

VIRTUAL

.DIBUIXOS DE CIRCUITS ELECTRÒNICS

. VÍDEOS DE ROBÒTICA APLICADA A LES AULES

MEMÒRIA DEL PROJECTE FINAL DE CARRERA

**Alumna: Maria Montserrat Areñas Vila
Consultor: Carlos Sánchez Rosa**

Aula1: PFC-Programació d'aplicacions per a mòbils usant HTML5



Índex

1. PLA DEL PROJECTE	3
1.1. Descripció del PFC	4
1.2. Metodologia	6
1.3. Temporitzaació detallada: Diagrama de Gannt	7
2. Entorn de treball	9
2.1.1. Alumnes de tercer d'ESO per la part d'electrònica	9
2.1.2. Alumnes de quart d'ESO per la part de robòtica	9
2.2. Material de suport. Procediment:	10
2.2.1. Anàlisi i primeres proves de l'aplicació	10
2.2.2. Creació de l'arxiu manifest.webapp	10
2.3. Anàlisi funcional	15
2.4. Disseny de pantalles	17
El primer que l'usuari trobarà de l'aplicació serà el logo de l'aplicació : Una xinxeta.	17
2.5. Anàlisi tècnic	21
2.5.1. Implementació	22
2.6. Creació de l'aplicació (II)	27
2.6.1. Implementació de la pantalla robòtica	27
2.6.1.1. Resultats inicials: n Pantalles en blanc	27
2.6.2.2. Resultats finals: Possibilitats de publicar els vídeos.	31
2.7. Publicació de l'aplicació	34
3. Vídeo presentació de la feina feta.	36
4. Cloenda	36
Annex: Referències Bibliogràfiques	38



1. PLA DEL PROJECTE

El projecte que he dut a terme té com a finalitat crear una aplicació web educativa que permeti a qualsevol usuari poder realitzar tasques eficients a partir d'un simulador; tasques que després es poden imprimir, analitzar, i millorar si convé. Per una banda permet realitzar un número infinit d'esquemes de circuits electrònics i per altra banda presenta un número ampliable de models bàsics de robots senzills que realitzen diferents activitats.

Els usuaris, si algun dia es pogués utilitzar l'aplicació a partir del mòbil, podrien realitzar esquemes de circuits electrònics de manera sistemàtica, ordenada i concreta. De moment, és permet la descàrrega en mòbils amb el sistema Firefox OS és possible des del **Marketplace**. Aquests esquemes realitzats amb el simulador i posteriorment en el mòbil ZTE o altres sistemes que presentin aquest sistema, poden ser funcionals o no, aleshores es tracta de posar en comú la viabilitat de l'esquema a partir del diàleg obert entre els mateixos usuaris o bé, si és el cas, fent servir les xarxes socials tipus **WhatsApp**, **Twitter** o **Facebook**.

L'aplicació facilita d'elements movibles necessaris per a realitzar els esquemes electrònics. El número d'elements, en alguns casos, no s'exhaureix i per tant es poden fer esquemes prou complicats, es tracta però de comprovar-ne la seva viabilitat i d'acostumar els usuaris a ser polits, concrets i precisos a l'hora de fer el disseny d'un sistema i alhora de revisar, repassar i comprovar que efectivament altres persones també el troben útil.

Els usuaris es poden trobar en que no tenen prou estabilitat, serenor i control dels elements presentats. Si és aquest el cas, l'aplicació desapareix de la pantalla. Evidentment és possible recuperar l'aplicació, però està pensat per fer desaparèixer tot el realitzat, i si es vol continuar cal tornar a començar. Aleshores l'aplicació serveix també per a treballar la motricitat fina, la concentració i la precisió, a més a més de ser útil per a tots els usuaris (estudiants o afeccionats a l'electrònica i/o a la robòtica).



1.1. Descripció del PFC

A partir de totes les lectures realitzades per conèixer i descobrir les característiques i avantatges del llenguatge HTML5 he pogut realitzar l'aplicació que presento. En principi és una aplicació amb dues opcions, dissenyada per a consolidar

- a) coneixements d'electrònica i
- b) de robòtica bàsica.

L'avantatge principal d'aquesta aplicació és el fet que motiva positivament els alumnes a participar i a comprovar que efectivament el que estan practicant en el laboratori o en el taller és correcte, o potser no del tot. En poden parlar i a més a més comparar resultats des de la observació del que han fet, de moment bàsicament a l'ordinador. Suposadament més endavant podran fer proves a amb el mòbil a qualsevol hora del dia, i a partir de la pràctica es demostra que efectivament milloren en la comprensió i domini de vocabulari i de conceptes com ara elements bàsics per a la instal·lació o connexió en sèrie, en derivació i millora en el disseny d'instal·lacions d'equipaments electrònics. Igualment poden optar per veure tot el procés de creació dels robots, creats a l'aula per mitjà de la filmació, editar els vídeos i pujar-los a l'aplicació.

El projecte l'anomeno **virtual** per el fet que els resultats de les tasques destacades que han realitzat els alumnes, i jo mateixa, han estat recollides en format vídeo i per tant es poden reproduir per a estudiar el cas i com a conseqüència revisar utilitzant elements multimèdia.

1.1.1. Passos seguits:

- a) Disseny clar d'una pantalla menú amb dues opcions: electrònica / robòtica.
- b) Elaboració i elecció d'icones que permetin relacionar la imatge amb la temàtica.
- c) Disseny de l'aplicació (jocs):
 - 1- connexió directe des del simulador, o des del telèfon mòbil quan es pugui, per a la realització de circuits en sèrie, paral·lel i mixt amb diferents elements.
 - 2- Visualitzar el moviment dels robots acabats en les filmacions.
- d) Aplicació web i simulació del funcionament.
- e) Redacció de materials i gravació de resultats simultània amb el progrés realitzat en l'elaboració del treball.
- f) Realització del vídeo de presentació del treball.



- g) Edició el vídeo a partir de subtítols significatius i música.
- h) Edició de tot el material.
- i) Comprovació del funcionament del material elaborat tenint en compte els comentaris dels alumnes.
- j) Redacció final de la memòria i entrega de tot el material.

1.1.2. Temporització global:

<p>Fase 1</p> <p>Familiarització</p> <p>26-febrer / 12-març</p>	<p>Introducció i estudi del llenguatge HTML5 i dels diferents dispositius per a implementar l'aplicació. Lectura de textos i observació de vídeos relacionats amb el treball.</p> <p>Elecció de tema i dispositius necessaris.</p> <p>Redacció de l'esborrany del projecte i presentació del pla de treball del projecte.</p> <p>Entrega de la PAC1.</p>
<p>Fase 2</p> <p>Simulacions</p> <p>13 març/ 9 -abril</p>	<p>Inici i estructuració en carpetes CSS, imatges, JS i altres de l'aplicació web de l'aplicació pensada.</p> <p>Estudi del FIREFOX OS, consolidació del llenguatge HTML5 i realització de proves inicials carregant el simulador.</p> <p>Connexió amb pàgines web externes de funcionament de simulacions d'equipament electrònic.</p> <p>Reestructuració del pla de treball si es necessari.</p> <p>Contacte amb el/s consultor/s i seguiment de les indicacions donades.</p> <p>Entrega de la PAC2.</p>
<p>Fase 3</p> <p>Gravació de resultats</p> <p>10 abril / 7 maig</p>	<p>Gravació de materials, presa d'imatges i realització de proves.</p> <p>Elaboració dels fitxers necessaris per a poder fer una simulació.</p> <p>Actualització del pla de treball i redacció inicial tant dels assoliments com de les dificultats observades per poder-les plasmar en una memòria.</p> <p>Entrega de la PAC3.</p>
<p>Fase 4</p> <p>Implementació i Cloenda</p> <p>8 maig / 4 de juny</p>	<p>Revisió d'errors si n'hi ha, rectificació de materials.</p> <p>Realització de simulacions .</p> <p>Producció final d'una aplicació funcional.</p> <p>Monitorització del codi font .</p> <p>Muntatge d'un vídeo virtual explicatiu de la feina feta.</p> <p>Redacció de la memòria i entrega de resultats.</p> <p>Entrega del projecte final.</p>

1.1.3. Objectius d'aprenentatge



1.1.3.1.Objectius generals:

primer:

Realitzar una aplicació didàctica per a dispositius mòbils utilitzant tecnologies web (HTML5, CSS3 i JavaScript), a partir d'un simulador , de manera que el resultat sigui aplicable a telèfons mòbils d'última generació.

segon:

Crear una aplicació mòbil basada en FIREFOX OS relacionada amb l'assignatura de tecnologia educativa per tal de poder-la posar a prova amb alumnes d'ESO.

tercer:

Crear una aplicació web educativa que permeti a qualsevol usuari dibuixar circuits electrònics de forma esquemàtica.

1.1.3.2. Objectius específics:

- Conèixer el funcionament del llenguatge HTML5.
- Saber utilitzar el simulador FIREFOX OS, pujar-hi aplicacions i comprovar que funcionen.
- Realitzar una aplicació funcional interactiva útil per als alumnes d'ESO, a partir de necessitats observades a l'aula de tecnologia, utilitzant una pàgines web externes a través d'Internet i el simulador FIREFOX OS.
- Carregar l'aplicació dissenyada de manera que es pugui utilitzar en els mòbils que tinguin instal·lada aquesta tecnologia.
- Comprovar que l'aplicació funciona i observar la reacció dels alumnes.
- Realitzar un vídeo explicatiu resum de la feina realitzada.
- Superar el treball del projecte de final de carrera.

1.2. Metodologia



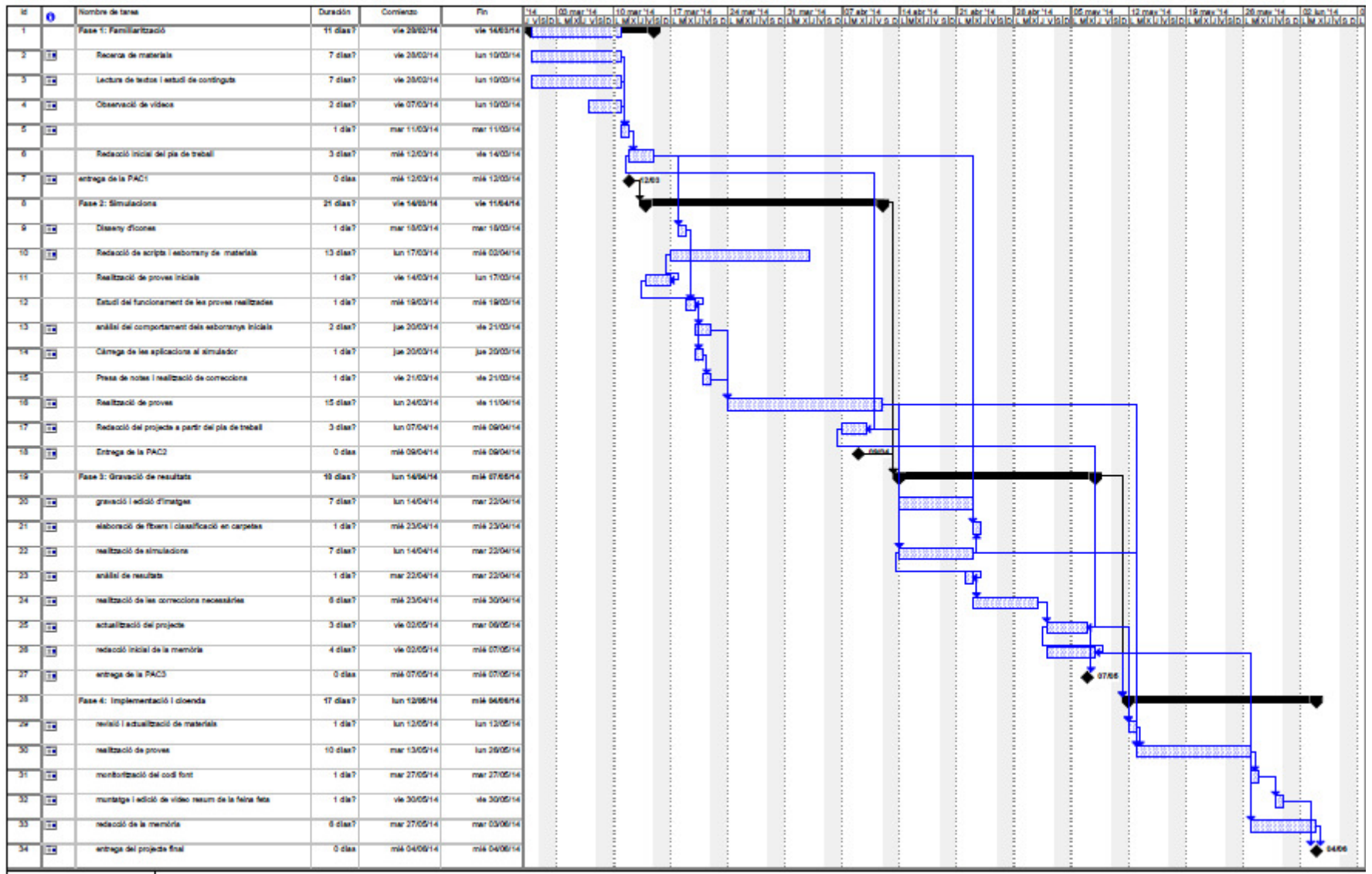
Per a aconseguir resultats cal ser ordenat, precís i seguir el pla de treball previst ja des del començament del projecte. D'aquesta manera, a partir de l'organització pròpia plantejada per a superar el projecte final de carrera és qüestió d'anar omplint els buits que es noten i anar aconseguint fites puntuals que podien resultar efectives inicialment, o potser no i que podien fins i tot implicar haver de treballar més i de dedicar-hi més hores de les previstes, pel fet de interactuar amb materials i tecnologia a l'abast però inicialment totalment nova per a mi.

Un cop conegut el tema de treball cal conèixer primer de tot, el llenguatge HTML5, buscar exemples i veure com resoldre problemes que precisament per el fet d'haver plantejat una determinada tasca sense saber si era possible realitzar-la en el marc treball que es plantejava, fàcilment podria haver resultat una fita molt complicada. A continuació se'm va fer palès que calia, necessitava, la col·laboració per part dels alumnes i que essent que jo sabia poc del que tenia entre mans i els alumnes encara menys, aleshores calia preparar-se bé, llegir molt i interpretar molt més, trobar més material i anar avançant. Val a dir que els alumnes han manifestat interès i tot ha estat tema de motivar-los, donar-los-hi la possibilitat de treballar, de criticar, cooperar i poder intervenir perquè col·laboressin gaudint amb el què feien. Al final tots els alumnes han exposat el seu interès i la conveniència de continuar treballant en la via experimental paral·lelament amb la teoria que en totes les classes de tecnologia és una de les parts fonamentals.

Plantejat el treball, la metodologia emprada va ésser:

- llegir, conèixer i assimilar la teoria.
- posar en pràctica el que els articles deien que funcionava. Si hi havia dificultats seguir buscant, pensar, llegir, provar, modificar i consolidar l'après. Si alguna cosa definitivament trobava que funcionava bé anar avançant.
- plantejar als alumnes el desenvolupament del material que inicialment semblava funcionar donant-los-hi la oportunitat de fer-se'l seu. Així en el tema dels robots cada grup va optar per aquell aspecte que li interessava treballar; fer un disseny propi i fer-ho tot al taller de tecnologia compartint les dificultats trobades entre tots.
- utilitzar els consells del consultor si arribava a un punt mort.
- seguir treballant a bon ritme per a poder entregar tots els resultats parcials i també el final el dia acordat.

1.3. Temporització detallada: Diagrama de Gannt



Proyecto: Project1
 Fecha: lun 10/03/14

Tarea: 0000000000 Progreso: Resumen: Tareas externas: Fecha límite: ↓

Hitos: Resumen del proyecto: Hitos externos:



2. Entorn de treball

2.1. Alumnes d'ESO de l'Institut Eugeni d'Ors de Badalona

Donat el temps limitat amb que es compta des que es coneix l'aspecte sobre el que es treballa en el projecte de final de carrera, la tasca que he hagut de realitzar, tot i ser totalment satisfactòria també ha resultat un pèl complexa. En el tema de robòtica no hi havia dubtes, però en el tema de dibuixar a partir del simulador. no només m'havia de reciclar jo personalment, sinó que essencialment al mateix temps que jo aprenia com anava tot plegat, feia de formadora dels alumnes sobre el mateix tema i a més a més, l'entorn de treball havia de ser de confiança, de cooperació i de domini del que estàvem fent.

Essent que la meva feina em permet tenir contacte amb un número elevat d'alumnes i que precisava usuaris efectius per a dur a terme les proves de l'aplicació dissenyada, vaig optar per consultar a les classes corresponents i sol·licitar-els-hi a veure quina opinió els mereixia la feina que els estava plantejant. La resposta va ser bona i a partir del moment en què els alumnes van accedir a col·laborar vam quedar que els meus alumnes de tercer d'ESO s'encarregarien de la part d'electrònica i els de quart de la part de robòtica.

2.1.1. Alumnes de tercer d'ESO per la part d'electrònica

La tasca encomanada a aquests alumnes ha estat de provar, criticar i intentar utilitzar el simulador amb els elements de dibuix per a construir circuits. La meva tasca ha estat presentar el material, fer possible l'aplicació, pensar quins elements eren els més indicats i buscar com col·locar cada element al seu lloc. Si calia a més a més poder allargar el fil i copiar els elements, tants com en necessitessin. Els alumnes haurien de comentar-me les possibilitats trobades, la utilitat i la dificultat. Per exemple, si algun alumne construïa un circuit sense bateria, o sense resistències, o simplement sense polsador, els alumnes ja ho notaven, en feien burla i per tant indicava que el resultat d'utilitzar el simulador era positiu per a l'aprenentatge.

2.1.2. Alumnes de quart d'ESO per la part de robòtica

Els alumnes de quart han hagut d'aprendre a negociar materials i preus, compartir coneixements entre els companys, instal·lar components electrònics, de programar i de dissenyar cobertes per els robots que posteriorment haurien de construir. La feina ha estat complexa però els resultats són afalagadors. Aquí la meva tasca ha consistit en orientar i monitoritzar les aportacions, acceptar algunes propostes i rebutjar-ne d'altres justificant les decisions preses. Un cop els alumnes van realitzar la tasca vaig procedir a gravar els resultats i a penjar els materials a la xarxa. Val a dir que tot i que els materials es poden veure en xarxa, l'aplicació del mòbil s'ha pogut publicar i es pot



descarregar de forma gratuïta al **Marketplace** (<https://marketplace.firefox.com/app/avirtual/>).

Per altra banda si utilitzés un servidor web, precisaria generar diferents arxius- entre cinc i sis empaquetats- per tal que aquest servidor em pogués reconèixer l'aplicació. Com que el temps és el que és i els resultats són prou bons, presento l'aconseguit fins ara.

2.2. Material de suport. Procediment:

2.2.1. Anàlisi i primeres proves de l'aplicació

El primer que calia fer era familiaritzar-me amb el sistema operatiu Firefox OS, sobre tot la part de crear el fitxer manifest. webapp, per tal de poder realitzar les simulacions de l'aplicació creada. A més a més, continuar amb el llenguatge HTML5, estudi dels jQuery i Kineticsjs.

2.2.2. Creació de l'arxiu manifest.webapp

Per poder assolir aquesta fita el que vaig fer és previsualitzar el vídeo "The Web Apps Manifest - Firefox OS Quick Tip" que hi ha al document online :

https://developer.mozilla.org/en-US/Apps/Quickstart/Build/Your_first_app

Per generar el fitxer manifest vaig utilitzar el software Notepad++ i introduir el codi següent:

```
{
  "name": "Virtual",
  "description": "Ejemplo básico de web app",
  "launch_path": "/index.html",
  "icons": {
    "64": "/imatges/pv-icon.png",
    "128": "/imatges/pv-icon.png"
  }
}
```

Aquest fitxer permet que s'instal·li l'aplicació i al mateix temps ens mostri la icona seleccionada.

La icona esta creada per mitjà del software **Fireworks CS5**, que va proporcionar-me la UOC en acabar el grau multimèdia, del paquet denominat **AdobeMasterCollectionCS5**. Vaig buscar una icona que pugues relacionar amb el que volia fer, vaig fer una col·lecció de mostres i al final la icona triada va ser una xinxeta, per el fet que permetia recordar que l'aplicació va de connectar components i poder observar el resultat final per mitjà d'una gravació del seu funcionament.

Dins de la figura de la xinxeta incorpo les inicials de l'aplicació . El resultat és el logo de la figura 1:



Fig. 1 :Resultat obtingut i logo de l'aplicació

La figura denominada pv-icon és la que el fitxer manifest crida per mitjà del codi:

```
"icon":{  
"60": "/imatges/pv-icon.png",  
"128": "/imatges/pv-icon.png"  
}
```

A continuació cal codificar l'índex de l'aplicació i anar omplint el contingut. Per fer-ho vaig utilitzar el **Dreamweaver (DW)**, per ser un programa que ja havia utilitzat per a la creació de pàgines web i CSS. És un programa que permet incorporar les etiquetes i que contempla les possibilitats que dona **HTML5, CSS3 i javascript**.

El primer resultat va ser:



Fig. 2: disseny provisional

A continuació, calia instal·lar tot l'efectuat al simulador de Firefox. Per fer-ho calia insertar al buscador **Google** el simulador **Firefox OS**. Un cop instal·lat ja vam poder realitzar proves per veure si el manifest ens instal·lava l'aplicació en el simulador del mòbil.

Primer, per conèixer el seu funcionament, vaig carregar una aplicació de mostra d'**Internet**. Per fer-ho, cal fer clic al botó **add** i localitzar l'aplicació que és vol carregar, indicant-li on es troba el manifest de l'aplicació.

Les aplicacions de mostra funcionaven bé, ara quan vaig realitzar les de la meua simulació de l'aplicació em van sorgir diferents problemes que vaig haver d'anar



superant tot interpretant els diferents missatges que anaven sortint i que, per tant, anava modificant.

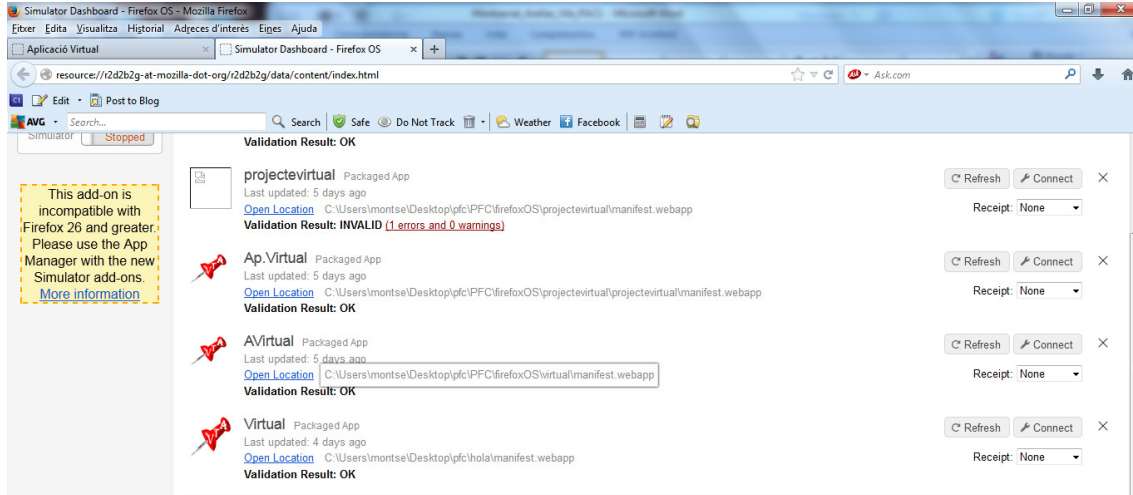


Fig. 3: inici proves

La primera de tot em va mostrar un error i la imatge no es va carregar. Vaig tornar a llegir quines condicions s'havien de complir i resulta que el nom de l'aplicació era massa llarg. El següent pas va ser modificar el nom, i en aquest cas es va carregar, però al fer clic en el mòbil buscant la icona resulta que no la trobava, vaig pensar que era el punt que havia posat, per tant vaig tornar a canviar el nom. Resultat : es carrega, però no torno a trobar l'aplicació. És a dir, alguna cosa havia de millorar. Vaig continuar fent proves i anar variant detalls. Moltes simulacions.

A l'última simulació el que vaig fer va ser crear un índex sense incorporar les etiquetes **header**, **seccion**, **footer** o millor dit utilitzar-les com a id de les div, com per exemple:

```
<div id="footer"><small> Copyright 2014 Virtual | M.MontserratAreñas Vila | UOC - PFC </small></div>
```

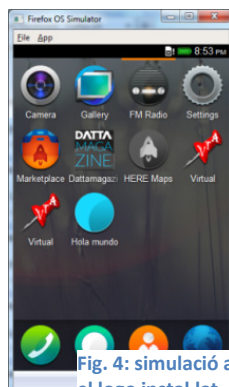


Fig. 4: simulació amb el logo instal·lat

Realitzo una nova simulació i el que observo que és carrega l'aplicació. Trobo la icona i em mostra les possibilitats que tinc.

Al veure la icona em va semblar que era un nou error. Vaig tornar-la a carregar i torno a observar que s'ha instal·lat novament. És a dir, ara sortia dues vegades. A continuació comprovo què mostra la pàgina index.html.

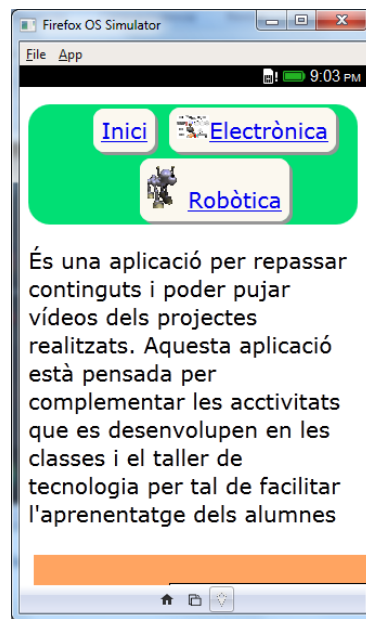


Fig. 5: Resultat de la instal·lació de l'aplicació en el simulador
L'aplicació comença a tenir forma.
L'estructura de l'aplicació tindrà una carpeta per les imatges, una pels estils i una altra per javascript.

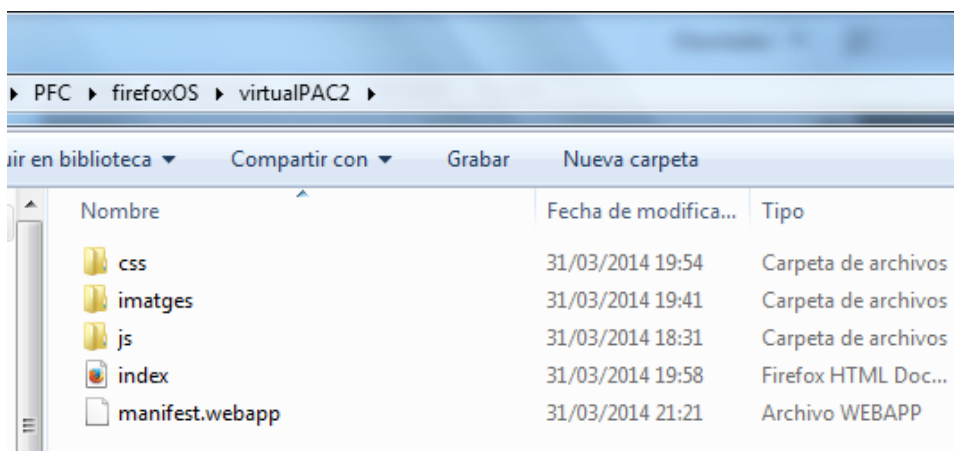


Fig. 6: Estructura de l'aplicació

A continuació començo a generar el codi per poder dibuixar, és a dir, incloc l'etiqueta canvas, i també genero el fitxer js que permetrà dibuixar.

Exposo el cas al tutor Sr. Carles Sànchez Rosa, i m'indica un treball que també va utilitzar **canvas**. Al llegir el treball observo que utilitza el **framework jQuery Mobile** i una llibreria **Kineticjs**. Al no conèixer el framework jQuery Mobile ni la llibreria busco manual o tutorials que em permetin familiaritzar-me amb els continguts i avançar en el projecte i continuo treballant amb el projecte recomanat.

Al introduir-me en el **jQuery**, començo a veure que és poden realitzar millores en la navegació i començo a modificar l'aplicació per obtenir un aspecte millorat.

Per poder millorar l'aspecte cal carregar o fer un link a arxius **jQuery Mobile**, tant css com js. També vaig creat un full d'estils per poder acabar de modificar el que em convingués segons el meu criteri i les necessitats que es presentessin.

Modifico l'índex utilitzant atributs de **jQuery** i torno a realitzar una nova simulació.

El resultat és que el manifest m'ha carregat la nova versió de l'aplicació anomenada **AVirtual**:

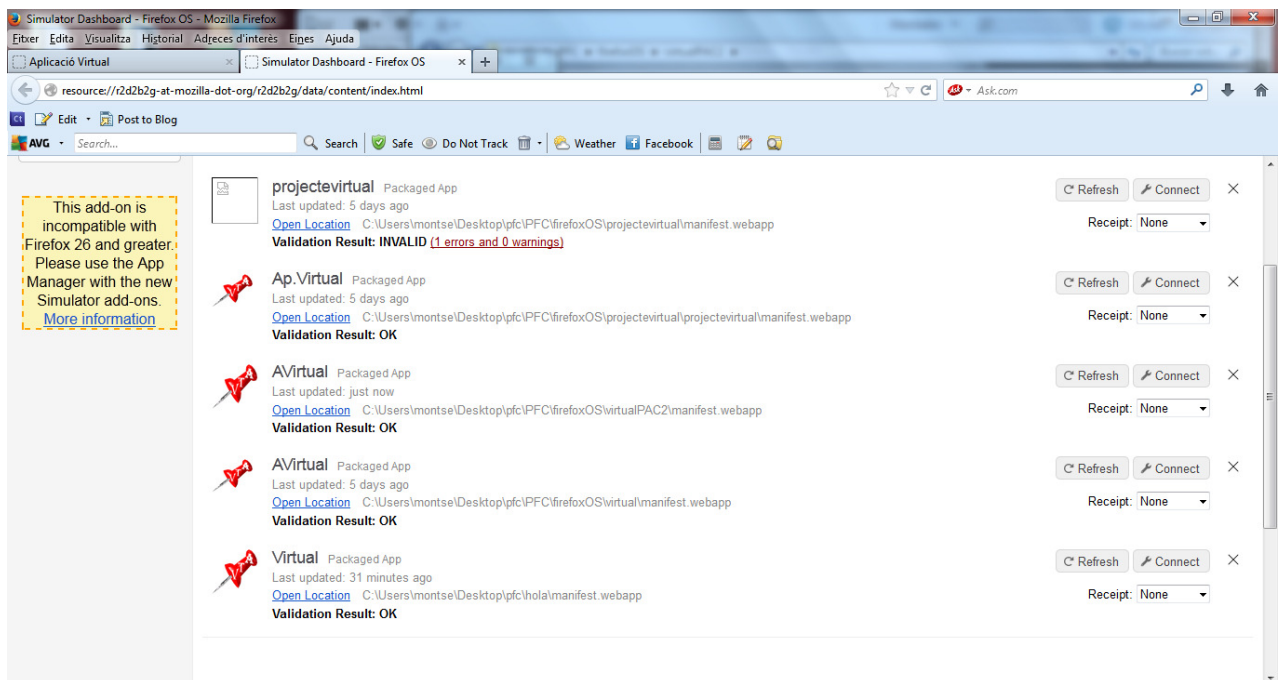


Fig. 7: Resultat de la nova aplicació

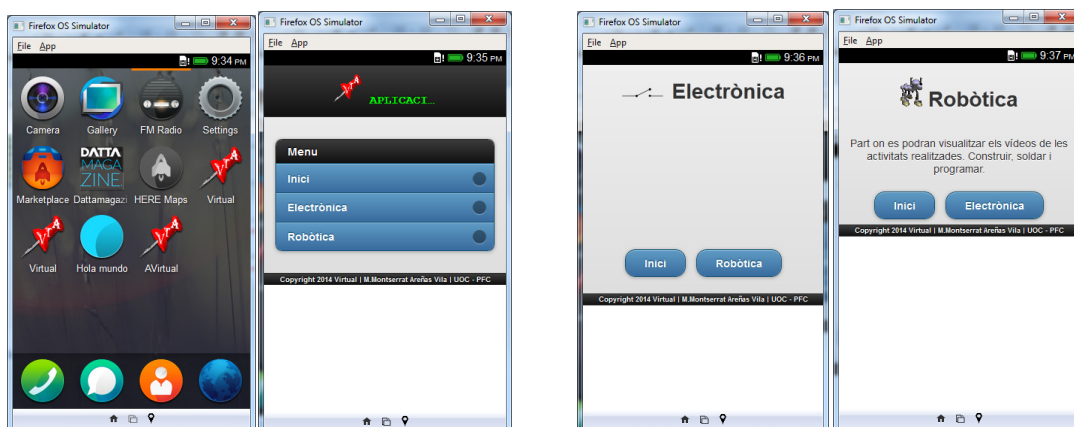


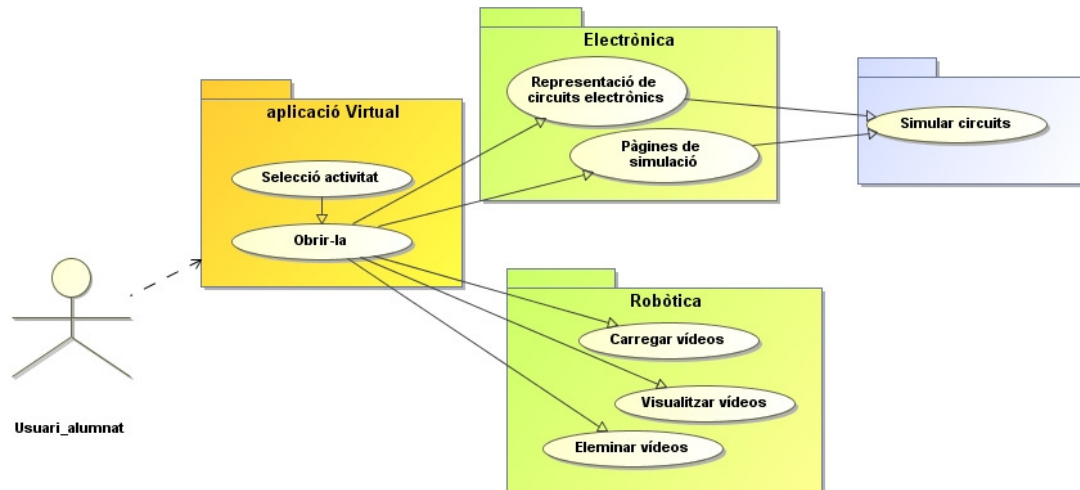
Fig. 8: Vistes del resultat obtingut utilitzant atributs del jQuery Mobile

Verifico que el **manifest.webapp** funciona correctament. Dono com a bo el resultat obtingut.

2.3. Anàlisi funcional

En aquest punt defineixo el que, a la pràctica pretenc que pugui fer qualsevol usuari, concretament i específica, l'alumnat, que faci servir l'aplicació Virtual.

Fig. 9: El què pot fer l'usuari



Segons el diagrama de casos d'ús, podríem dividir –ho en tres parts o casos:

- Primer cas: Selecció de l'activitat a realitzar.
- Segon cas: Hi trobem el que poden fer en cada una de les parts de l'aplicació.
- Tercer cas: l'usuari pot dibuixar i realitzar les diferents simulacions dels circuits elèctrics.

a) Definició dels casos

a.1) Primer cas:

Cas d'ús	Seleccionar activitat
Actors	Alumnat 4t ESO
Tipus	Generalització
Propòsit	Escollir un tema (electrònica o robòtica)
Resum	El cas d'ús defineix l'acció de seleccionar una activitat.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació → Seleccionar → Activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

Cas d'ús	Obrir activitat
Actors	Alumnat 4t ESO
Tipus	Generalització



Propòsit	Realitzar l'activitat escollida en l'aplicació.
Resum	El cas d'ús defineix el que pot fer quan estem dins d'una activitat.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació → Seleccionar → Obrir activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

a.2) Segon cas:

Cas d'ús	Realització de circuits electrònics
Actors	Alumnat 3r ESO
Tipus	Bàsic
Propòsit	Simulació de la representació dels circuits.
Resum	El cas d'ús defineix l'acció de seleccionar una activitat.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació → Seleccionar → Realitzar activitat
Subflux	Entrar o obrir activitat.
Excepcions	No n'hi ha.

Cas d'ús	Pàgines de simulació
Actors	Alumnat 3r ESO
Tipus	Bàsic
Propòsit	Escollir pàgina de simulacions.
Resum	El cas d'ús defineix l'acció de seleccionar pàgines web de simuladors.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació → Seleccionar → Realitzar activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

a.3) Tercer cas:

Cas d'ús	Carregar vídeos
Actors	Alumnat 4t ESO
Tipus	Bàsic
Propòsit	Pujar els vídeos o imatges dels projectes realitzats.
Resum	El cas d'ús defineix l'acció de carregar fitxers.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació → Seleccionar → Realitzar activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

Cas d'ús	Eliminar vídeos
Actors	Alumnat 4t ESO
Tipus	Bàsic

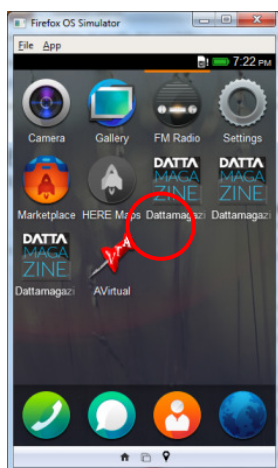
Propòsit	Esborrar els fitxers antics.
Resum	El cas d'ús defineix l'acció de seleccionar una activitat.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació► Seleccionar► Realitzar activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

Cas d'ús	Visualitzar vídeo
Actors	Alumnat 4t ESO
Tipus	Generalització
Propòsit	Veure els projectes dels companys
Resum	El cas d'ús defineix l'acció de visualitzar fitxers.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació► Seleccionar► Realitzar activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

Cas d'ús	Simular circuits
Actors	Alumnat 3r ESO
Tipus	Generalització
Propòsit	Utilitzar diferents simuladors de circuits.
Resum	El cas d'ús defineix l'acció de dibuixar circuits.
Precondicions	Cal que existeixi una activitat seleccionada per executar el cas d'ús.
Flux principal	Entrada a l'aplicació► Seleccionar► Realitzar activitat
Subflux	No n'hi ha.
Excepcions	No n'hi ha.

2.4. Disseny de pantalles

El primer que l'usuari trobarà de l'aplicació serà el logo de l'aplicació : Una xinxeta.



El logo consisteix en una xinxeta precisament perquè quan es treballa en circuits elèctrics el que s'ha de fer és unir els diferents elements per mitjà de cables o dels components propis; per aquest motiu vaig escollir aquest logo.

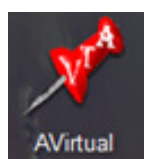




Fig. 11: Xinxeta representativa de l'aplicació.

Fig. 10: Logo dins del context del simulador.

a) Primera pantalla :

presentació de l'aplicació, és a dir, el logo permet identificar, seleccionar i diferenciar l'aplicació. A més a més, el seu cos presenta les lletres que donen nom a l'aplicació .

Per entrar a l'aplicació cal fer un clic al logo, que obre un menú per poder seleccionar el que es vol fer, en aquest cas té un caràcter de tipus educatiu.

b) Segona pantalla. Menú de l'aplicació



En aquesta pantalla el que es vol és que l'usuari esculli què vol repassar, es correspon al primer cas d'ús. Si el que escull és l'electrònica (segon cas d'ús), quan faci clic al botó se li obrirà un menú en forma d'acordió, és a dir, li apareixeran dues opcions que li permetran dibuixar o simular diferents circuits amb diferents elements o bé anar diferents pàgines web corresponents a simuladors que existeixen online de caràcter gratuït.

Fig. 12: Opció escollida: electrònica



Fig. 13: Dues opcions dins del menú principal

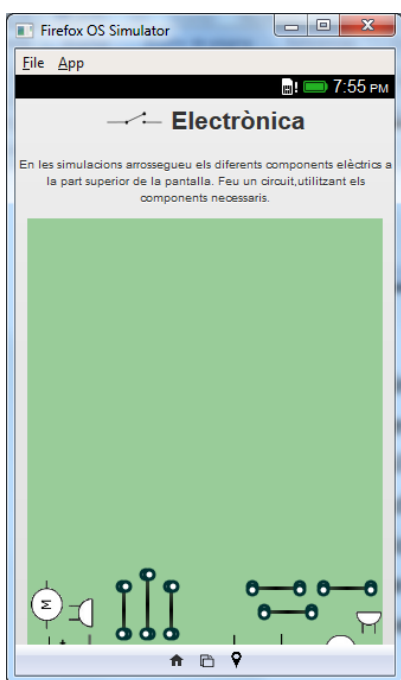
Si escull **robòtica** (tercer cas d'ús), el resultat serà completament diferent. És a dir, en un inici aquesta opció està pensada per poder veure maquetes de diferents tipus de robòtica bàsica preparades per els mateixos alumnes.



Fig. 14: Botó corresponent a la segona opció principal.

- Secció b.1:

Suposem que l'usuari escull la pantalla **dibuixar circuits**. Aquesta pantalla té el propòsit de poder realitzar esquemes de circuits elèctrics. Per poder-ho fer, cal conèixer la simbologia que presenten els diferents components. Aleshores un treball previ consisteix en treballar amb els alumnes per tal que coneguin no només el nom sinó també la simbologia dels components més elementals dels circuits com són els **interruptors**, **pulsadors**, com elements de comandament; **bombeta**, **motor** i **zumbadors** com a receptors o actuadors i una **bateria** com a generador de corrent.



Suposadament un cop coneixen la simbologia han de conèixer també que poden unir aquest components per a dibuixar un circuit funcional. El següent pas consisteix en la descripció de les instruccions, és a dir, que hi ha una mini explicació del què cal fer, tipus: *"Aquests components s'ha d'arrossegar a la part superior de la pantalla, etc."*, i és feina de l'usuari aleshores, pensar en quin tipus de circuits vol fer. És a dir, en sèrie, paral·lel o mixta.

A més hi ha els **cables**, que es poden escalar, o sigui estirar per les puntes, però cal tenir en compte que aquests elements presenten unes dimensions determinades, i quan s'escalen més del que permeten els elements desapareixen de l'escenari o pantalla.

Resultat de possibles circuits:

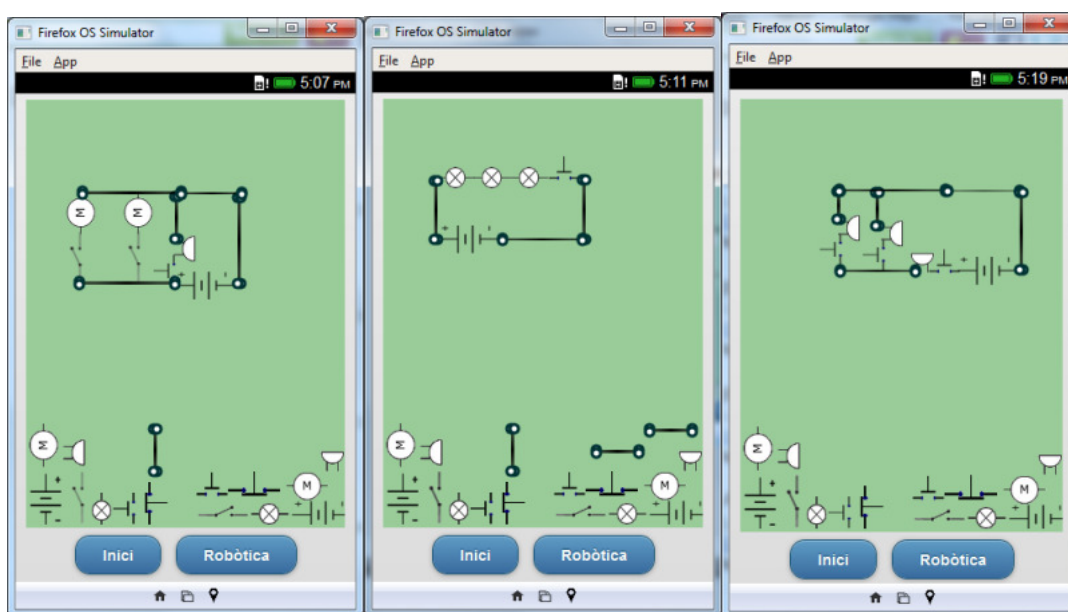


Fig. 15: Exemple de la pantalla electrònica i resultats obtinguts en la construcció de diferents circuits.

-Secció b.2:

Pantalla **Pàgines de simulacions**: En aquesta pantalla es presenten simuladors online, no s'han de instal·lar i faciliten la tasca d'aprenentatge dels alumnes. En la meua recerca he trobat pocs programes gratuïts que permetin accés a fer això. Per una banda demanen donar-se d'alta, o en altres casos estan pensat per alumnes de batxillerat.



El resultat de la pantalla és el següent:

Pàgines seleccionades: Ohm zone i Circuit LAB. Totes dues són simuladors que permeten representar des de circuits simples fins a altres de més complexes.

Per exemple el cas de Circuit LAB el resultat és:

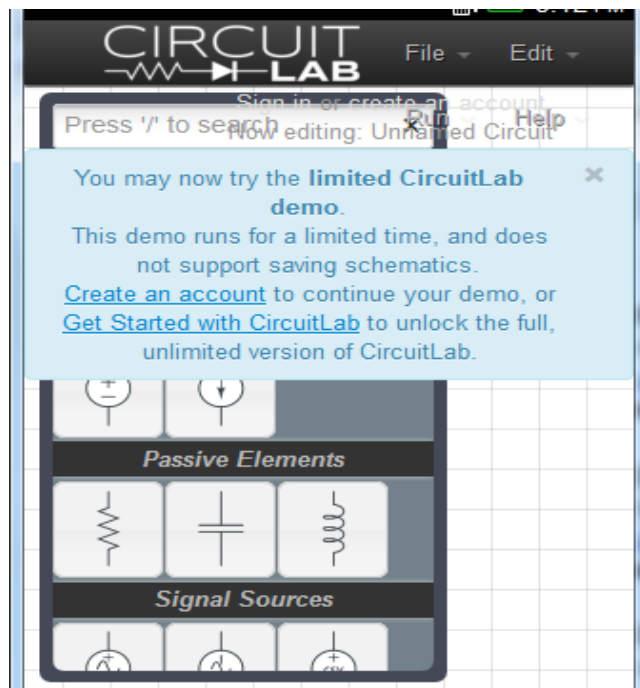


Fig. 16: Mostra de material online

I en el **Ohm zone** el simulador de **Firefox OS** mostra que manquen **plugins**. En altres paraules, ofereixo aquesta possibilitat als usuaris amb el ben entès que aquestes pàgines possiblement completaran i milloraran les seves prestacions en un futur pròxim, però evidentment aquí només es pot fer el que les pàgines en qüestió permetin.

3) Opció escollida: Robòtica



c) Pantalla **Robòtica**: aquesta pantalla està destinada a poder pujar diferents fitxers el



quals mostrin el resultat de la part teòrica realitzada a les aules, portada a la part pràctica. És a dir, els alumnes hauran d'haver construït un cert nombre de robots amb diferents aplicacions, models i funcionalitats a partir de materials variats, gravar el resultat i penjar-ho al sistema .

Aquesta funcionalitat presenta una part introductòria del per què serveix aquesta pantalla. El que es tracta és d'arrossegar el fitxer i posteriorment poder-lo visualitzar directament a la xarxa.

Fig. 17: Opció de robòtica (en fase d'elaboració).

Els alumnes hi han treballant i han realitzant vídeos breus i captat imatges fixes de la feina que hem fet durant aquest trimestre amb els robots.

Els models de robot són senzills, és a dir, consisteixen en un sensor, i alguns altres també són més complicats, programables, que poden realitzar dues o tres ordres. Al mateix temps vaig utilitzar el **JQuery Update** i dedicar una quantitat d'hores superior a les esperades. Ha resultat ser completament satisfactori i molt enriquidor per la quantitat d'expectatives que m'ha creat i el camp que he descobert que està a les beceroles i m'adono que quan més llegeixo, més rectificacions cal fer i més coses es poden fer per a millorar el que ja està fet. No obstant això el resultat és prou bo i he complert, al meu entendre amb escriure, els objectius formulats al projecte inicial.

2.5. Anàlisi tècnic

Relatiu a l'estructura, les vistes es realitzen utilitzant bàsicament **HTML5 i Javascript** ; i per poder realitzar el moviment dels elements que té l'aplicació s'utilitza la llibreria **KineticJS** que ens permet crear l'escenari i poder-hi incorporar el què necessitem. De totes maneres, la sensació que dóna és quan es treballa amb aquesta llibreria o amb el **Canvas**és com una mena d'extensió de **Javascript i HTML5**. Per facilitar el seu manteniment o modificar l'aplicació s'ha separat el codi com si fossin capes, és a dir, per una banda tenim el **codi HTML5** que representa la capa de presentació, però que no és funcional per sí sol ja que cal cridar la part d'estil i **javascript** que representaria la capa de negoci. Aquest sistema presenta una gran avantatge, que és que si es donés



el cas que s'haguessin de fer modificacions per poder-la adaptar a altres dispositius que no fossin el **firefox OS** seria fàcil de fer-ho modificant només la part pertinent.

2.5.1. Implementació

L'aplicació presentada, tal i com està definida en l'apartat anterior, treballa amb objectes que estan vinculats però alhora separats, és a dir, tenen un funcionament propi.

El que he hagut de crear són les funcions per tal de generar els diferents objectes que apareixeran en l'aplicació. Al mateix temps també és necessari que estiguin limitats en un espai. Per poder aconseguir-ho cal crear un mètode o funció que serà cridada en un moment determinat en la presentació de l'aplicació.

A tall d'exemple en la pantalla de disseny de circuits hi intervenen varis objectes que estan limitats per les dimensions que s'ha donat al llençol (Canvas). Per crear l'espai on interactuaran els objectes ho faig pel codi següent :

```
function initStage(images) {  
  
    var stage = newKinetic.Stage({  
  
        container: 'container',  
  
        width: 300,  
  
        height: 400  
  
    });
```

Codi que serveix per a crear un document de javascript en el qual creem la funció **initStage** que serà on apareixeran els objectes.

A continuació diem on han d'aparèixer .En aquest cas és en l'etiqueta `div` que presenta una id amb el nom de `container`. Ara només ens queda donar unes dimensions a l'espai `container` .

El document HTML5 és el que mostra aquest espai per mitjà de la línia:

```
<div id= "container"></div>
```

En el mateix document javascript generem els diferents elements que actuaran en l'espai creat. Cal per tant, anar inserint el següent codi per cada una de les imatges que apareixeran :



```
// bateria vertical
var baterialmg =
newKinetic.Image({
image: images.bateria,
x: 3,
y: 350,
width: 30,
height: 58,
draggable: true,
stroke: 'red',
strokeWidth: 10,
```

Amb això el que s'aconsegueix és incorporar totes les imatges que necessitem per la nostre aplicació.

A continuació s'ha de crear un mètode perquè les carregui a l'espai.

Per fer això es fa per mitjà del codi:

```
function loadImages(sources, callback) { // funció que cridarà les imatges
var images = {}; // definició de la variable images
    varloadedImages = 0; // inciem el contador de les imatges guardades a 0
    var numImages = 0; // el nombre d'imatges inicials també és 0
    // agafa les diferents fonts per mitjà del bucle for
for(var src in sources) {
numImages++;
    }
for(var src in sources) {
images[src] = newImage();
images[src].onload = function() {
if(++loadedImages>= numImages) {
callback(images); // mètode per cridar les imatges
    }
};
    images[src].src = sources[src]; // motrarà les imatges que ha
trobat
    }
}
```

La qual tancarem al final de totes les imatges que volem incorporar en l'aplicació.



La dificultat principal que m'he trobat ha estat a l'hora de poder fer el tancament del circuit, ja que per definició, un circuit elèctric és un camí tancat, per tant necessitava poder dibuixar línies o un altre medi per unir els diferents components que en formen part. Després de varis dies de provar de dibuixar una línia per mitjà del ratolí utilitzant la llibreria **KineticJS**, seguint els exemples que trobava per Internet i veure que a mi no se'm generava la línia he optat per fer un procediment similar al de les imatges però donant una altra propietat que és poder escalar la línia; amb això ja puc tancar el circuit. Per poder fer això, he fet grups d'imatges i definit tots els seus extrems per donar la possibilitat de la seva variació. El que es tracta és de completar una mica més el codi de javascript amb el següent:

```
//creacio del grups que serviran per unir els components
var unioGroup = newKinetic.Group({
    x:80,
    y:310,
    draggable:true
});
//afegir els grups en les capes.
Var layer = newKinetic.Layer();
// variables de grup en una capa
layer.add(unioGroup);
layer.add(unionsGroup);
layer.add(unGroup);
layer.add(dobleGroup);
layer.add(horitzontalGroup);
layer.add(horitzoGroup);
//afegir la capa al llençol o espai
stage.add(layer);
```

Ara només faltava definir els extrems de les línies per poder-les escalar. Per fer això calia incorporar el codi següent:

```
Function update(activeAnchor){
    var group = activeAnchor.getParent(); // crear la variable group i
    crida a la funcioactiveAnchor get
    // part superior de les figures grup
    vartopLeft = group.find('.topLeft')[0];
    vartopRight = group.find('.topRight')[0];
    // part inferior de les figures grup
    varbottomRight = group.find('.bottomRight')[0];
    varbottomLeft= group.find('.bottomLeft')[0];
    var image = group.find('.image')[0];
    varanchorX = activeAnchor.x();
    varanchorY = activeAnchor.y();

    //casos de les diferents possibilitats de la variable activeAnchor
    // utilitzacio del condicional switch
    switch (activeAnchor.name()){
        case'topLeft':
            topRight.y(anchorY);
            bottomLeft.x(anchorX);
            break;
        case'topRight':
            topLeft.y(anchorY);
            bottomRight.x(anchorX);
            break;
    }
```




```
        case 'bottomRight':
            bottomLeft.y(anchorY);
            topRight.x(anchorX);
            break;
        case 'bottomLeft':
            bottomRight.y(anchorY);
            topLeft.x(anchorX);
            break;
    }
    image.setPosition(topLeft.getPosition()); // posició de les dimensions
de la imatge
    var width = topRight.x() - topLeft.x();
    var height = bottomLeft.y() - topLeft.y();
    if (width&&height){
        image.setSize({width:width, height:height});
    }
}
Function addAnchor(group, x, y, name){ // funció d'afegir els grups en la
capa com a l'espai
    var stage = group.getStage();
    var layer = group.getLayer();
var anchor = new Kinetic.Circle({ // definició de com es l'ancle dels extrems
de la imatge
    x:x,
    y:y,
    stroke: '#003333',
    fill: '#fff',
    strokeWidth:3,
    radius:4,
    name: name,
    draggable:true,
    dragOnTop:false

});
// diferents casos que és pot fer amb la imatge
anchor.on('dragmove', function(){ // mou
    update(this);
    layer.draw();
});
anchor.on('mousedowntouchstart', function(){ // el ratolí a sobre
    group.setDraggable(false);
    this.moveToTop();
});
anchor.on('dragend', function(){ // final de l'arrossegament
    group.setDraggable(true);
    layer.draw();
});
//canviar la forma del punter del ratolí
anchor.on('mouseover', function(){
    var layer = this.getLayer();
    document.body.style.cursor = 'pointer';
    this.setStrokeWidth(5);
    layer.draw();
});
anchor.on('mouseout', function(){
    var layer = this.getLayer();
    document.body.style.cursor = 'default';
    this.setStrokeWidth(3);
    layer.draw();
});
group.add(anchor); // afegir les ancle al group
}
```



Aquest document js serà mostrat en el document HTML5 que s'escriu el codi:

```
<script src="nom.js"></script>
```

L'estructura del document HTML5 és un únic document que presenta diferents enllaços interns i cada pantalla està definida pel codi:

```
<div data-role="page" id="electronica">
```

```
    <h2>Electrònica</h2>
```

```
<div><p id="text"> En les simulacions arrossegeu els diferents components elèctrics a la part superior de la pantalla. Feu un circuit, utilitzant els components necessaris.</p>
```

```
</div>
```

```
<div id="container">
```

```
</div>
```

```
    <script src="js/kipropi.js"></script>
```

```
<div id="peu">
```

```
<a href="#inici" data-role="button" data-theme="b" data-inline="true" > Inici</a>
```

```
<a href="#robotica" data-role="button" data-theme="b" data-inline="true"> Robòtica</a>
```

```
</div>
```

```
</div>
```

```
<div data-role="footer" id="footer"><small> Copyright 2014 Virtual | M.MontserratAreñas Vila | UOC - PFC</small>
```

```
</div>
```

```
</div>
```

2.5.2. Funcionalitat de l'aplicació

Quan l'usuari entri a l'aplicació i seleccioni una de les possibles activitats a realitzar es crida la funció que té associada en el document js i es carregen tots els mètodes corresponents.

La part que pràcticament està completada és la pantalla de dibuixar circuits i és la que en aquesta entrega s'ha explicat en més detall.

2.6. Creació de l'aplicació(II)

2.6.1. Implementació de la pantalla robòtica .

La pantalla robòtica serveix entre altres coses per visualitzar els vídeos realitzats pels alumnes . Els alumnes d'ESO, agrupats per interessos han elaborat i dut a terme aquest trimestre, sota la meua monitorització i orientació, els seus projectes relacionats amb la robòtica. Passem directe als resultats, és a dir, a la part que no hi ha intervingut els alumnes. El primer que cal fer són les gravacions de diferents parts del procés de construcció de la maqueta, a continuació realitzar la seva edició i finalment guardar-la en el format adequat segons el sistema que s'utilitza. En aquest cas, ha de ser .webm i .mp4

2.6.1.1. Resultats inicials: n Pantalles en blanc

Un cop fet aquest procés, cal que l'aplicació ens mostri els vídeos. Per poder fer aquesta tasca, el que vaig fer en principi és donar-me d'alta en el **blogTumblr**, el qual presenta una apartat per a desenvolupadors d'**Apps**. Hi vaig pujar els vídeos que tenia en format .mp4

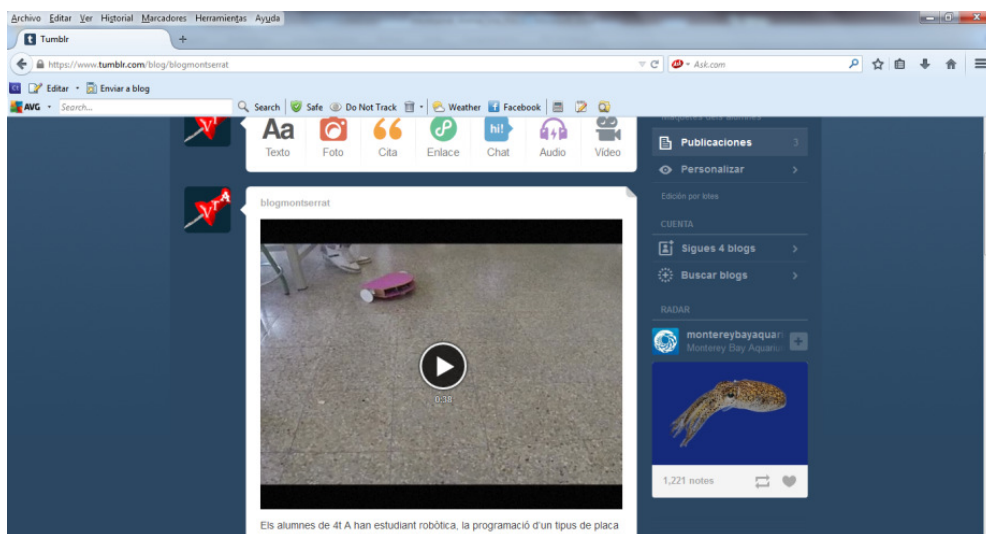


Fig. 10: Mostra els videos penjats al Tumblr

A continuació calia registrar l'aplicació. Quan es realitza aquest pas s'obtenen dues claus per poder tenir accés als diferents posts que s'hagin creat en el blog i procedir seguint les indicacions que tenen establertes.

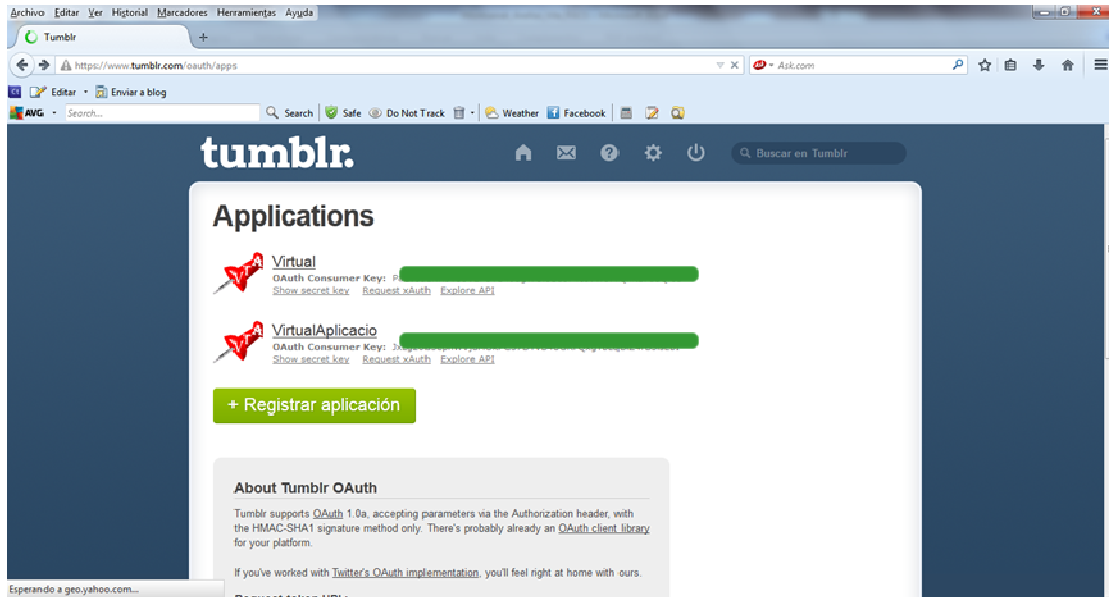


Fig. 19: Resultat de registrar l'aplicació

Cal omplir un formulari i després s'obtenen les dues claus, una correspon a **OAuth Consumer Key** i l'altre és la clau secreta.

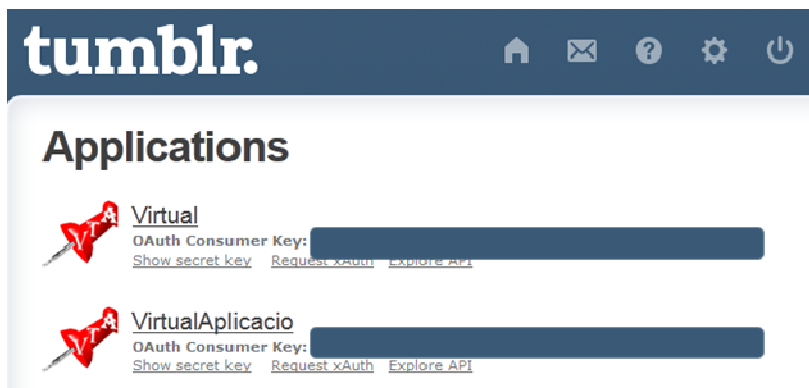


Fig. 20: Detall de la clau OAuthconsumerkey

I ja per últim només cal cridar els posts en l'aplicació. Per això vaig analitzar el PFC d'en xfernandezs . Vaig fer la crida pertinent, però el resultat va ser **una pantalla en blanc** en el simulador de **Firefox OS**. El codi que vaig fer servir és:

```
//funció que farà una crida als vídeos per mitjà del get
function getVideos(json){
    var videos = []; // variable que està formada
per un array [ ]
    var f_url; // variable que es direcciona a
una adreça url, on hi ha els vídeos
    var f_caption;
```



```
        var post = json['response']['posts']; //
variable que retorna els post trobats en el blog
tumblr

        var tag =
window.localStorage.getItem("HashTagDetall");

        for(i = 0; i <post.length; i++){ //
bucle per mostrar tots els post trobats

                if(post[i]['tags']==tag){

                        f_caption = post[i]['caption'];

f_caption=f_caption.replace("<p>", "");

f_caption=f_caption.replace("</p>", "");

                        f_url = post[i]['video_url'];

                        videos.push({caption: f_caption,
url: f_url});

                }

        }

        var llistavideos = []; // variable que serveix
per mostrar els videos sempre i quan es doni el
bucle for

                for(i=0; i<videos.length; i++){

                        var tall = videos[i].caption;

// condicions o característiques que presenta el
vídeo que es mostra

                        var img = "<videoid=video"+i+"
width='200' height='200'
controls><sourcesrc='"+videos[i].url+"
type='video/mp4'></video>"; llistavideos.push(img);
} $('#videos_segment').html(llistavideos.join(''));
}

function retornaTumblrVideos(){ // funció que
permetrà veure els vídeos

        $.ajax({ // indica l'adreça que els ha de
buscar i la clau privada proporcionada
                url:"http://api.tumblr.com/v2/blog/blogmontse
rrat.tumblr.com/posts", method: 'get',
                data: ({

                        api_key:
'PaEv9I1NHxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx', type:
'video' })),

                dataType:"jsonp",

                jsonpCallback: "getVideos", // els sistema
que es fa servir per obtenir els vídeos es per mitjà
del mètode callback

                complete: function() { },
```

```
// en cas de no trobar-ne mostrarà un missatge d'error  
  
error: function() { alert("error");}
```

A l'observar que no funcionava, vaig començar a buscar alternatives, una d'elles: donar-me d'alta en el **Parse**:

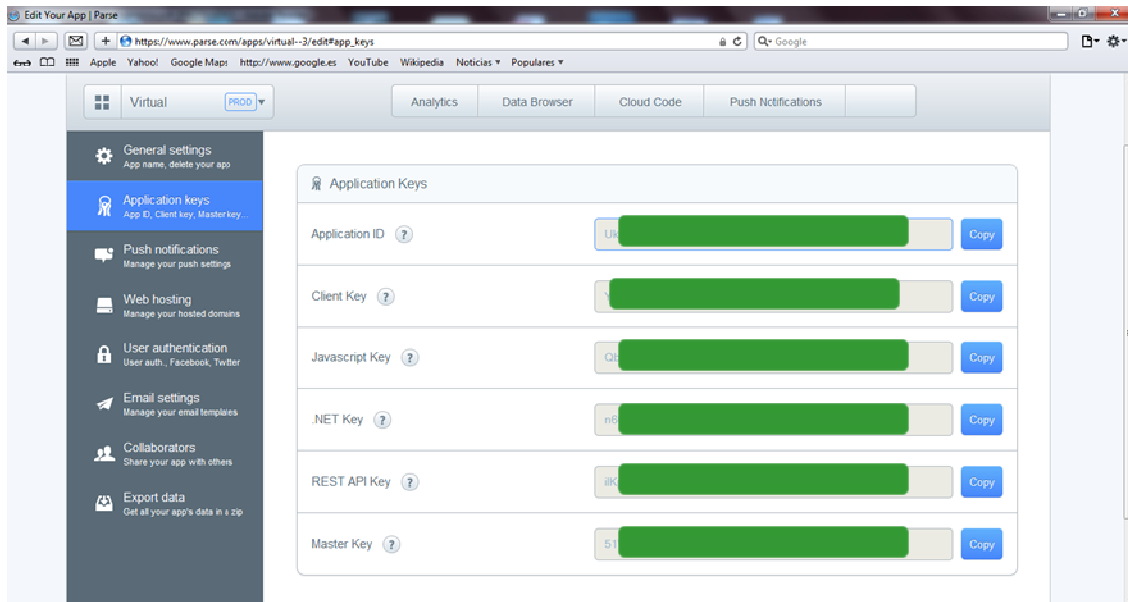


Fig. 21: Claus del Parse

Un cop registrada l'aplicació intento pujar els arxius. El primer en què em trobo és que no reconeix l'arxiu **manifest.webapp**, primera dificultat. Aleshores estic en un punt mort i cal solucionar el tema.

Buscant informació de com es pot publicar una aplicació pel sistema **Firefox OS** trobo que es pot fer per mitjà del **Marketplace** o bé en dominis propis. En aquest darrer cas cal fer alteracions ja que els servidors reconeixen arxius **.json**, de manera que l'extensió **.webapp** no la reconeixen. Conclusió: hauria de crear un fitxer amb l'extensió **.htaccess**. per tal de que el servidor reconegués a l'arxiu **.webapp** com a **json**.

El codi del fitxer seria:

```
#afegir aquesta en htaccess( en servidor Apache)  
AddType application/x-web-app-manifest+json.webapp  
  
#afegir en mime.types aquesta línia( en servidor NGiNX)  
  
types{ application/x-web-app-manifest+jsonwebapp;}
```

Davant aquest nou entrebanc busco informació per poder accedir a les dades de **Tumblr**. i en la recerca trobo una pàgina que mostra el resultat i el codi emprat per incorporar els diferents tipus de post que presenta. Incorporo el codi en l'aplicació; en principi el resultat va ser la **pantalla en blanc**.

En aquest moment es quan em platejo que possiblement l'error no és el codi, sinó que el tipus de fitxer de vídeo. Per solucionar això, el que faig és inserir el codi per vídeo. Prèviament he creat una carpeta amb els vídeos.

Per exemple:

```
<video id=2 controls loop width="250" height="150" >// etiqueta que permet  
incorporar un vídeo amb codi HTML5  
  
<source src="videos_exemple/grup4tB.mp4" type="video/mp4">// aquesta etiqueta  
fa la funció de buscar el vídeo especificat  
  
</video>
```

Vaig al simulador de **Firefox OS**, el resultat torna ser la **pantalla en blanc**. Cal seguir treballant i efectuar canvis.

2.6.2.2. Resultats finals: Possibilitats de publicar els vídeos.

Incorpo la possibilitat de que el vídeo sigui webm:

```
<video id=2 controls loop width="250" height="150" >  
<source src="videos_exemple/grup4tB.mp4" type="video/mp4">  
<source src="videos_exemple/grup4tB.webm" type= video/webm>  
</video>
```

Torno a anar al simulador per comprovar el resultat.



El vídeo es mostra. Ara cal provar si el **Tumblr** accepta aquest tipus de fitxer de vídeo. Accedeixo al **Tumblr** i carrego el fitxer amb extensió .webm, s'ha carregat i espero s'hi es pugui visualitzar, el resultat és:

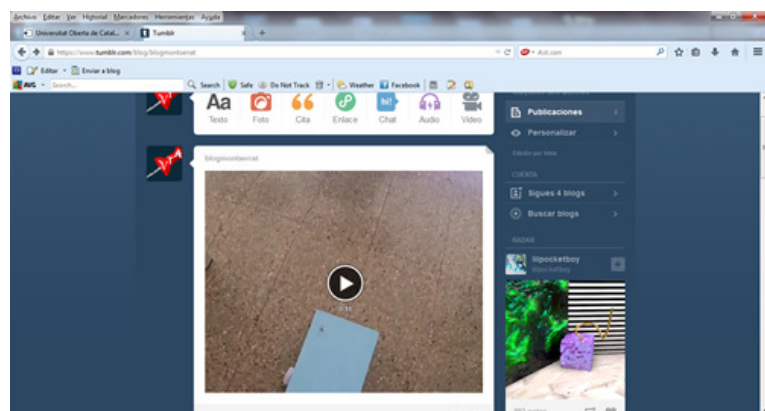


Fig. 11: Resultat del simulador



Repeteix el procés per les diferents gravacions realitzades, i comprovo que totes són visibles en el simulador.

A continuació cal comprovar el codi que permet interactuar amb el **blogTumblr** perquè mostri el post que conté. Part del codi és el següent:

```
<script type='text/javascript'>

    //

    var listaTU=newArray();

    var maxsTU=0;

    var actualTU=0;

    functiontumblrCallback(obj) {

    // ----- dades usuari -----

    var usuario = "http://blogmontserrat.tumblr.com/"; // la URL de Tumblr

    var cuantosPosts = 10; // quantitat de post que es llegirant

    // ----- final dades -----

    var contarAux=0;

    var tumbhrs = obj;

    for (var i=0; i&lt;cuantosPosts; i++){

    elPost= tumbhrs.posts[i]; // llegir cada un dels posts

    // algunes dades, els llegirem com a text pla eliminant els formats

    textoPlano = elPost["regular-body"] || elPost["photo-caption"] ||
    elPost["quote-source"] || elPost["link-text"] || elPost["link-url"] ||
    elPost["video-caption"] || elPost["audio-caption"] || "";

    // filtrem els tipus de posts a mostrar

    if (/regular|photo|quote|link|video|audio/.test(elPost.type)) {

    //i en funció del tipus de post, llegim el contingut

    if (elPost.type === "photo") { // les fotos

    photoURL = elPost["photo-url-250"];

    contenido = '&lt;div class="TumblrPhoto" &gt;&lt;imgsrc="' + photoURL + '"
    /&gt;&lt;/div&gt;'

    } elseif (elPost.type === "quote") { // les citas

    contenido = '&lt;div class="TumblrCita"&gt;&lt;span&gt;' + elPost["quote-text"] +
    '&lt;/span&gt;' + textoPlano + '&lt;/div&gt;';

    } elseif (elPost.type === "link") { // els enllaços

    contenido = '&lt;div class="TumblrLink"&gt;&lt;a href="' + elPost["link-url"] +
    '"&gt;' + textoPlano + '&lt;/a&gt;&lt;/div&gt;';

    } elseif (elPost.type === "video") { // els vídeo</pre></div><div data-bbox="138 936 603 957" data-label="Page-Footer"><p>Maria Montserrat Areñas Vila <img alt="32 in a box" data-bbox="325 936 362 957"/> PFC: VIRTUAL, una aplicació educativa</p></div>
```


El codi de la sortida és:

```
if (contenido) {  
    // contingut que es mostra  
    salida = contenido;  
    // enllaç directe al post  
    salida += '<div id=\"verTumblr\"><a target=\"_blank\" href=\"' +  
    elPost.url + '\">ver post</a></div>';  
    // i es guarda en llista  
    listaTU[contarAux] = salida;  
    contenido = \"\";  
    salida = \"\";  
    contarAux =contarAux +1;  
}  
}  
}  
  
//quan acabem de llegir, mostrem l'últim post publicat  
maxSTU=contarAux;  
document.getElementById('contenedorTumblr').innerHTML = listaTU[0];  
}
```

Un cop tinc tot el codi fet només cal comprovar si funciona correctament, és a dir, si apareixen els vídeos que s'han penjat en el **Tumblr**. El resultat que obtinc és un requadre:

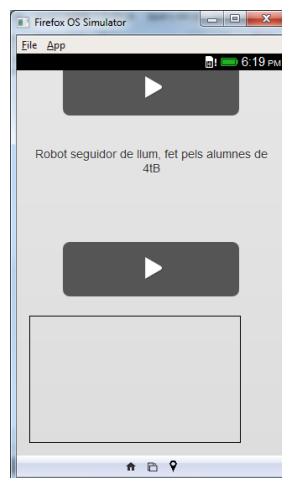


Fig. 23: Resultat obtingut després d'aplicar el codi font. Mostra només el requadre on haurien de sortir els posts de Tumblr

2.7. Publicació de l'aplicació



Per poder veure els resultats de manera efectiva cal publicar l'aplicació. Segons les lectures realitzades, per a la publicació d'una aplicació feta amb **Firefox OS** hi ha dues possibilitats: Una per mitjà del **marketplace** o l'altra, tenir un domini propi; a més a més, segons el nivell d'accés intern que requereix les aplicacions poden ser:

Fig. 24: Detall del visionat de dos vídeos

- Aplicacions certificades: Reben aquest nom aquelles aplicacions que tenen accés a funcions crítiques del telèfon mòbil. En aquestes el permís a API's crítiques està acceptat implícitament.
- Aplicacions privilegiades: són aquelles aplicacions que tenen un accés a API's internes del SO de manera explícita.
- Aplicacions regulars: s'anomenen aquelles aplicacions que no necessiten accedir a cap API interna, per tant no es demanen permisos i no hi tenen dret.

En el meu cas l'aplicació és regular. Ara manca tenir accés a un servidor. En el cas del **Marketplace** segons diu la literatura només cal fer un zip de l'aplicació, fa una revisió i a continuació és produïx la publicació.

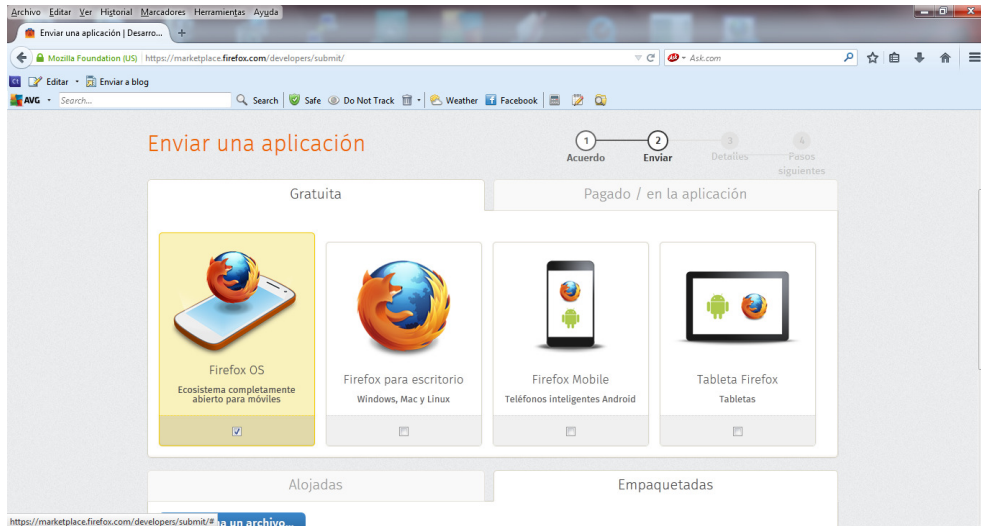



Fig. 25: Detall de la pantalla de publicaci3n del marketplace

Seguidament he omplert un formulari. Finalment un cop acabat el formulari i enviat el fitxer .zip.

Per email rebo la certificaci3n de l'aplicaci3n.

GLOBAL RATING TOOL



Rating Certificate

Application Name:	AVirtual
Storefront:	(Mozilla) Firefox Marketplace
Certificate issued to:	marenas6@xtec.cat
Date issued:	Jun 4 2014 2:44PM
Submission ID:	6524
Security Code:	I0K09TQ






Rating System	Region	Rating Category	Descriptors
PEGI	Europe		
USK	Germany		
ESRB	The Americas		
CLASSIND	Brazil		

Fig. 26: Detall de la certificaci3n

I finalment també m'envien un email on consta l'adreça que farà efectiva la descarrega de l'aplicaci3n i demanant que modifiqui l'icona, la qual ha de presentar un marc i unes dimensions determinades.

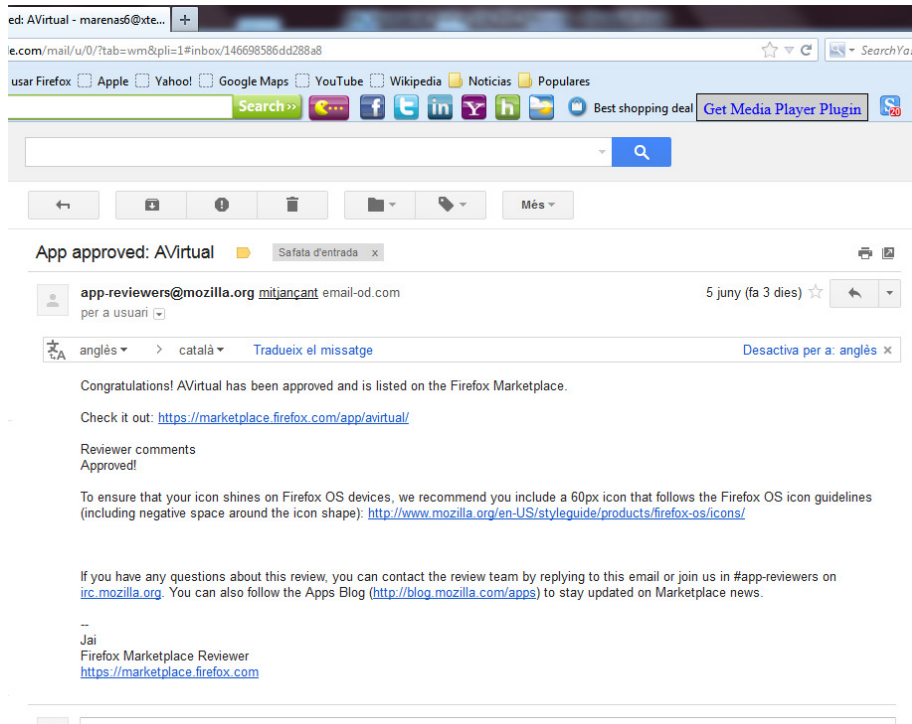


Fig. 27: Correu on indiquen l'adreça i modificació de la icona

Adreça de l'aplicació: <https://marketplace.firefox.com/app/avirtual/>

3. Vídeo presentació de la feina feta.

4. Cloenda

A partir del moment en que va ser possible efectuar els dibuixos i progressivament després van poder fer el visionat de la part final simulador els alumnes s'han entretingut tant com han volgut per a dibuixar esquemes de circuits, com per localitzar, veure i reconèixer els robots que ells mateixos havien fet. La satisfacció de la feina feta, tot i deixar molts interrogants per la quantitat d'hores que hi he invertit i per els inconvenients que mica en mica he hagut de superar és suficient com per deixar una porta oberta a continuar investigant per fer noves aplicacions. A partir d'aquest moment doncs, ja es pot donar per tancada la tasca inicial prevista per el treball de final de carrera i esperar que efectivament sigui també satisfactòria per el consultor qualificador.

Les millores en un futur poden ser canviar el elements que serveixen per connectar els diferents components d'un circuit, és a dir incorporar la llibreria fabric per poder dibuixar les connexions per mitjà de línies que unrien els components que formen el circuit.



Incorporar la possibilitat de pujar els vídeos directament a l'espai de l'aplicació per tal de poder-los compartir.



Annex: Referències Bibliogràfiques

- Lectura recomanada per prendre el primer contacte amb les app FIREFOX OS (https://developer.mozilla.org/en-US/Apps/Quickstart/Build/Your_first_app)
- Roset, Roman. Els enfoc per escollir un framework([http://biblioteca.uoc.edu/prestatgeries/11116/Escollir_el_framework_JS_\(i_l_estil\)_de_treball.pdf](http://biblioteca.uoc.edu/prestatgeries/11116/Escollir_el_framework_JS_(i_l_estil)_de_treball.pdf))
- Roset, Roman. Introducció a les aplicacions per a telefonia mòbil (http://biblioteca.uoc.edu/prestatgeries/11116/Introduccio_a_les_aplicacions_per_a_telefonia_mobil.pdf)
- Guia a la galàxia d'aplicacions mòbils(http://www.enough.de/fileadmin/uploads/dev_guide_pdfs/Guide_13thEdition_Catalan_WEB.pdf)
- El gran libro de HTML5, CSS3 y Javascript(<https://sistemasquimica.wikispaces.com/file/view/El+gran+libro+de+HTML5,+CSS3+y+Javascript+-+Juan+Diego+Gauchat.pdf>)
- Vídeos:
 - Développer installer une application Firefox OS (https://www.youtube.com/watch?v=E_-tmJ1MiQ)
 - Firefox OS installation(<https://www.youtube.com/watch?v=g3laHNypTul>)
 - Firefox OS Simulator 3.0 (<https://www.youtube.com/watch?v=a8EiP1p2kGc>)
 - Firefox OS Simulator (<https://www.youtube.com/watch?v=cqVXpg6n6Mg>)
- icona: (<http://www.pekegifs.com/adivinaquees/chincheta.gif>)
- Manual de jQuery Mobile (<http://www.desarrolloweb.com/manuales/manual-jquery-mobile.html>)
- Article d'exemple de creació d'una aplicació(<http://www.noupe.com/tutorial/jquery-mobile-tutorial-creating-a-restaurant-picker-web-app.html>)
- Kineticjs tutorial (<http://www.html5canvastutorials.com/kineticjs/html5-canvas-events-tutorials-introduction-with-kineticjs/>)
- Memòria jmontoyaoTFC0113(<http://openaccess.uoc.edu/webapps/o2/handle/10609/18829>)
- Tutorial per crear una pàgina web per dibuixar una espècie de painttoline utilitzant jQuery i l'element canvas de HTML5 (<http://html5tutoriales.com/tutoriales/jquery-canvas-dibujar-con-jquery-canvas.html>)
- Manera d'incorporar el blogtumblr a altres pàgines web.
<http://vagabundia.blogspot.com/2009/04/tumblr-y-blogger-parte-1.html>
<http://vagabundia.blogspot.com/2009/04/tumblr-y-blogger-parte-2.html>
- Api de Tumblr
- Memòria de Xavier Fernández
- Publicació d'una aplicació Firefox OS
<http://www.pabloglesias.com/desarrollando-para-firefox-os-iii-archivos-de-una-aplicacion/>
- MDN
https://developer.mozilla.org/es/Marketplace/Submission/Submitting_an_app

