt és massa gran o massa petit, si el comparem amb les últimes 8 lectures.

En el cas de lectura errònia i del consum massa gran o massa petit, s'ha de tornar a fer la lectura i comprovar que aquesta sigui correcta.

En el cas que la lectura sigui incorrecta, s'introdueix la nova i es torna a analitzar.

En el cas que la lectura sigui correcta i saltin qualsevol dels dos, avisos actuarem de la següent manera:

- La lectura actual és inferior a la del trimestre anterior i no és una volta de la numeració: es notifica a la brigada municipal per a que verifiqui el correcte funcionament del comptador, mirant que no hagi estat manipulat.
S'ha de tenir en compte que quan hi ha un tall en el subministrament de l'aigua, i per efecte de la gravetat, l'aigua que hi ha a l'interior de les cases surt cap a l'exterior i fa girar els comptadors d'aigua al revés, restant consum d'aigua. Això és degut a que els comptadors d'aigua són mecànics, no electrònics.

- El consum que surt és massa gran: Si és molt elevat s'avisarà al propietari d'aquesta incidència. Si aquest no ha fet res d'anormal pel que fa al consum, és possible que tingui alguna fuga d'aigua (cisterna del lavabo que no tanca bé, canyeria trencada,...), la qual cosa haurà de reparar per tal de no tenir una sorpresa en la següent factura.
- El consum que surt és massa petit: Si això es produeix durant uns trimestres consecutius es pot mirar de fer una revisió del comptador. Podria ser que s'hagués quedat encallat o que, per a l'efecte de la calç, el comptador marqués menys consum.

Un cop tenim totes les lectures entrades, es procedeix a fer la facturació amb els paràmetres de cobrament que toquen i aquest s'envien a cobrament.

2. Paraules clau

MVC, Sencha Touch, HMTL5, Aptana, Github, Javascript, Iphone, Android, IOS, Smartphone, telèfon mòbil.

3. Descripció del projecte

El procés de la lectura de l'aigua dels comptadors, com ja hem vist, és un procés molt lent degut a que s'ha d'anar porta a porta a buscar les dades.

Aquest procés encara s'alenteix més quan es produeixen errors o incidències en les lectures. Això implica tornar a rellegir comptadors que es troben dispersos per tot el nucli urbà sense seguir cap patró, com quan fem les lectures inicials, que es fan per carrers i seguint un itinerari òptim.

El que em proposo de fer és un aplicatiu per a mòbils que eviti fer aquesta segona lectura. Des d'aquest aplicatiu podrem fer les següents operacions:

1. Carregar dades prèvies – Es carregaran les dades prèvies de cada comptador d'aigua per tal de fer un control de les lectures que entrem en un fitxer *json*. Les dades que hi haurà en aquest fitxer seran: el número del comptador, l'adreça, la lectura del trimestre anterior, la lectura mínima acceptable i la lectura màxima acceptable.

La lectura mínima i màxima acceptable són dos valors de fita, que es calculen prèviament:

- la lectura mínima acceptable serà = la mitja dels 8 últims consum -20%.
- la lectura màxima acceptable serà = la mitja dels 8 últims consum +20%.

Així, quan entrem una lectura fora del rang, el dispositiu ens avisarà i podrem fer al moment una comprovació de que hem introduït correctament les lectures.

2. Introducció de les lectures.

- Primer es farà una cerca del comptador, ja sigui per nom del carrer i número o per número del comptador.
 - Si trobem el comptador, introduïm la lectura.
 - Si aquesta està dins dels marges, s'accepta.
 - Sinó, surt un avís informant que està fora dels límits.
 - ✓ Podem confirmar les dades.
 - ✓ Podem tornar a entrar una nova lectura.

- Un cop entrada la lectura podrem introduir alguna incidència.
 - Si no trobem el comptador, introduïm totes les dades del comptador nou (número de comptador, adreça i lectura).
 - Introduïm totes les dades del comptador nou (número de comptador, adreça i lectura).
 - Podem introduir alguna incidència.
3. Un cop finalitzat el procés de lectures podem enviar les dades per a ser tractades pel responsable del servei.

Aquesta aplicació, que anomenaré LECTCOMP, la implementaré amb HTML5 + CSS3 + Javascript per tal que es pugui executar amb mòbils amb Android OS i iPhone OS.

4. Objectiu del projecte

L'objectiu és realitzar una aplicació (HTML5 + CSS3 + Javascript) àgil, entenedora i fàcil d'usar que ens permeti reduir el temps de recollida de lectures, així com evitar errors en aquestes i que puguin donar facturacions errònies.

Resultats esperats

Un cop finalitzat el projecte, s'espera haver assolit els resultats següents:

- Reducció considerable del temps dedicat a la recollida de lectures. Això implicarà una reducció de costos de personal.
- Reducció dels errors posteriors de facturació. La qual cosa també implica una reducció de costos de personal i un millor servei al ciutadà.
- Resolució de les incidències més ràpidament. Les fuites, comptadors trencats..., es detectaran amb antelació.
- S'eliminaran els possibles errors de transcripció de les lectures del paper a l'ordinador.

5. Organització

En aquest apartat es descriu amb detall l'organització temporal del projecte.

Activitats del projecte

El projecte s'ha dividit en 4 parts que corresponen als 4 lliuraments parcials que s'han de dur a terme durant el semestre.

TASCA	DATA DE LLIURAMENT
PAC1 Elaboració del pla de treball del projecte.	3 d'octubre de 2012
PAC2 Instal·lació del programari necessari, estudi de les necessitats, i anàlisi, disseny i implementació de la interfície que hi haurà entre l'usuari i el servidor.	31 d'octubre de 2012
PAC3 Anàlisi, disseny i implementació del procés de càrrega de dades.	28 de novembre de 2012
Anàlisi, disseny i implementació del procés d'emmagatzematge de dades i lliurament final	4 de gener de 2013

PAC1

L'objectiu de la PAC1 és definir el projecte i fer una proposta al nostre tutor. Un cop acceptat s'ha de definir el projecte i marcar el llibre de ruta a seguir, elaborant el pla de treball que haurem de seguir durant el projecte.

- Recerca prèvia sobre el camp del projecte.
- Redacció d'una proposta de projecte.
- Definició del projecte.
- Elaboració del pla de treball.
- Revisió del pla de treball.
- Lliurament de la PAC1.

PAC2

En aquesta fase es procedirà a instal·lar el programari necessari (XAMPP, Sencha Touch 2,...), estudi de les necessitats, anàlisi, disseny i implementació de la interfície que hi haurà entre l'usuari i el servidor. Aquesta interfície haurà de donar resposta a les peticions de l'usuari a través de la pantalla del mòbil mitjançant les funcions que haurem programat al servidor.

- Instal·lació del programari.
- Anàlisi i disseny de la interfície.
- Implementació de la interfície.
- Proves i correccions.
- Informe de la PAC2.

PAC3

En aquesta fase es definirà i implementarà el procés de carregar de dades.

- Anàlisi i disseny del procés de càrrega de dades.
- Implementació.
- Proves i correccions.
- Informe de la PAC3.

Lliurament final

En aquesta fase es definirà i implementarà el procés d'emmagatzematge de dades. Després es durà a terme el joc de proves final, així com la memòria i les diapositives de la presentació.

- Definir i implementar el procés d'emmagatzematge de dades.
- Elaboració del joc de proves.
- Elaboració de la memòria.
- Revisió de la memòria.
- Elaboració de les diapositives de la presentació.

6. Calendari

Desglossarem detalladament el calendari del projecte per a cadascuna de les fases.

Calendari general del projecte

El projecte s'ha dividit en 4 parts que corresponen a les 4 entregues parcials que s'han de dur a terme durant el semestre.

	Nombre de tarea	Duración	Comienzo	Fin
1	Fita externa 1: Inici del curs	0 días	mié 19/09/12	mié 19/09/12
2	+ PAC1	14 días	mié 19/09/12	mié 03/10/12
9	+ PAC2	27 días	jue 04/10/12	mié 31/10/12
23	+ PAC3	27 días	jue 01/11/12	mié 28/11/12
32	+ Lliurament final	37 días	jue 29/11/12	vie 04/01/13

Il·lustració1: calendari general del projecte

El total de dies que s'han previst són 105. Per a fer els càlculs s'ha elaborat un calendari amb 2 i 3 hores diàries i 7 dies a la setmana en funció de la fase i de la disponibilitat personal.

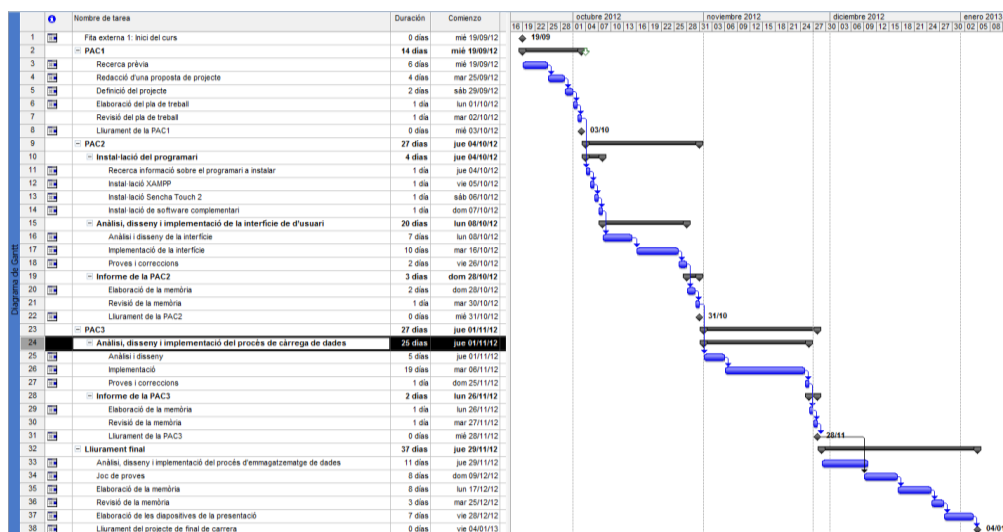
No s'han especificat els dies festius ja que, si és necessari, s'aprofitaran.

Calendari detallat del projecte

El projecte s'ha estructurat en 4 fases:

PAC1	Elaboració del pla de treball del projecte.
PAC2	Instal·lació del programari necessari, estudi de les necessitats i anàlisi, disseny i implementació de la interfície que hi haurà entre l'usuari i el servidor.
PAC3	Anàlisi, disseny i implementació del procés de càrrega de dades.
	Anàlisi, disseny i implementació del procés d'emmagatzematge de dades i lliurament final.

El diagrama complet del projecte és:



Il·lustració2: calendari del projecte.

Calendari de la PAC1

1	Fita externa 1: Inici del curs	0 dies	mié 19/09/12	
2	PAC1	14 días	mié 19/09/12	
3	Recerca prèvia	6 dies	mié 19/09/12	
4	Redacció d'una proposta de projecte	4 dies	mar 25/09/12	
5	Definició del projecte	2 dies	sáb 29/09/12	
6	Elaboració del pla de treball	1 dia	lun 01/10/12	
7	Revisió del pla de treball	1 dia	mar 02/10/12	
8	Lliurament de la PAC1	0 dies	mié 03/10/12	

Il·lustració3: calendari de la PAC1.

Calendari de la PAC2

9	PAC2	27 días	jue 04/10/12	
10	Instal·lació del programari	4 días	jue 04/10/12	
11	Recerca informació sobre el programari a instal·lar	1 dia	jue 04/10/12	
12	Instal·lació XAMPP	1 dia	vie 05/10/12	
13	Instal·lació Sencha Touch 2	1 dia	sáb 06/10/12	
14	Instal·lació de software complementari	1 dia	dom 07/10/12	
15	Anàlisi, disseny i implementació de la interfície de d'usuari	20 días	lun 08/10/12	
16	Anàlisi i disseny de la interfície	7 dies	lun 08/10/12	
17	Implementació de la interfície	10 dies	mar 16/10/12	
18	Proves i correccions	2 dies	vie 26/10/12	
19	Informe de la PAC2	3 días	dom 28/10/12	
20	Elaboració de la memòria	2 dies	dom 28/10/12	
21	Revisió de la memòria	1 dia	mar 30/10/12	
22	Lliurament de la PAC2	0 dies	mié 31/10/12	

Il·lustració4: calendari de la PAC2.

Calendari de la PAC3

23	PAC3	27 días	jue 01/11/12	
24	Anàlisi, disseny i implementació del procés de càrrega de dades	25 días	jue 01/11/12	
25	Anàlisi i disseny	5 dies	jue 01/11/12	
26	Implementació	19 dies	mar 06/11/12	
27	Proves i correccions	1 dia	dom 25/11/12	
28	Informe de la PAC3	2 días	lun 26/11/12	
29	Elaboració de la memòria	1 dia	lun 26/11/12	
30	Revisió de la memòria	1 dia	mar 27/11/12	
31	Lliurament de la PAC3	0 dies	mié 28/11/12	

Il·lustració5: calendari de la PAC3.

Lliurament Final

32	Lliurament final	37 días	jue 29/11/12	vie 04/01/13	
33	Anàlisi, disseny i implementació del procés d'emmagatzematge de dades	11 dies	jue 29/11/12	dom 09/12/12	
34	Joc de proves	8 dies	lun 10/12/12	lun 17/12/12	
35	Elaboració de la memòria	8 dies	mar 18/12/12	mar 25/12/12	
36	Revisió de la memòria	3 dies	mié 26/12/12	vie 28/12/12	
37	Elaboració de les diapositives de la presentació	7 dies	sáb 29/12/12	vie 04/01/13	
38	Lliurament del projecte de final de carrera	0 dies	vie 04/01/13	vie 04/01/13	

Il·lustració6: calendari de la PAC4.

Fites principals del projecte

Data	Descripció
19/09/2012	Fita externa 1: Inici del curs
03/10/2012	Lliurament de la PAC1
31/10/2012	Lliurament de la PAC2
28/11/2012	Lliurament de la PAC3
04/01/2013	Lliurament final

7. Requeriments de material

Detallarem els requeriments de maquinari i programari necessaris per al desenvolupament del projecte.

Requeriments de maquinari

Per desenvolupar i executar el servidor de dades utilitzarem un ordinador estàndard amb sistema operatiu Windows 7.

Per executar l'aplicació utilitzaré un iPhone 4 i un iPad 2 amb IOS 6, per als quals s'han optimitzat les pantalles. El motiu d'aquesta elecció és que l'Ajuntament només disposa d'aquest models.

Requeriments de programari

Programari que s'utilitzarà:

- Sistema operatiu: Windows 7.
- Processador de text: Microsoft Word 2007.
- Disseny del calendari: Microsoft Project 2007.
- Fulla de càlcul: Microsoft Excel 2007.
- Microsoft PowerPoint 2007.
- Java 1.6.
- XAMPP 1.8.1, que inclou:
 - Apache 2.4.3.
 - MySQL 5.5.27.
 - PHP 5.4.7.
 - phpMyAdmin 3.5.2.2.
 - FileZilla FTP Server 0.9.41.
 - Tomcat 7.0.
 - Strawberry Perl 5.16.1.1 Portable.
 - XAMPP Control Panel 3.1.0 (from hackattack142).
- Sencha Touch 2.
- Aptana 3.
- Prototyper – Justinmind.
- Delphi 2010
- Camtasia 8

8. ESTUDI DE COSTOS

Per desenvolupar i executar el projecte no necessitarem adquirir cap dispositiu ni servei, ja que tot es farà utilitzant el material i la infraestructura de l'Ajuntament de Riudoms.

En cas de no tenir aquesta infraestructura, només necessitaríem comprar un dispositiu mòbil, un ordinador per a desenvolupar l'aplicatiu i una connexió a internet.

Els programes utilitzats són tots gratuïts.

DESCRIPCIO	COST
Ordinador per desenvolupar el projecte	600 euros
IPAD 3G	600 euros
3 mesos de connexió a internet	75 euros
Servidor Web (per exemple hostinger)	Gratuït
Despeses de RRHH <ul style="list-style-type: none"> • 16 setmanes x 30 hores setmanals x 40 euros/hora 	12800 euros
COST TOTAL	14075 euros

9. ENTORN DE TREBALL

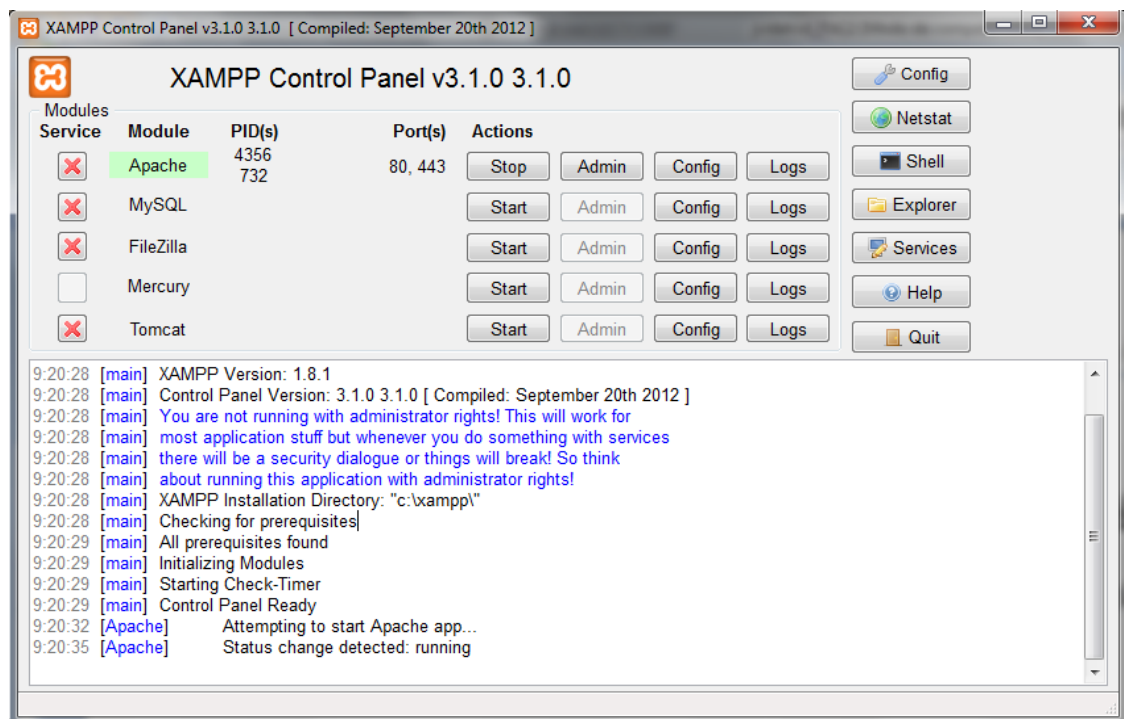
Per a poder desenvolupar el projecte necessitarem les següents eines:

XAMPP

Per tal d'executar el codi, necessitarem un servidor web. Utilitzarem XAMPP, una distribució del servidor web Apache que conté MySQL, PHP i Perl.

És molt fàcil d'instal·lar. Només cal baixar l'executable pel sistema operatiu sobre el que vulguem treballar. Com que treballa en Windows 7, he descarregat la versió Windows des d'aquest link <http://www.apachefriends.org/en/xampp-windows.html>.

Un cop instal·lat l'executable posem en funcionament el servidor a través de Panell de control.



Il·lustració7: panell de control de XAMPP.

Els projectes que vulguem executar es dipositaran en el directori xampp/htdocs.

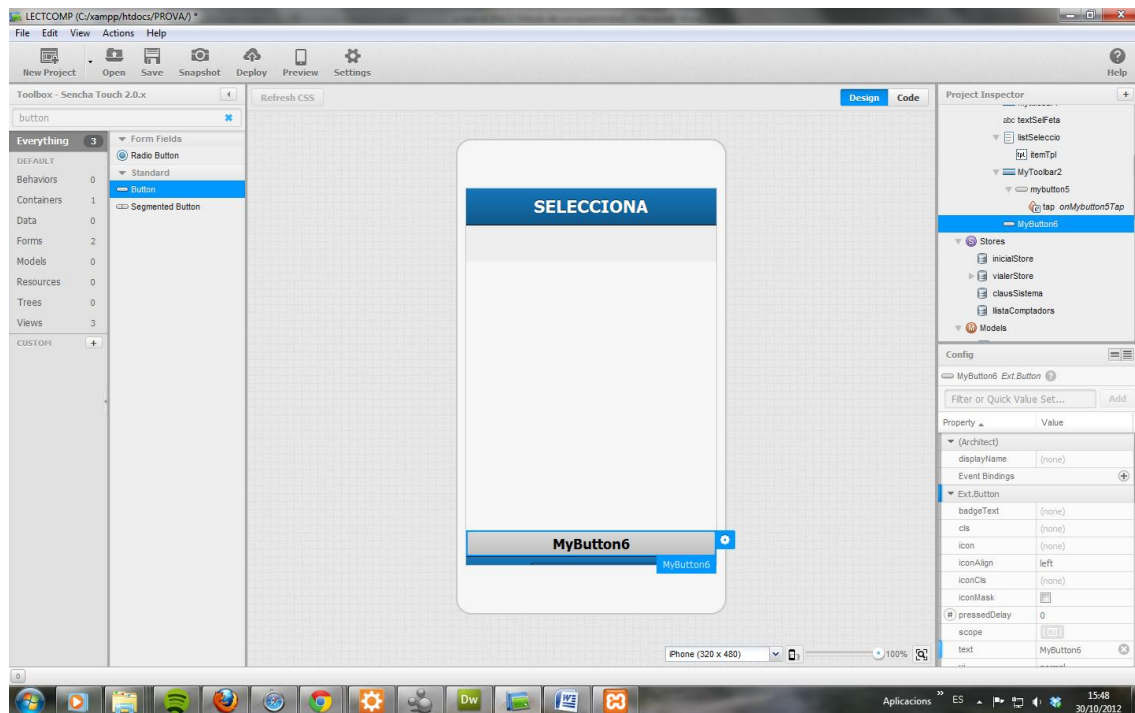
Per executar el projecte posarem en un navegador <http://localhost/xampp/> o [http://localhost/el nostre projecte/](http://localhost/el_nostre_projecte/).

Sencha Touch2

Sencha Touch2 és un entorn de desenvolupament que permet crear aplicacions que es puguin executar en dispositius amb sistemes operatius diferents i sense utilitzar el codi nadiu d'aquest (Android, iOS, BlackBerry...). Aquest entorn utilitza HTML5, CSS i Javascript.

Per tal de fer l'aprenentatge hem instal·lat Sencha Touch 2 Architect. Hi ha disponible una versió gratuïta de 30 dies. Aquesta aplicació permet desenvolupar aplicacions amb Sencha a través d'un entorn gràfic: <http://www.sencha.com/products/architect>.

Un cop instal·lat, Architect ofereix una interfície gràfica fàcil d'utilitzar i que està molt ben complementada per la seva API (<http://docs.sencha.com/touch/2-0/>).



Il·lustració8: entorn de treball de Sencha Touch 2 Architect.

Google Chrome v22 o Safari 5.1

Per tal d'executar Sencha és necessari disposar d'un navegador que ho permeti. Els dos navegadors que ho permeten són Google Chrome (<http://www.google.es/Chrome>) i Safari (<http://www.apple.com/safari/>).

Github

Github és un repositori per dur a terme control de versions que permet realitzar un treball col·laboratiu entre membres d'un mateix.

Hi ha dos tipus de repositoris:

- Els gratuïts: són públics per a tothom.
- De pagament: et permet tenir repositoris de caràcter privat.

A més, mitjançant la branca “gh-pages”, et permet executar el codi en qualsevol navegador de qualsevol dispositiu que soporti Sencha.

El meu repositori és <https://github.com/jmvidalvidal/LECTCOMP>.

Aptana 3

Aptana Studio és un entorn de desenvolupament integrat de software lliure basat en Eclipse i desenvolupat per Aptana Inc., i que pot funcionar sota Windows, Mac y Linux i dóna suport a llenguatges com: Php, Python, Ruby, CSS, Ajax, HTML i Adobe AIR.

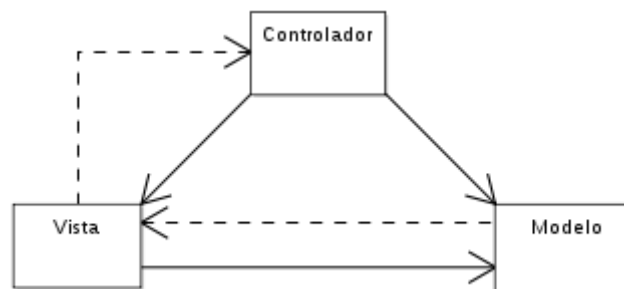
També permet publicar i fer control de versions del programari amb GitHub.

Aquest software l'utilitzarem per desenvolupar el codi, un cop ens haguem familiaritzat amb Sencha a través d'Architect.

10.ARQUITECTURA DEL SISTEMA

Sencha Touch permet desenvolupar aplicacions amb el patró MVC (Model-Vista-Controlador). Aquest model estructura la seva arquitectura en tres parts ben diferenciades:

- El model: defineix l'estructura de camps i dades.
- La vista: tots els components i accions que conformen les pantalles.
- El controlador: respon a events, accions de l'usuari i invoca peticions al model i a la vista.



Il·lustració9: diagrama del Model-Vista-Controlador.

El model MVC ens assegura que l'aplicació serà escalable, modular i fàcil de mantenir.

11. DISSENY I ANÀLISI

Un cop analitzades les necessitats del sistema, dissenyarem una interfície gràfica pel client amb 4 fases ben diferenciades:

1. **Info:** hi haurà un procés informatiu que ens servirà de pantalla d'inici que ens mostrarà les dades del trimestre que estiguem llegint.
2. **Carregar:** hi haurà 5 processos que ens permetran carregar les dades del trimestre. Llegirem 4 fitxers que contindran les dades dels comptadors, el vialer amb el carrers, les dades del trimestre i les claus del sistema. També podrem inicialitzar el fitxer de lectures. Per executar els diferents processos haurem d'introduir un codi de seguretat.
3. **Lectures:** aquí, sempre i quan estiguem dins de les dates establertes, podrem introduir les lectures del comptadors, ja sigui introduint el número de comptador o l'adreça de l'edifici. També podrem afegir un comptador nou que no es trobi a la base de dades.
4. **Enviar:** un cop fetes les lectures, podrem enviar les dades al servidor per a ser tractades amb el software de l'Ajuntament.

Per dur a terme totes les tasques, necessitarem 5 models de dades que carregarem/descarregarem a través de 5 fitxers *json*.

Estructura dels "Models-Store" i fitxers .json

Els models de dades que tindrem seran:

- **modelComptadors.js:** contindrà les dades de tots el comptadors per tal de poder fer les lectures. Els camps que tindrà són aquests:
 - NServei – Número identificatiu del contracte. És únic per a cada casa.
 - nComptador – Número de comptador. Aquest no és sempre el mateix ja que, per les raons que sigui, els comptadors es poden canviar d'un trimestre a un altre.
 - codiCarrer, nomCarrer, numero i pis – Ens donen l'adreça del comptador.
 - lecturaAnterior – Ens mostra la dada de la lectura anterior.
 - minim i maxim – És l'interval òptim que hauria de tenir la nova lectura. Aquests valors es calculen fent la mitjana de les últimes 8 lectures. El mínim és un 25% per sota de la mitjana i el màxim un 25% per sobre de la mitjana.
 - llegit – Aquest camp servirà per saber si hem introduït alguna lectura en el comptador. Inicialment, tots els registres tindran el camp "llegit" en blanc. Per a cada comptador es podran entrar diverses lectures, ja que pot ser necessari haver de rellegir el comptador durant el període de lectures. Un cop llegit, s'actualitzarà el comptador per a que es visualitzi amb una senyal visual.

Fitxer json Comptadors.json	Model modelComptadors.js	Store llistaComptadors.js
<pre>{ "comptadors": [{ "nServei": 744, "nComptador": "0000344", "nomCarrer": "CR SANT VICEN# (DE)", "codiCarrer": "00060", </pre>	<pre>config: { fields: [{name: 'nServei', type: 'int'}, {name: 'nComptador', </pre>	<pre>data: [{ "nServei": 744, "nComptador": "0000344", "nomCarrer": "CR SANT VICEN# (DE)", </pre>

<pre> "numero": "14", "pis": "", "lecturaAnterior": "1138", "minim": "1151", "maxim": "1161", "llegit": 0, {"nServei": 1418, "nComptador": "0000401", "nomCarrer": "CR LLUIS COMPANYS I JOVER", "codiCarrer": "00136", "numero": "10", "pis": "", "lecturaAnterior": "81", "minim": "94", "maxim": "104", "llegit": 0, }} </pre>	<pre> sortDir: 'DESC', sortType: 'asText', type: 'string', {name: 'nomCarrer', type: 'string'}, {name: 'codiCarrer', type: 'string'}, {name: 'numero', type: 'string'}, {name: 'pis', type: 'string'}, {name: 'lecturaAnterior', type: 'int'}, {name: 'minim', type: 'int'}, {name: 'maxim', type: 'int'}, {name: 'llegit', type: 'int'} } } </pre>	<pre> "codiCarrer": "00060", "numero": "14", "pis": "", "lecturaAnterior": "1138", "minim": "1151", "maxim": "1161", "llegit": 0 }, {"nServei": 1418, "nComptador": "0000401", "nomCarrer": "CR LLUIS COMPANYS I JOVER", "codiCarrer": "00136", "numero": "10", "pis": "", "lecturaAnterior": "81", "minim": "94", "maxim": "104", "llegit": 0 }, } } </pre>
--	---	--

- vialerModel.js** - Contindrà les dades de tots els carrers del poble per tal de poder localitzar els comptadors. Els camps que tindrà són aquests:
 - idCarrer – Registre clau per a l’store.
 - codiCarrer – Codi del carrer dins el sistema informàtic de l’Ajuntament.
 - nomCarrer – Nom del carrer. Aquest camp pot semblar redundant amb el següent, però aquest el farem servir per poder ordenar els carrers alfabèticament, ja que el camp nomCarrerVia porta el tipus de via davant del nom.
 - nomCarrerVia – Nom del carrer amb el tipus de via incorporat. Això és necessari pel fet que tenim carrers amb el mateix nom, però amb tipus de via diferent (C/Sant Antoni – PI/ Sant Antoni).

Fitxer json vialer.json	Model vialerModel.js	Store vialerStore.js
<pre> {"vialer": [{"idCarrer": 1, "codiCarrer": "00179", "nomCarrer": "ALBEREDES (DE LES)", "nomCarrerVia": "CM ALBEREDES (DE LES)" }, {"idCarrer": 2, "codiCarrer": "00098", "nomCarrer": "ALCOLEA DEL PINAR (D')", "nomCarrerVia": "CA ALCOLEA DEL PINAR (D')" }, ]} </pre>	<pre> config: { fields: [{name: 'idCarrer', type: 'int'}, {name: 'codiCarrer', type: 'string'}, {name: 'nomCarrer', type: 'string'}, {name: 'nomCarrerVia', type: 'string'}] } </pre>	<pre> config: { data: [{idCarrer: 1, codiCarrer: '00179', nomCarrer: 'ALBEREDES (DE LES)', nomCarrerVia: 'CM ALBEREDES (DE LES)' }, {idCarrer: 2, codiCarrer: '00098', nomCarrer: 'ALCOLEA DEL PINAR (D\')', nomCarrerVia: 'CA ALCOLEA DEL PINAR (D\')' }, ] } </pre>

- inicialModel.js** – Contindrà les variables del trimestre. Aquestes dades ens permetran saber sobre quin trimestre estem treballant i en quines dates podem fer les lectures. Els camps que tindrà són aquests:
 - idCamp – Nom de la variable que guardem.
 - textCamp – Text identificatiu de la variable.

- valorCamp – Valor de la variable.

Fitxer json Info.json	Model inicialModel.js	Store inicialStore.js
<pre>{ "info": [{ idCamp: 'trimestre', textCamp: '', valorCamp: '3r Trimestre 2012' }, { idCamp: 'iniTrim', textCamp: 'Inici lectures:', valorCamp: '30/08/2012' }, { idCamp: 'fiTrim', textCamp: 'Fi lectures:', valorCamp: '30/12/2012' }] }</pre>	<pre>config: { fields: [{ name: 'idCamp' }, { name: 'textCamp' }, { name: 'valorCamp' }] }</pre>	<pre>data: [{ idCamp: 'trimestre', textCamp: '', valorCamp: '3r Trimestre 2012' }, { idCamp: 'iniTrim', textCamp: 'Inici lectures:', valorCamp: '30/08/2012' }, { idCamp: 'fiTrim', textCamp: 'Fi lectures:', valorCamp: '30/12/2012' }],</pre>

- **clausModel.js** – Aquest model contindrà els codis de seguretat per poder fer la càrrega dels fitxers al sistema a l'inici del trimestre. Els camps del model seran aquests:
 - id – Camp identificador per a l'store.
 - clau – Codi de seguretat.

Fitxer json ClausSistema.json	Model clausModel.js	Store clausSistema.js
<pre>{ "clausSistema": [{ id: 1, clau: 'ecros84' }, { id: 2, clau: 'carbon600' }, { id: 3, clau: 'vidi' }] }</pre>	<pre>config: { fields: [{ name: 'id', type: 'int' }, { name: 'clau', type: 'string' }] }</pre>	<pre>data: [{ id: 1, clau: 'ecros84' }, { id: 2, clau: 'carbon600' }, { id: 3, clau: 'vidi' }],</pre>

- **lecturesModel.js** – Aquest model contindrà les lectures que anem introduint. Aquest model ens generarà un fitxer de sortida. Els camps del model seran aquests:
 - valor – Valor de la lectura.
 - nComptador – Número del comptador.
 - dataLectura - Data de la lectura.
 - esVolta – Ens indicarà si al fer la lectura hem detectat que el comptador ha arribat al final de la seva numeració i s'ha posat a 0.
 - incidencia – Ens guardarà les incidències introduïdes.

Fitxer json lectures.json	Model lecturesModel.js	Store lecturesStore.js
<pre>{ "clausSistema": [{ valor: 2442, nComptador: "93252390", dataLectura: "25/09/2012 09:00", esVolta: 0, incidencia: "" }, { valor: 2345, nComptador: "24485258", dataLectura: "25/09/2012 09:02", esVolta: 1 }] }</pre>	<pre>config: { fields: [{ name: 'valor', type: 'int' }, { name: 'nComptador', type: 'string' }, { name: 'dataLectura', type: 'string' }, { name: 'esVolta', type: 'int' }, { name: 'incidencia', type: 'string' }] }</pre>	<pre>config: { data: [{ id: 2442, nComptador: '93252390', dataLectura: '25/09/2012 09:00', esVolta: 0, incidencia: "" }, { valor: 2345, nComptador: '24485258', dataLectura: '25/09/2012 09:02', esVolta: 1 }] }</pre>

<pre>incidencia: "Comptador deteriorat" }, }} }</pre>		<pre>incidencia: 'Comptador deteriorat' }, }}</pre>
---	--	---

Estructura del controlador

Tindrà un controlador, anomenat controlador.js, que s'executarà a l'inici de l'aplicació i farà la càrrega inicial de les dades del trimestre i de les claus del sistema.

Estructura de la vista Info

Serà la pantalla principal per poder accedir als diferents menús, i ens mostrarà la informació del trimestre sobre la que estiguem treballant i les dates sobre les que es faran les lectures dels comptadors. Quan iniciem l'aplicació, es carregaran les dades del trimestre i les claus que es troben al servidor en format json.



Il·lustració10: pantalla inicial de l'aplicació

Mitjançant els botons inferiors podrem accedir a les altres 3 etapes (Carregar, Lectures i Enviar).

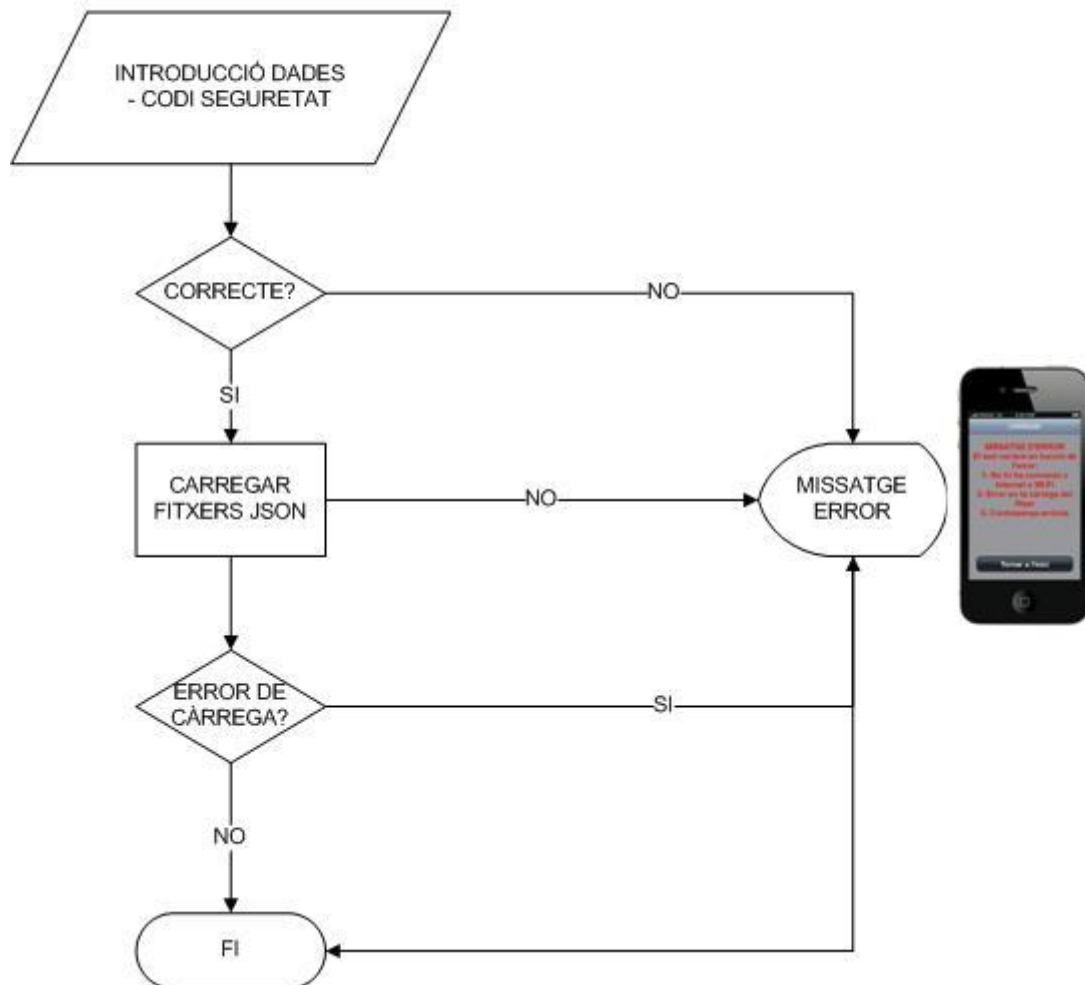
Estructura de la vista Carregar

En aquesta fase podrem carregar el fitxer json amb les dades dels comptadors que hem de llegir, el vialer amb tots els carrers del poble, els codis de seguretat i les dades del trimestre.

Podrem inicialitzar l'store amb les lectures introduïdes, així com, saber quantes lectures portem introduïdes.

El flux de dades dels 4 processos de càrrega de fitxers json serà el següent:

CÀRREGA DELS FITXERS JSON



Il·lustració11: diagrama de flux de la càrrega dels fitxers json.

En aquesta pantalla podem realitzar 5 processos diferents:

- **“Carregar Dades Trimestre”** - Càrrega del fitxer json amb les dades del trimestre i les claus del sistema. Aquest procés no necessita contrasenya.
- **“Carregar Vialer”** - Càrrega del fitxer json amb les dades del vialer. Aquest procés necessita ser validat mitjançant una contrasenya.
- **“Carregar Comptadors”** - Càrrega del fitxer json amb les dades dels comptadors. Aquest procés necessita ser validat mitjançant una contrasenya.
- **“Inicialitzar fitxer de lectures”** – Inicialitza l’store que conté les lectures que s’han anat introduint. Aquest procés necessita ser validat mitjançant una contrasenya.
- **“Número de lectures”** – Informa del número de registres que té l’store lectures.

En el cas que no s'hagi fet la càrrega de les dades del trimestre i/o les claus del sistema, l'aplicació bloquejarà el botons que permeten carregar el vialer i els comptadors, i la inicialització les lectures.



Il·lustració12: pantalla per carregar els fitxers json.

Durant els processos en que sigui necessari introduir la contrasenya, quan aquesta sigui errònia, i durant els processos de càrrega amb errors, el sistema ens retornarà un error.

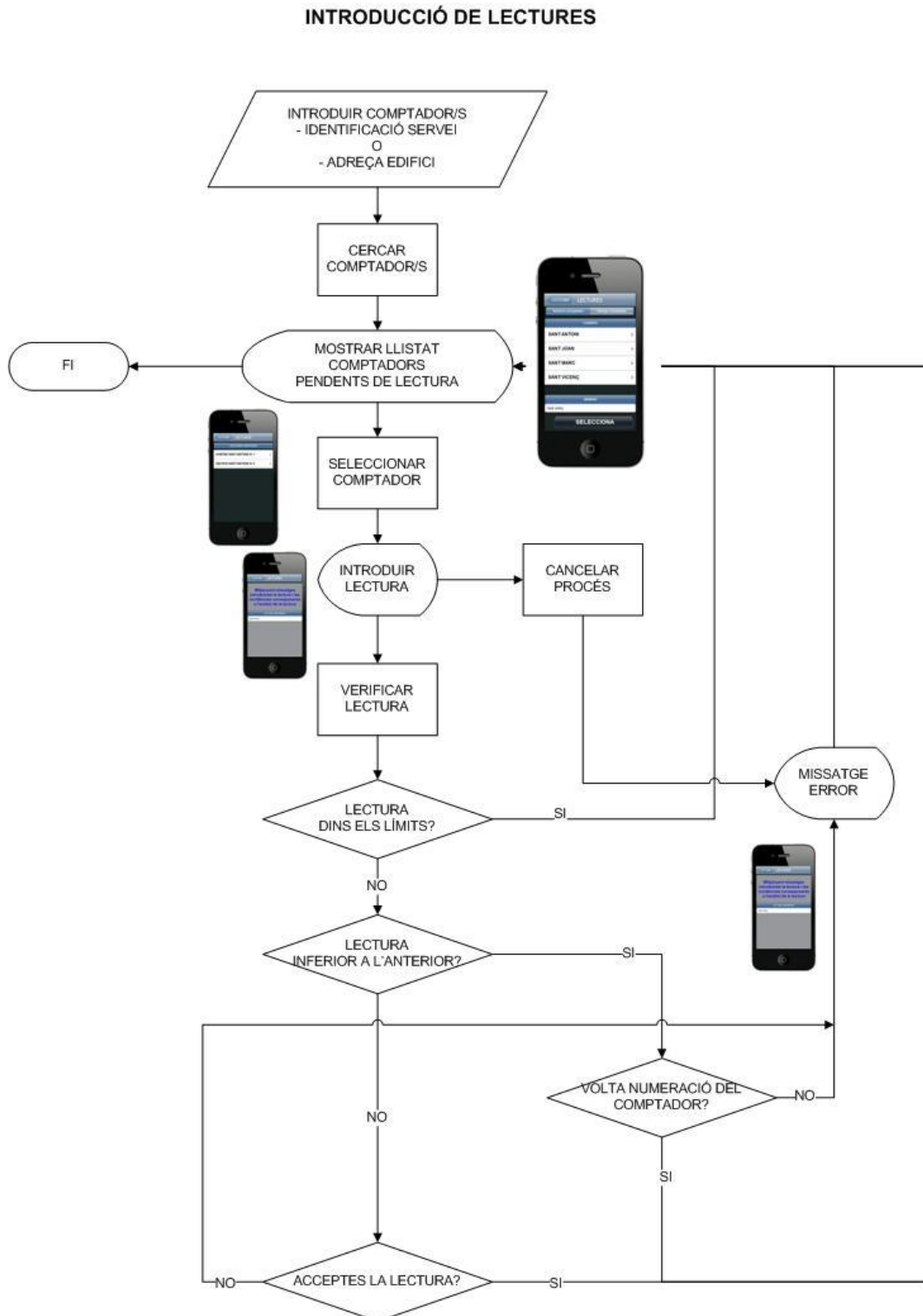


Il·lustració13: missatge d'error de la càrrega de les dades inicials.

Estructura de la vista Lectures

En aquesta fase anirem introduint les lectures dels comptadors que anem seleccionant.

El flux de dades d'aquesta fase simplificat és el següent:



Il·lustració14: diagrama de flux per introduir i validar lectures.

Primer de tot, haurem de fer una selecció de comptadors. Aquesta selecció es pot fer introduint el número de comptador o introduint l'adreça i número d'una finca.



Il·lustració15: pantalles per seleccionar els comptadors per número de carrer o adreça, o afegir-ne un de nou

En els casos que es faci una selecció, ens donarà com a resultat una llista de comptadors.



Il·lustració16: pantalla que ens mostra la selecció de comptadors feta.

Ara és qüestió d'anar prement sobre el comptador del que volem introduir la lectura. El procés d'introducció de la lectura, les incidències i l'anàlisi d'aquesta es farà mitjançant missatges de diàleg que ofereix Sencha. Així ens estalviem crear pantalles noves i passar paràmetres.

Primer de tot ens sortirà un missatge que ens permetrà introduir la lectura. Tenim dues opcions:

1. Cancel·lar el procés – Es mostra un missatge de lectura cancel·lada i es retorna a la llista de selecció.
2. Introduir la lectura – Introduïm la lectura i l'analitzem.

Un cop analitzada la lectura, mostrem els possibles resultats:

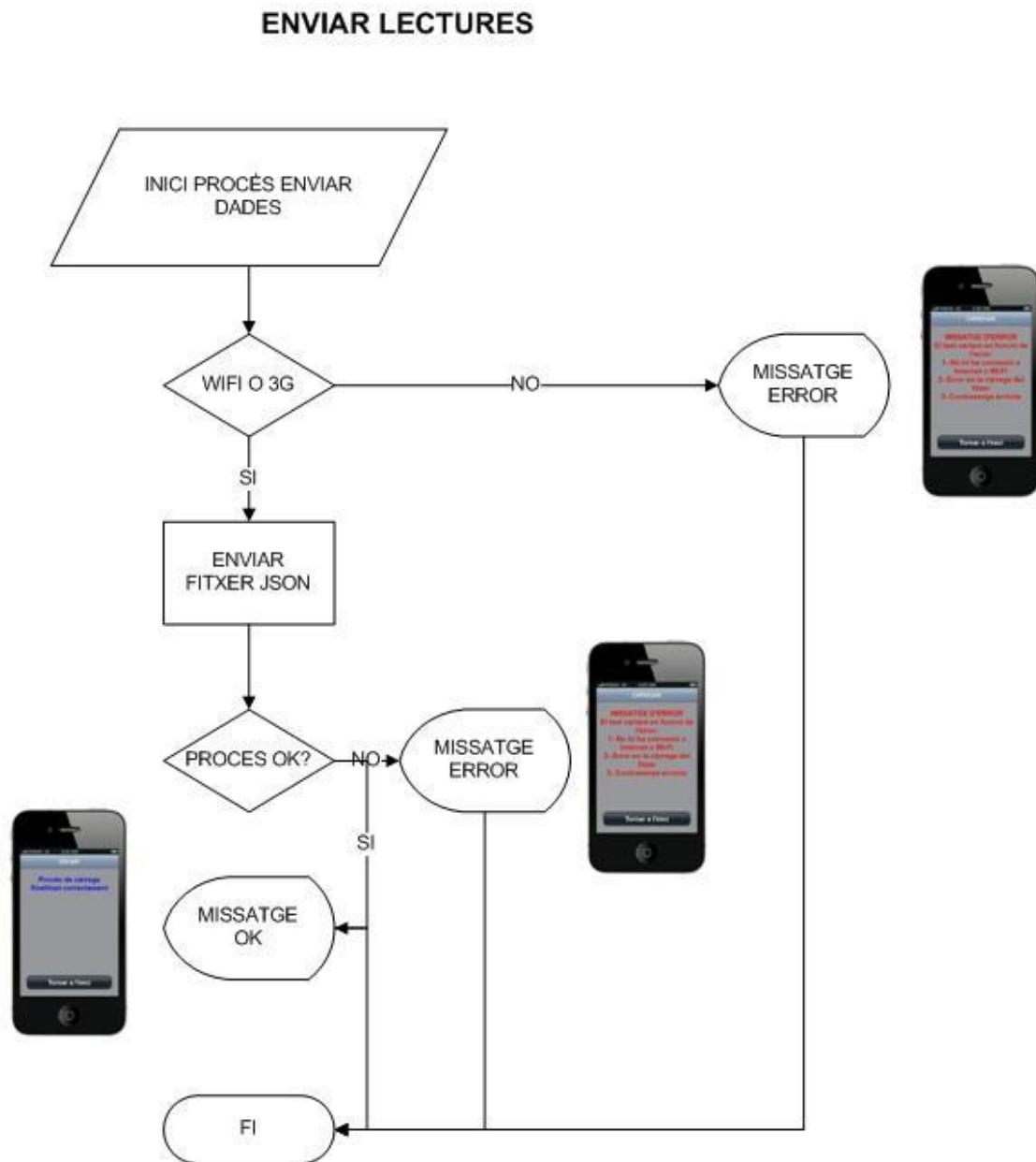
1. Lectura dins dels límits raonables. Ens demana conformitat de la lectura:
 - a. Podrem cancel·lar la lectura - Es mostra un missatge de lectura cancel·lada i es retorna a la llista de selecció.
 - b. Acceptar-la – S'emmagatzema la lectura i es retorna a la llista de selecció.
 - c. Acceptar-la introduint una incidència.
 - i. Introduïm la incidència, guardem la lectura i retornem a la llista de selecció.
 - ii. Cancel·lem, atès que no s'ha introduït cap incidència i retornem a la llista de selecció.
 - iii. Cancel·lem el procés de lectura i retornem a la llista de selecció
2. Lectura inferior a la del trimestre anterior.
 - a. El comptador ha donat la volta?
 - i. Sí. Acceptar-la – S'emmagatzema la lectura i es retorna a la llista de selecció.

- ii. Sí. Acceptar-la introduint una incidència.
 - 1. Introduïm la incidència, guardem la lectura i retornem a la llista de selecció.
 - 2. Cancel·lem, atès que no s'ha introduït cap incidència i es retorna a la llista de selecció.
 - 3. Cancel·lem el procés de lectura i retornem a la llista de selecció
- iii. No
 - 1. Acceptar-la introduint una incidència.
 - a. Introduïm la incidència, guardem la lectura i retornem a la llista de selecció.
 - b. Cancel·lem, atès que no s'ha introduït cap incidència i es retorna a la llista de selecció.
 - c. Cancel·lem el procés de lectura i retornem a la llista de selecció.
 - 2. Cancel·lar el procés i es retorna a la llista de selecció.
- 3. La lectura és més gran que l'anterior, però està fora dels límits considerats normals.
 - a. Acceptar – Guardem la lectura i es retorna a la llista de selecció.
 - b. Acceptar-la, introduint una incidència.
 - i. Introduïm la incidència, guardem la lectura i retornem a la llista de selecció.
 - ii. Cancel·lem, atès que no s'ha introduït cap incidència i es retorna a la llista de selecció.
 - iii. Cancel·lem el procés de lectura i retornem a la llista de selecció
 - c. Cancel·lar la lectura i es retorna a la llista de selecció.

En la pantalla lectura, també podem introduir les dades dels comptadors que no es trobin en la nostra base de dades. Això serà degut a que el comptador s'haurà instal·lat durant el procés de lectures. En aquest cas, l'aplicació ens permetrà introduir el número de comptador, l'adreça del comptador i la lectura, i serà introduïda al sistema sense dur a terme cap validació.

Estructura de la vista Enviar

En aquesta fase podrem enviar les lectures que hem anat recollint. El flux de dades d'aquest procés serà el següent:



Il·lustració17: diagrama de flux d'enviar lectures.

Aquest procés no tindrà cap pantalla associada. Des del menú inicial, premerem el botó de lectures i s'executarà el procés.

Ens demanarà confirmació per enviar les dades.



Il·lustració18: missatge demanant conformitat per iniciar el procés d'enviar el fitxer.

Un cop acceptat, el sistema intentarà enviar el fitxer. Si no tenim connexió a Internet o WI-FI mostrarà una pantalla d'error, i finalitzarà la fase.



Il·lustració19: missatge d'error del procés d'enviar les dades.

En el cas que hi hagi connexió, procedirem a enviar el fitxer lectures.json.

Si aquest procés falla, es visualitzarà la pantalla d'error anterior amb el missatge corresponent i finalitzarà la fase.

12. IMPLEMENTACIÓ

La implementació s'ha iniciat amb Sencha Touch 2 Architect, però en el moment que he assimilat l'estructura del codi, he passat a implementar l'aplicació amb el codi font. Per a fer-ho, he utilitzat Aptana. A més, aquest programa m'ha permès anar penjant les diferents versions al repositori GITHUB.

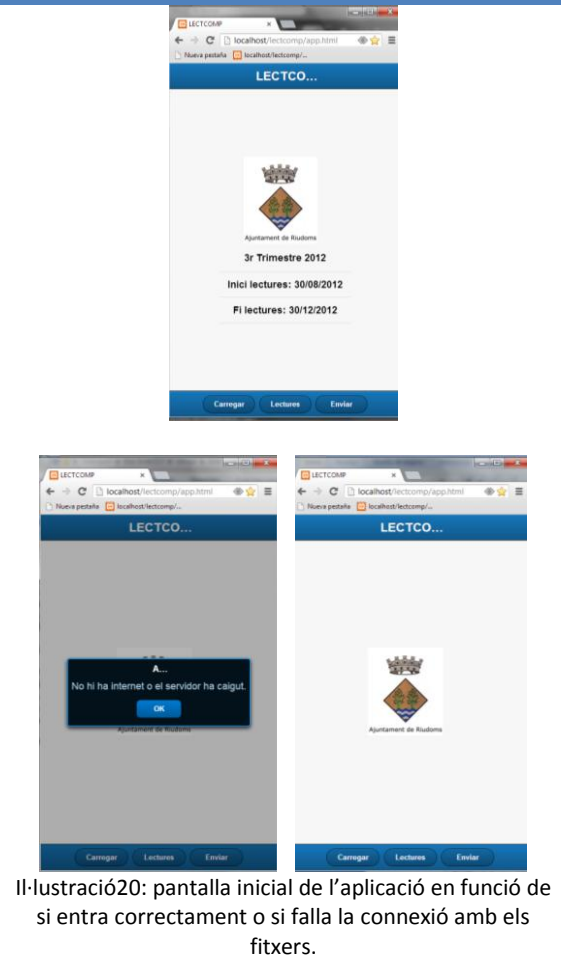
Les captures de pantalla les he fet amb el navegador Safari, ja que encara no he carregat l'aplicació en cap dispositiu mòbil.

INFO

CÀRREGA INICIAL DE L'APLICACIÓ

Al carregar l'aplicació el sistema intentarà carregar els fitxers json del servidor que contenen les dades del trimestre i el fitxer de les contrasenyes.

En el cas, que no puguem fer la connexió amb els fitxers, el sistema ens mostrarà un missatge . Llavors, a la pantalla principal no es mostraran els valors del trimestre.



CARREGAR

CÀRREGAR DELS FITXERS JSON

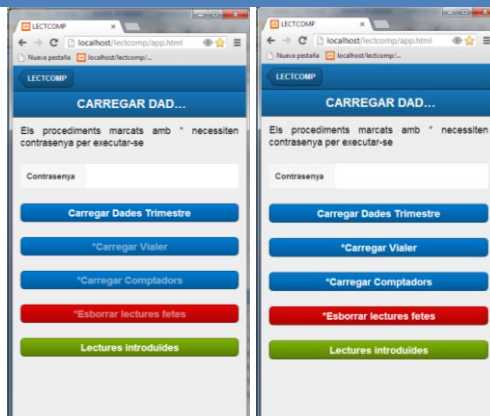
El botó “Carregar” de la pantalla principal ens permetrà accedir a la pantalla “Carregar”.

Podrem carregar els fitxers que vulguem, així com inicialitzar el fitxer lectures i veure les lectures que portem introduïdes.

Els botons que tenen un asterisc necessiten un codi de seguretat. Aquests són uns processos crítics, ja que esborren dades del sistema.

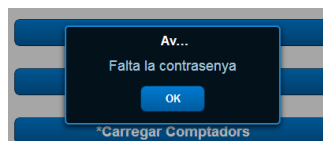
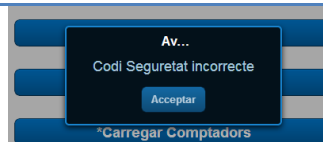
En el cas que les dades del trimestre o les claus no s’hagin carregat correctament al principi, els botons de carregar el vialer i els comptadors, i el d’esborrar les lectures, estaran inhabilitats.

Amb el botó LECTCOMP podrem tornar a la pantalla principal.



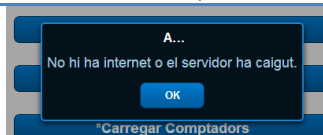
Il·lustració21: pantalla per carregar els fitxers json, esborrar les lectures i consultar les lectures introduïdes.

Els processos que necessitem introduir la contrasenya, si aquesta no és correcta, ens retornarà un missatge d’error.



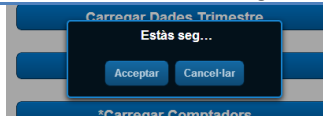
Il·lustració22: missatge informant que el codi de seguretat és incorrecte o que no s’ha introduït.

En el cas que intentem fer una càrrega de fitxers json i durant el procés es produeixi algun error, ens mostrarà un missatge.



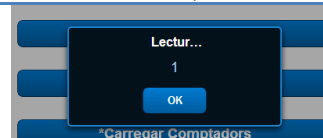
Il·lustració23: missatge informant que hi ha hagut un error durant la càrrega.

Si volem esborrar les lectures que hi ha introduïdes fins al moment, el sistema ens demanarà una conformitat, ja que és un procés crític.



Il·lustració24: missatge demanant la conformitat per executar el procés.

Si volem saber quantes lectures portem introduïdes, premem l’últim botó i ens donarà el resultat

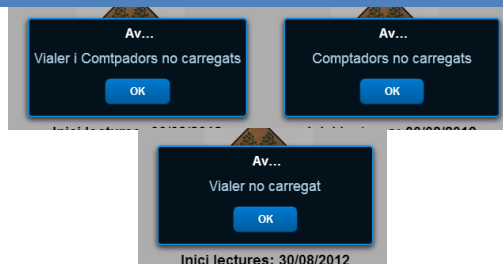


Il·lustració25: missatge donant el nombre de lectures fetes.

LECTURES

PANTALLA PRINCIPAL PER INTRODUIR LES LECTURES

El botó “Lectures” de la pantalla principal ens permetrà accedir a la pantalla “Lectures”. Però per accedir-hi, l'aplicació comprovarà que els fitxers vialer i comptadors estiguin carregats, sinó ens retornarà un missatge d'error.

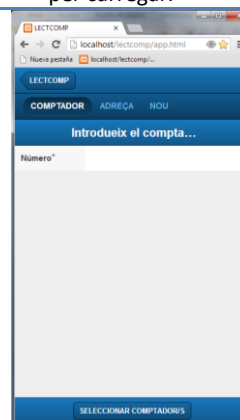


Il·lustració26: missatge informant de quins fitxers falten per carregar.

La pantalla “Lectures” està dividida en tres tabuladors: un per cercar el comptador pel seu número, l'altre per cercar els comptadors d'una finca per l'adreça i l'altre per poder introduir un comptador que no es trobi a la base de dades.

En aquesta pantalla podrem realitzar 3 accions:

1. Cercar un comptador pel seu número.
2. Cercar comptadors per l'adreça.
3. Afegir un nou comptador.
4. Tornar al menú principal amb el botó LECTCOMP.

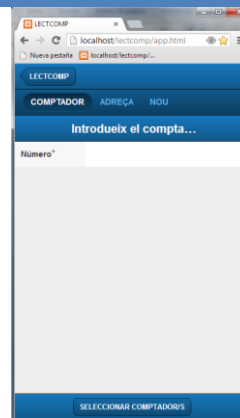


Il·lustració27: pantalla per introduir les lectures.

CERCAR COMPTADOR PEL SEU NÚMERO

Introduïm el número del comptador i premem seleccionar.

Això ens porta a la pantalla de selecció de comptadors per poder entrar les lectures.



Il·lustració28: pantalla per introduir el número del comptador que volem cercar.

CERCAR COMPTADORS PER ADREÇA

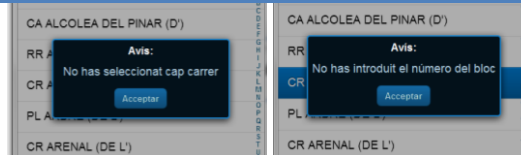
Aquí podem seleccionar el comptador seleccionant el carrer i introduint el número de l'edifici.

Un cop fet, premem "SELECCIONAR COMPTADORS" i accedirem a la pantalla de selecció de comptadors per poder entrar les lectures.



Il·lustració29: pantalla per introduir el carrer i el número del comptador que volem cercar.

Si no seleccionem cap carrer o no introduïm el número, els sistema ens donarà missatge d'error.



Il·lustració30: missatge d'error al fer la selecció de carrer i número.

SELECCIÓ DELS COMPTADORS

Aquesta pantalla ens mostrarà el resultat de la selecció de comptadors feta en la pantalla "Lectures".

Per introduir les lectures hem de clicar damunt de cada comptador. Això iniciarà el procés d'introducció de lectures i de validació d'aquestes.

Per aquest procés no hem creat cap pantalla. Es farà mitjançant diàlegs que anirà mostrant el sistema.

Els comptadors llegits és marquen amb una icona.

Tot i això, els comptadors marcats es poden tornar a llegir. Això és així ja que en comptadors problemàtics es fan diverses lectures durant el període per veure l'evolució del consum.



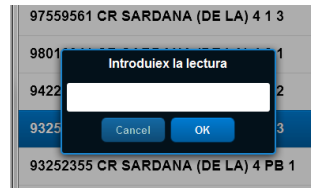
Il·lustració31: pantalla amb la selecció de comptadors feta.

PROCÉS D'INTRODUCCIÓ DE LECTURES

Per iniciar el procés d'introducció de les lectures, hem de picar damunt del comptador escollit.

Això ens obrirà un diàleg que ens permetrà fer dues operacions:

1. Cancel·lar el procés.
2. Introduir la lectura.



Il·lustració32: diàleg per poder introduir la lectura.

Si cancel·lem el procés ens mostrarà un missatge i finalitzarà.

Passarà el mateix si acceptem la lectura i aquesta està en blanc.



Il·lustració33: diàleg d'error a l'introduir la lectura.

INTRODUCCIÓ D'UNA LECTURA AMB UN CONSUM ÒPTIM

Un cop introduïda la lectura i acceptada, el sistema l'analitza.

Si aquesta està dins dels límits òptims, el sistema ens demanarà que confirmem la lectura. Les accions que podem realitzar són:

1. Cancel·lar el procés de lectura.
2. Acceptar la lectura i guardar-la.
3. Acceptar la lectura i guardar-la introduint una incidència.



Il·lustració34: diàleg demanant la confirmació de la lectura entrada.

INTRODUCCIÓ D'UNA LECTURA INFERIOR A LA LECTURA DEL TRIMESTRE ANTERIOR

Un cop introduïda la lectura i acceptada, el sistema l'analitza.

Si aquesta és inferior a la lectura del trimestre anterior, pot ser degut a tres casos.

1. Hem introduït malament la lectura.
2. El comptador ha arribat al final de la numeració i ha tornat a 0.
3. Realment la lectura és inferior: ens vam equivocar a l'introduir la lectura del trimestre anterior, s'ha manipulat el comptador,....

Les accions ha realitzar són aquestes:

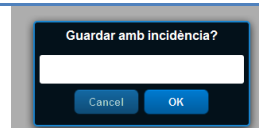
1. No hi ha volta de comptador.
2. Hi ha volta de comptador. Per tant, acceptem i guardem la lectura.
3. Hi ha volta de comptador introduint una incidència. Per tant, acceptem i guardem la lectura amb una incidència.



Il·lustració35: diàleg que mostra que la lectura entrada és inferior a la del trimestre anterior.

Si la lectura no correspon a una volta de comptador, el sistema ens permetrà fer dues accions:

1. Cancel·lar la lectura.
2. Acceptar-la introduint una incidència.



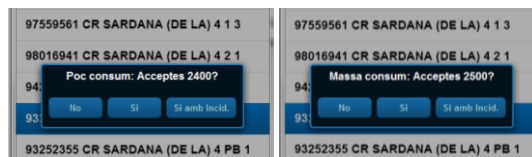
Il·lustració36: diàleg que ens permet guardar la lectura amb incidència o cancel·lar el procés.

INTRODUCCIÓ D'UNA LECTURA QUE NO ÉS ÒPTIMA

Un cop introduïda la lectura i acceptada, el sistema l'analitza.

Si aquesta està per sota o per sobre del que seria òptim, el procés ens permetrà realitzar les següents accions:

1. Cancel·lar la lectura.
2. Acceptar la lectura i guardar-la.
3. Acceptar la lectura i guardar-la, introduint una incidència.

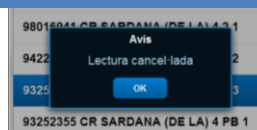


Il·lustració37: diàleg que mostra que la lectura introduïda té poc consum o massa.

CANCEL·LAR LECTURA

Aquesta opció ens apareix diverses vegades en el procés de validació de la lectura.

El procés es cancel·larà i ens donarà un missatge.

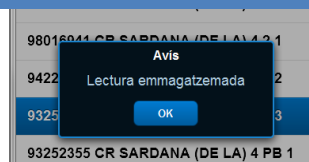


Il·lustració38: diàleg informant que s'ha cancel·lat la lectura

ACCEPTAR I GUARDAR LECTURA

Aquesta opció ens apareix diverses vegades en els procés de validació de la lectura.

El procés guarda la lectura i ens dona un missatge.



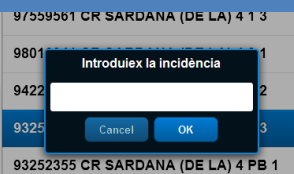
Il·lustració39: diàleg informant que s'ha guardat la lectura.

ACCEPTAR I GUARDAR LECTURA AMB INCIDÈNCIA

Aquesta opció ens apareix varies vegades en els procés de validació de la lectura.

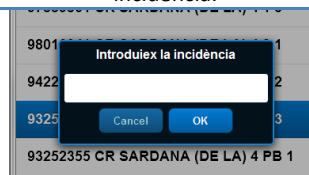
El procés ens permet introduir una incidència juntament amb la lectura.

Podem cancel·lar la lectura o podem introduir la incidència i acceptar la lectura.



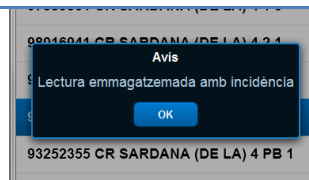
Il·lustració40: diàleg demanant que s'introdueixi una incidència.

Si acceptem la lectura amb incidència, però aquesta la deixem en blanc, el sistema cancel·la la lectura i ens dona un missatge.



Il·lustració41: diàleg demanant que s'introdueixi una incidència.

Si acceptem la lectura amb una incidència, el sistema ens donarà un missatge.

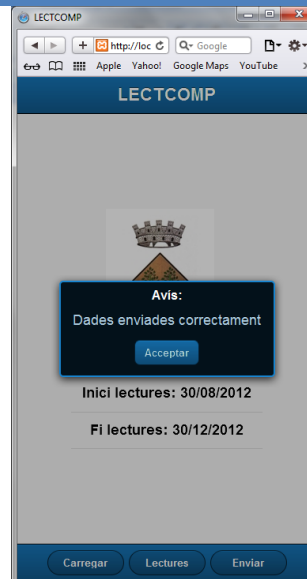


Il·lustració42: diàleg informant que s'ha guardat la lectura amb una incidència.

ENVIAR

ENVIAR EL FITXER JSON AMB LES LECTURES

El botó “Enviar” iniciarà el procés d'enviament de les lectures. Un cop realitzat l'enviament, el sistema ens donarà un missatge.



Il·lustració43: diàleg informant que les dades s'han enviat correctament.

Per poder realitzar l'enviament, el que he fet és crear un duplicat de l'store que connecta amb el protocol AJAX, amb un fitxer PHP, que rep el resultat i l'emmagatzema en un fitxer al servidor.

D'aquesta fase només s'ha programat aquesta tasca i es deixa per desenvolupar la interpretació de les dades per a un projecte futur.

El fitxer PHP simplement el que fa és rebre les dades a través d'un REQUEST i emmagatzemar-les en un fitxer.

```
<?php
foreach($_REQUEST as $var => $value)
{ $fh = fopen("datos_out.json", 'w')
  or die("Error l'obrir el fitxer");
  $dadesJSON = $value;
  fwrite($fh, $dadesJSON) or die ("Error escrivint al fitxer");
  echo "Registres guardats";
}
fclose($fh);
?>
```

OPTIMITZACIÓ DE LA VISUALITZACIÓ DE LA LLISTA DE CARRERS

Durant la programació de l'aplicació m'he trobat amb un problema de disseny que després d'hores de proves, consultes amb el tutor i cerques per internet, hem arribat a la conclusió que és un problema d'execució de Sencha.

El problema el trobem en la llista de carrers per seleccionar en el menú lectures. Aquesta llista s'hauria d'adaptar en alçada i amplada automàticament en funció de la mida de la pantalla.

El fet de trobar-se dins "TabView", no ens mostra la llista si no li posem una llargada fixa. Com que tenim dispositius diferents ens trobem que, per a que quedi bé i ompli tota la pantalla, no sabem quina ha de ser la mida.

Per solucionar el tema he optat per redimensionar la llista al crear-la en funció del dispositiu amb el que treballem.

En el cas de que tinguem un iPhone (ho deduïm per la mida de la pantalla) hi posarem una mida de 220 píxels. En cas que tinguem un iPad (serà el dispositiu utilitzat per a fer les lectures) li posarem una mida de 780 píxels. Per a la resta de casos li posarem una mida del 60% de la mida de la pantalla.

```
h=screen.height;
if (h==480){//iPHONE 4
    Ext.ComponentQuery.query('#listAdreca')[0].setHeight(220);
} else if (h==1024) {//IPAD
    Ext.ComponentQuery.query('#listAdreca')[0].setHeight(780);
} else {//altres dispositius
    h=parseInt(screen.height*60/100);
    Ext.ComponentQuery.query('#listAdreca')[0].setHeight(h);
}
```

13.Joc de proves

Per fer el joc de proves he estat treballant amb diferents comptadors. A continuació poso un exemple amb les dades entrades i els resultats obtinguts:

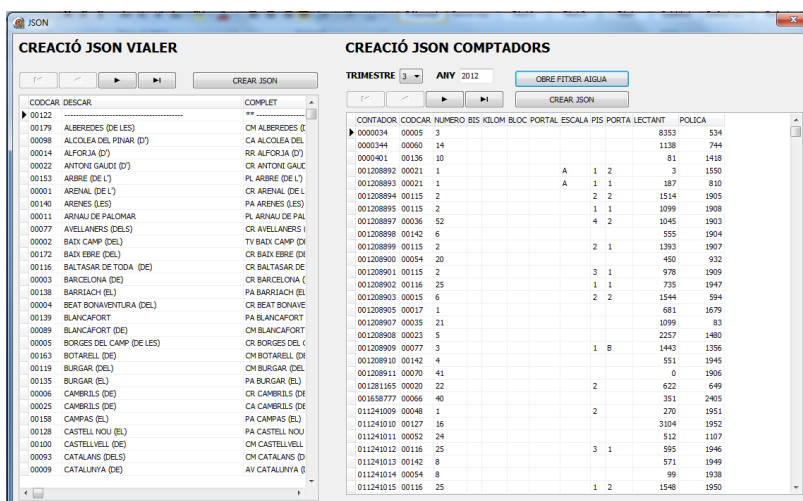
- Carregar el sistema –
 - Pels procediments en que és necessària una contrasenya, introduïm “vidi”. Si ho deixem en blanc, retorna un missatge d’error.
 - Anem prement els diferents botons i farà la funció especificada.
- Lectures –
 - Cerquem comptadors:
 - Si cerquem el comptador 93252350, ens retorna la llista amb el comptador del C/Sardana, 4 2-3.
 - Si cerquem els comptadors del C/Sardana, 4, ens retorna una llista amb 9 comptadors.
 - Cliquem sobre el comptador 93252350 per introduir la lectura
 - Introduïm 2424.
 - La lectura és òptima.
 - Acceptem la lectura.
 - Lectura acceptada.
 - Acceptem la lectura amb incidència.
 - Introduir incidència.
 - Lectura acceptada.
 - No introduïm incidència.
 - Lectura cancel·lada.
 - Cancel·lem el procés de lectura.
 - Lectura cancel·lada.
 - Cancel·lem la lectura.
 - Lectura cancel·lada.
 - Cancel·lem el procés d’introducció de lectura.
 - -----
 - Introduïm 2300.
 - Lectura inferior a la del trimestre anterior. És volta de comptador?
 - Sí
 - Lectura guardada.
 - Sí, introduint una incidència.
 - Demana que introduïm la incidència.
 - Introduïm incidència.
 - Lectura guardada.
 - No entrem incidència.
 - Lectura cancel·lada.
 - Cancel·lem el procés de lectura.
 - Lectura cancel·lada.
 - No.
 - Acceptem introduint incidència.
 - Introduïm incidència.
 - Lectura guardada.
 - No entrem incidència.
 - Lectura cancel·lada.

- Cancel·lem el procés de lectura.
 - Lectura cancel·lada.
 - Cancel·lem la lectura.
 - Lectura cancel·lada.
 - Cancel·lem el procés d'introducció de lectura.
 - -----
 - Introduïm 2400 o 2500.
 - Lectura amb poc o massa consum.
 - Acceptar la lectura.
 - Lectura guardada.
 - Acceptem introduint incidència.
 - Introduïm incidència.
 - Lectura guardada.
 - No entrem incidència.
 - Lectura cancel·lada.
 - Cancel·lem el procés de lectura.
 - Lectura cancel·lada.
 - Cancel·lem la lectura.
 - Lectura cancel·lada.
 - Cancel·lem el procés d'introducció de lectura.
 - Enviar – Aquí no hem d'entrar cap dada. Ens retorna que ja ha enviat correctament els fitxers.

14. Creació JSON

Com ja hem vist anteriorment necessitarem 5 fitxers json, dos dels quals contenen molts registres i es construeixen a partir de les dades que tenim en el sistema informàtic de l'Ajuntament.

És per això, que he creat un petit software amb Delphi (llenguatge de programació que utilitzo a l'Ajuntament) que permet crear els fitxers vialer.json i comptadors.json.



Il·lustració44: aplicació per crear els fitxers vialer.json i comptadors.json.

Per crear el fitxer vialer.json, simplement es fa una consulta sql al servidor de dades i aquesta la transformem en un fitxer de text en format json.

```
select BDG_VIALER.codcar, BDG_VIALER.descar, (BDG_VIA.cod_aca || ' ' ||BDG_VIALER.descar) complet
FROM BDG_VIALER LEFT JOIN BDG_VIA ON BDG_VIALER.codvia=BDG_via.codvia
WHERE BDG_VIALER.rany='2012' AND BDG_VIA.RANY='2012'
ORDER BY BDG_VIALER.descar
```

```
AssignFile( F, 'c:\temporal\vialer.json' );
Rewrite( F );

WriteLn( F, '{"vialer":' );
WriteLn( F, '[' );

query1.First;
query1.Next;
i:=1;
while not query1.Eof do
begin
    s:='    {"idCarrer":'+inttostr(i)+'', "codiCarrer":""+query1['codcar']+'",
"nomCarrer":""+query1['descar']+'", "nomCarrerVia":""+query1['complet']+'"}';

    query1.Next;
    if not query1.Eof then
        s:=s+',';
    WriteLn( F, s );
    inc(i);
end;

WriteLn( F, ']' );
WriteLn( F, '}' );
CloseFile( F ); CloseFile( F );
```

Per crear el fitxer el fitxer comptadors.json, accedim a la taula amb els valors dels comptadors i creem el fitxer.

```
AssignFile( F, 'c:\temporal\comptadors.json' );
Rewrite( F );

WriteLn( F, '{"comptadors":' );
WriteLn( F, '[' );

table1.First;
table1.Next;
i:=1;
while not table1.Eof do
begin
    if table1['numero']<>null then
        numero:=table1['numero']
    else
        numero:="";
    pis:=comprimeix_pis;
    if table1['lectAnt']<>null then
        lectAnt:=table1['lectAnt']
    else
        lectAnt:=0;

    calcul_min_max;
    s:='    {"nServei":'+inttostr(table1['polica'])+'', '+
"nComptador":""+table1['contador']+'", '+
"nomCarrer":""+table1['nom_carrer']+'", '+
"codiCarrer":""+table1['codcar']+'", '+
"numero":""+numero+'", '+
"pis":""+pis+'", '+
"lecturaAnterior":""+inttostr(lectAnt)+'", '+
"minim":""+inttostr(min)+'", '+
"maxim":""+inttostr(max)+'", '+
"llegit":0}';
```

```

table1.Next;
if not table1.Eof then
  s:=s+'';
  WriteLn( F, s );
  inc(i);
end;

WriteLn( F, ' ]' );
WriteLn( F, '}' );
CloseFile( F );

```

Per calcular el màxim i el mínim, s'obren les taules dels 8 trimestres anteriors i es fa la mitjana del consum. Després es calcula el 25%, per damunt i per sobre, i s'aplica a la lectura anterior.

S'ha de tenir en compte que, de vegades, no hi ha lectura, ja que no s'ha pogut accedir al comptador en aquell trimestre.

```

procedure TForm2.calcul_min_max;
var lectura, lecturaInicial, i, consum, variacio:integer;
begin
  consum:=0;
  if table1['lectant']<>null then
    lectura:=table1['lectant']
  else
    lectura:=0;
  lecturaInicial:=lectura;
  i:=0;
  if s1.Locate('polica',table1['polica'],[]) then
    begin
      if (s1['lectant']<> lecturaInicial) and (s1['lectant']<>null) then
        begin
          lecturaInicial:=s1['lectant'];
          i:=1;
        end;
    end;
  if s2.Locate('polica',table1['polica'],[]) then
    begin
      if (s2['lectant']<> lecturaInicial) and (s2['lectant']<>null) then
        begin
          lecturaInicial:=s2['lectant'];
          i:=2;
        end;
    end;
  if s3.Locate('polica',table1['polica'],[]) then
    begin
      if (s3['lectant']<> lecturaInicial) and (s3['lectant']<>null) then
        begin
          lecturaInicial:=s3['lectant'];
          i:=3;
        end;
    end;
  if s4.Locate('polica',table1['polica'],[]) then
    begin
      if (s4['lectant']<> lecturaInicial) and (s4['lectant']<>null) then
        begin
          lecturaInicial:=s4['lectant'];
          i:=4;
        end;
    end;
  if s5.Locate('polica',table1['polica'],[]) then
    begin
      if (s5['lectant']<> lecturaInicial) and (s5['lectant']<>null) then
        begin
          lecturaInicial:=s5['lectant'];
          i:=5;
        end;
    end;
end;

```



```
if s6.Locate('polica',table1['polica'],[]) then
begin
  if (s6['lectant']<> lecturaInicial) and (s6['lectant']<>null) then
  begin
    lecturaInicial:=s6['lectant'];
    i:=6;
  end;
end;
if s7.Locate('polica',table1['polica'],[]) then
begin
  if (s7['lectant']<> lecturaInicial) and (s7['lectant']<>null) then
  begin
    lecturaInicial:=s7['lectant'];
    i:=7;
  end;
end;
if s8.Locate('polica',table1['polica'],[]) then
begin
  if (s8['lectant']<> lecturaInicial) and (s8['lectant']<>null) then
  begin
    lecturaInicial:=s8['lectant'];
    i:=8;
  end;
end;

if lectura=lecturaInicial then
begin
  min:=0;
  max:=99999999;
end
else if lectura>lecturaInicial then
begin
  consum:=lectura-lecturaInicial;
  consum:=trunc(consum/i)+1;
  variacio:=trunc(consum*0.25)+1;
  min:=lectura+(consum-variacio);
  max:=lectura+(consum+variacio);
end
else
begin
  min:=0;
  max:=99999999;
end;
end;
```

15. PORTAFOLI CRONOLÒGIC DEL DESENVOLUPAMENT DEL PROJECTE

Estudi i desenvolupament d'un servidor web amb node.js

Durada: 8 dies

La primera tasca que he dut a terme és la de conèixer i familiaritzar-me amb la programació d'un servidor amb Node.js.

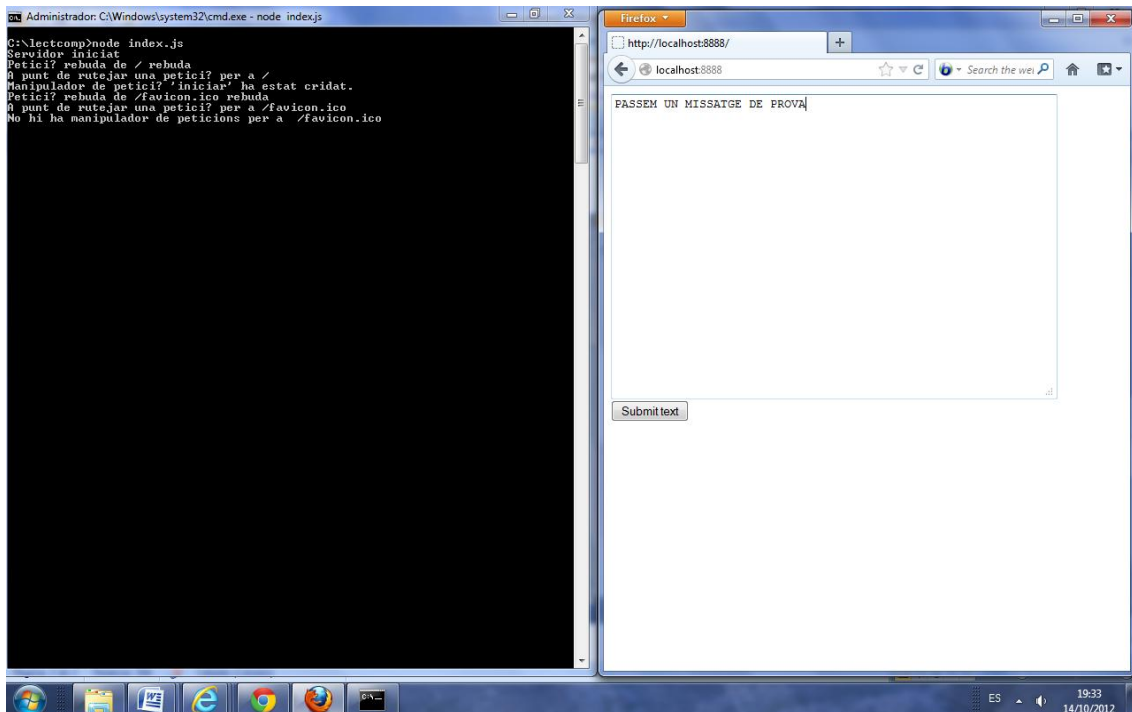
Node.js és un entorn de programació a la capa del servidor basat en Javascript, amb E/S de dades en una arquitectura orientada a events, i basat en el motor Javascript V8 de Google. Permet crear servidors web senzills i escalables.

Després de documentar-me, he fet un petit servidor web que permet gestionar GET i POSTS des d'un navegador.

El servidor està format per quatre fitxers: index.js, server.js, router.js i requestHandler.js.

El fitxer index.js simplement carrega els fitxers i executa el fitxer server.js.

El fitxer server.js posa en funcionament el servidor i es queda escoltant en el port 8888 totes aquelles peticions que li arribaran. Per tal que el servidor sigui més àgil i no es bloquegi, les dades que es rebran es tractaran per trossos. Així, en cas de rebre un paquet de dades molt gran, no es bloquejarà el servidor.

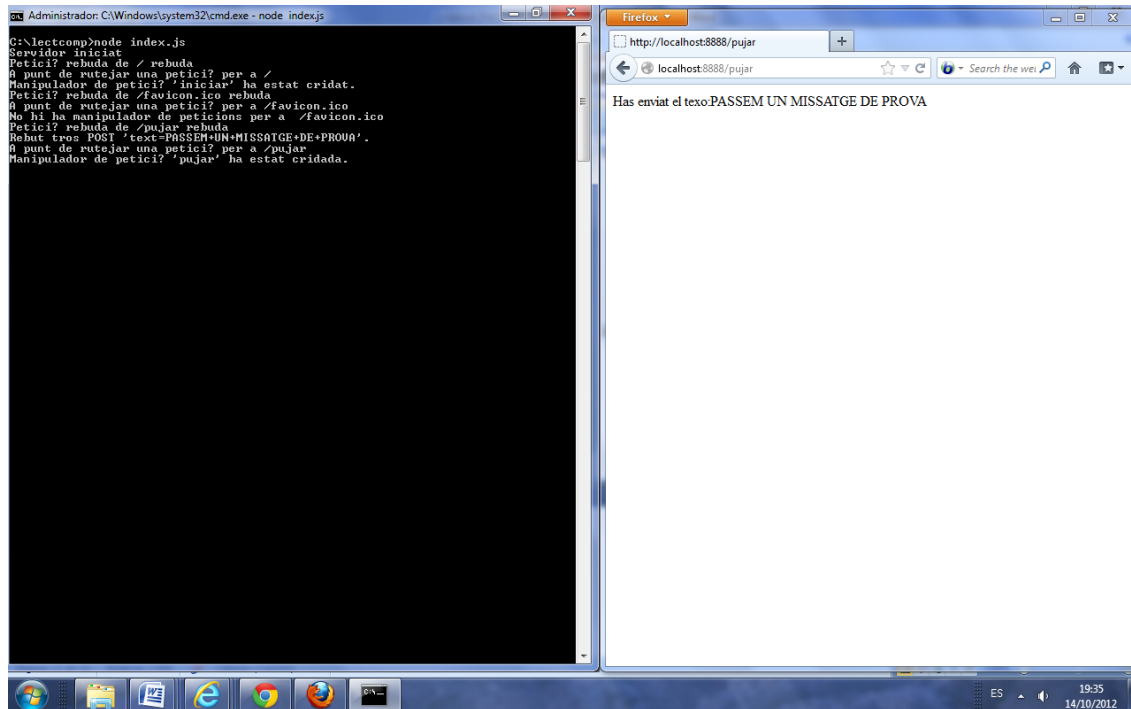


Il·lustració45: captura de pantalla un cop executat el servidor i feta la primera crida.

Un cop rebuda la petició, el servidor la passa a l'enrutador de peticions (router.js) i aquest manejarà aquesta petició amb el manejador de peticions (requestHandlers). S'ha programat d'aquesta manera per tal d'evitar les peticions que puguin bloquejar el servidor.

Node.js pot manejar molts temes concurrents, però no ho fa dividint-ho amb fils (threads). Només fa córrer en un sol fil i va executant un loop d'events. Per tant, hem d'evitar operacions bloquejants, sempre que sigui possible, i utilitzar operacions no-bloquejants en el seu lloc.

Suposem que tenim un petició que el primer que fa és esperar 10 segons per a fer alguna cosa. Com que Node.js només fa córrer un fil, mentre espera els 10 segons, bloquejarà tota la resta de peticions. Per solucionar això, hem programat un manejador de peticions que permet executar peticions concurrents.



Il·lustració46: captura de pantalla després de fer un POST a la pàgina inici.

Estudi i desenvolupament de l'aplicació per a dispositius mòbils

Un cop desenvolupat el servidor web node.js i recollir informació, m'he adonat que el full de ruta que m'he marcat no és el correcte.

Fins ara, en els treballs client-servidor que he realitzat a la Universitat, primer dissenyava la part del servidor, i posteriorment, dedica el temps restant a la part del client.

En el projecte que estic desenvolupant, la part principal, i més important, és la del client. La part del servidor és pot desenvolupar amb eines ja existents, i no és necessari desenvolupar-ne un de nou.

Per tant, he deixat de banda el servidor node.js, i m'he centrat a desenvolupar la part del client.

Lliurament de la PAC2

En aquest lliurament he fet el disseny de totes les pantalles necessàries per poder treballar, així com, mostrar el llistat de carrers i comptadors.

També he desenvolupat els processos per tal de verificar el funcionament correcte de l'anàlisi de les lectures, el qual ens permet saber si la lectura introduïda és òptima o no.

Lliurament de la PAC3

En aquest lliurament s'ha redissenyat totalment la vista de "càrrega". He pres la decisió que aquest procés el dividiré en 4 processos ben diferenciats, per tal de poder executar-los per separat. Aquest 4 processos de carregaran les dades a través de fitxers .json que es troben en un servidor local.

I per últim he dissenyat i programat un controlador d'events. D'aquesta manera, utilitzaré els tres patrons que ofereix el model MVC: els models, les vistes i els controladors.

El controlador que he programat s'executa al carregar l'aplicació, i el que fa és buscar i carregar els fitxers claus.json i trimestre.json que hi ha al servidor. Aquest fitxers són els que condicionen els usos dels menús de l'aplicació.

Lliurament Final

En primer de tot vaig configurar un servidor web en un PC de l'Ajuntament per tal de poder executar l'aplicació. He posat el servidor XAMPP i he modificat la configuració del router per redirigir el port 8082 al PC on hi ha l'aplicació.

L'adreça d'execució serà **`http://216.126.56.102:8082/LECTCOMP/app.html`**

També he creat l'estructura per poder connectar amb el servidor a través del fitxer php.

El fitxer PHP el trobarem **`http://216.126.56.102:8082/LECTCOMP/php/rebre-json1.php`**

He optimitzat l'edició de la llista de carrers per a l'iphone i l'ipad.

I, per últim, he generat tota la documentació per al lliurament final.

El vídeo amb la presentació el podeu trobar a:

<https://vimeo.com/56749639>

He fet un vídeo extra que mostro com he creat els fitxers json amb el llenguatge de programació DELPHI.

<https://vimeo.com/56750302>

16. Conclusions

Aquest projecte m'ha permès començar a introduir-me en el món de la programació en dispositius mòbils.

Aquest camp era totalment desconegut per a mi, i per aquest motiu he hagut de fer una petita introducció en aquest món per tal de situar-me.

Després d'haver treballat amb Sencha he pogut observar que és un llenguatge de programació bastant intuïtiu tan bon punt adquireixes agilitat, però he trobat que li falta documentació i que encara està una mica verd.

També he notat que té molts problemes de rendiment i que hauria de millorar en aquest tema.

Per que fa a l'aplicatiu desenvolupat, crec que m'ha sortit prou bé, tot i que a mesura que el fem servir el podem anar polint per a fer-lo més àgil.

17. Línies de futur

Queda pendent de desenvolupar la part final del servidor, és a dir, agafar el resultat obtingut i passar-lo a json i tractar-lo per tal de passar-lo al nostre sistema.

Durant el propers mesos em dedicaré a desenvolupar aquesta tasca per tal de que el puguem utilitzar a la feina.

També seria interessant poder fer fotos en els casos en que trobem comptadors conflictius o deteriorats per tal de poder analitzar-los posteriorment.

18. Bibliografia

- <http://es.wikipedia.org/>
- <http://www.apachefriends.org/en/xampp-windows.html>
- <http://www.sencha.com/products/touch>
- Projecte de final de carrera de Sílvia Barea Pagan

Node.js

- <http://vicentfernandezicapilla.blogspot.com.es/2012/09/creacio-dun-servidor-simple-amb-nodejs.html>
- <http://cibersocietat.blogspot.com/2012/06/01-curso-de-nodejs-introduccion-nodejs.html>
- http://www.youtube.com/watch?feature=player_embedded&v=vnRGAsZdhx0
- <http://caballe.cat/wp/tag/nodejs/>
- <http://nodejs.org/>

- <http://www.nodebeginner.org/index-es.html>
- <http://www.genbetadev.com/frameworks/introduccion-a-la-programacion-asincrona-con-nodejs>
- <http://blog.koalite.com/2011/11/tutorial-node-js-express-jquery-i-creando-la-aplicacion/>
- Llibre “DESARROLLO DE APLICACIONES EN LA NUBE PARA DISPOSITIVOS MOVILES”
- <http://www.cristalab.com/tutoriales/aplicacion-basica-con-nodejs-express-jade-y-stylus-c101503/>
- Llibre “Online” <http://book.mixu.net/ch5.html>
- <http://illasaron.com/html/content/01-curso-de-nodejs-introducci%C3%B3n-nodejs>
- <http://www.safaribooksonline.com/>
- Sams Teach – Learning node – Node up and running – Building hypermedia
- Sams Teach – Node.js in 24 hours
- <http://www.x10demos.com/pdfnew/>
- <http://fernetjs.com/2012/08/buenos-amigos-nodejs-mongodb/>

Sencha

- <http://miamicoder.com/2012/how-to-create-a-sencha-touch-2-app-part-1/>
- http://www.google.es/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=3&cad=rja&ved=0CDQQFjAC&url=http%3A%2F%2Fmiamicoder.com%2F2012%2Fhow-to-create-a-sencha-touch-2-app-part-2%2F&ei=KQd3UJbIH8SKhQfHkoCADA&usg=AFQjCNGfOSZUMyWWkMxP6QuXTBk79zkr7w&sig2=CRqFJDrNesBxShe7_UJ2yg
- <http://www.youtube.com/watch?v=SJaReMmUJO4&feature=relmfu>
- <http://www.justinmind.com>
- <http://www.youtube.com/watch?v=XvWGZO0f6cQ>
- <http://docs.sencha.com/touch/2-0/http://miamicoder.com/2012/how-to-create-a-sencha-touch-2-app-part-2/>
- <http://docs.sencha.com/touch/2-0/#!/guide/forms>
- <http://www.mysamplecode.com/2012/08/sencha-touch-search-field-example.html>

19. ANNEX 1 – SERVIDOR NODE.JS

INDEX.JS

```
var server = require("./server");
var router = require("./router");
var requestHandlers = require("./requestHandlers");

handle={}
handle["/"]=requestHandlers.iniciar;
handle["/iniciar"]=requestHandlers.iniciar;
handle["/pujar"]=requestHandlers.pujar;

server.iniciar(router.route,handle);
```

SERVER.JS

```
var http = require("http");
var url = require("url");

function iniciar(route,handle){
    function onRequest(request, response) {
        var pathname = url.parse(request.url).pathname;
        var dataPostejada = "";
        console.log("Petició rebuda de " + pathname + " rebuda");

        request.setEncoding("utf8");
        request.addListener("data", function(trosPostejat) {
            dataPostejada += trosPostejat;
            console.log("Rebut tros POST " + trosPostejat + ".");
        });

        request.addListener("end", function() {
            route(handle, pathname, response, dataPostejada);
        });
    }

    http.createServer(onRequest).listen(8888);

    console.log("Servidor iniciat");
}

exports.iniciar = iniciar;
```

ROUTER.JS

```
function route (handle,pathname,response,postData){
    console.log("A punt de rutejar una petició per a "+ pathname);
    if (typeof handle[pathname]=== 'function'){
        handle[pathname](response,postData);
    }
}
```

```
        } else {
            console.log("No hi ha manipulador de peticions per a "+pathname);
            response.writeHead(404,{"Content-Type":"text/html"});
            response.write("404 No trobada");
            response.end();
        }
    }

    exports.route = route;
```

REQUESTHANDLERS.JS

```
var querystring = require("querystring");

function iniciar(response,postData){
    console.log("Manipulador de petició 'iniciar' ha estat cridat.");

    var body = '<html>'+
        '<head>'+
        '<meta http-equiv="Content-Type" content="text/html;charset=UTF-8" />'+
        '</head>'+
        '<body>'+
        '<form action="/pujar" method="post">'+
        '<textarea name="text" rows="20" cols="60"></textarea>'+
        '<input type="submit" value="Submit text" />'+
        '</form>'+
        '</body>'+
        '</html>';

    response.writeHead(200,{"Content-Type":"text/html"});
    response.write(body);
    response.end();
}

function pujar(response,dataPostejada){
    console.log("Manipulador de petició 'pujar' ha estat cridada.");
    response.writeHead(200,{"Content-Type":"text/html"});
    response.write("Has enviat el texo:" +
    querystring.parse(dataPostejada)["text"]);
    response.end();
}

exports.iniciar = iniciar;
exports.pujar = pujar;
```