

# *dEIC Kiosk*. Desenvolupament d'un punt d'informació tàctil basat en web

David González Tejerina

**Resum**– En aquest treball es presenta el desenvolupament de *dEIC Kiosk*, un quiosc digital tàctil que ofereix informació sobre el Departament d'Enginyeria de la Informació i de les Comunicacions. El quiosc extreu informació de la base de dades del departament i de webs externes i la mostra en una pantalla tàctil situada en el propi departament o en un dispositiu remot, convenientment adaptada a la seva resolució. El quiosc respon a la necessitat de donar informació com dades de professors, horaris de laboratoris... a les persones que estan físicament al departament. En aquest treball es mostra com s'ha desenvolupat el quiosc i s'analitzen les diferents metodologies de desenvolupament, conceptes de disseny i estructura de la web, tecnologies i llenguatges (PHP, Bootstrap, jQuery...), i diferents aspectes de configuració de la pantalla. El resultat es troba accessible al passadís del departament i a la web del dEIC.

**Paraules clau**– quiosc, tàctil, disseny web, aplicació web, Bootstrap, responsive, PHP, jQuery, Google Maps API, web scraping

**Abstract**– This work presents the development of *dEIC Kiosk*, a digital touch kiosk that offers information about the Information and Communications Engineering Department. The kiosk extracts information from the department database and external websites and shows it in a touch screen which can be found in the department or by a remote device, suitably adapted to its resolution. This kiosk meets the need to give information about professors, laboratory schedules, etc., to people who are in person in the department. This work shows the development of the kiosk and analyses different development methodologies, design concepts and web structure, technologies and languages (PHP, Bootstrap, jQuery...) and different aspects of the screen configuration. The result is available in the hallway of the department and in the dEIC website.

**Keywords**– kiosk, tactile, web design, web application, Bootstrap, responsive, PHP, jQuery, Google Maps API, web scraping



## 1 INTRODUCCIÓ

EXISTEIX multitud d'informació que es pot requerir al passadís dels despatxos d'una Universitat. Conèixer el despatx d'un professor, si els laboratoris estan lliures o ocupats... O, de vegades, necessitats que poc tenen a veure amb la institució en sí; com conèixer els horaris del transport la ruta més ràpida per arribar a un punt, estar al dia de les últimes notícies... Totes aquestes necessitats es poden cobrir amb un quiosc digital, cada cop més freqüents a diferents entitats i institucions. En aquest projecte es desenvoluparà un quiosc digital per al departament d'Enginyeria de la Informació i les Comunicacions de la Universitat Autònoma de Barcelona.

- E-mail de contacte: davidgonzalezte@e-campus.uab.cat
- Menció realitzada: Tecnologies de la Informació
- Treball tutoritzat per: Sergi Robles Martínez (dEIC)
- Curs 2015/16

El dEIC (Departament d'Enginyeria de la Informació i de les Comunicacions) disposa d'una pantalla tàctil situada al passadís dels despatxos. Abans de la consecució del projecte es disposava d'una aplicació web provisional que necessitava ser millorada i dotada de més funcionalitats.

Per aquests despatxos poden circular multitud de persones: estudiants, professors, visitants... Aquest punt d'informació o quiosc podria oferir múltiples serveis als usuaris. Tals com informació dels professors del departament, reserves d'aules, informació del trànsit o sobre el transport públic, entreteniment...

En aquest projecte es tracta d'identificar quines són aquestes necessitats i serveis que es poden oferir i desenvolupar una aplicació web que les satisfaci. El desenvolupament d'aquest projecte ha servit, també, per explorar les diferents alternatives de disseny web i valorar les alternatives de configuració de la infraestructura.

L'objectiu d'aquest projecte és desenvolupar una aplicació web per a un punt d'informació tàctil. El punt d'informació està situat en un departament de la Universitat

Autònoma de Barcelona i ha de cobrir les necessitats informatives d'estudiants, professors i visitants.

A continuació es definiran els subobjectius necessaris per assolir amb èxit l'objectiu del projecte ordenats per prioritats:

1. Analitzar i realitzar el disseny web que s'adapti millor al projecte en qüestió.
2. Analitzar els diferents mètodes i tecnologies per consumir dades d'altres webs o serveis.
3. Estudiar les tecnologies web que són tendència actualment i ens han de permetre la implementació.
4. Estudiar les diferents alternatives per fer una web que s'adapti a les mides de la pantalla.
5. Afrontar el problema de seleccionar una metodologia de treball i desenvolupament.
6. Estudiar els diferents tipus de proves que es poden realitzar en l'àmbit de les aplicacions web.
7. Analitzar quines consideracions i configuracions calen per tal de poder executar una aplicació web en una pantalla tàctil.

En aquesta introducció s'ha contextualitzat el projecte i explicat la motivació del mateix. A continuació s'han exposat els objectius que es pretenen assolir i, per últim, es presenta una guia de l'article.

Seguidament es pot trobar una secció amb l'estat de l'art. En aquest apartat s'analitzen les diverses metodologies que es poden seguir per assolir un projecte d'aquestes característiques i algunes de les tecnologies que cal tenir en compte per assolir el projecte de la manera més efectiva possible.

A continuació es realitza un anàlisi dels requisits del projecte i els recursos disponibles per tal d'oferir una idea clara de l'abast del projecte que es pretén desenvolupar.

Fruit de l'anàlisi dels requisits es defineix una metodologia que s'adeqüi al projecte i permeti assolir-lo amb èxit amb els recursos destinats.

Posteriorment es presenten les principals tecnologies que s'han utilitzat al projecte justificant el seu ús. Un cop presentades les principals tecnologies utilitzades s'aprofundeix en alguns aspectes del desenvolupament del projecte com la configuració de la màquina, el patró de disseny utilitzat i la forma d'extreure les dades.

A continuació es presenten les proves i accions realitzades per assegurar la fiabilitat i qualitat del codi.

Posteriorment es presenten els resultats obtinguts i les línies de millora.

Per últim, s'exposen les conclusions extretes del desenvolupament del projecte.

## 2 ESTAT DE L'ART

En aquesta secció s'analitzen alternatives similars al projecte desenvolupat, per tal de recollir l'experiència d'altres projectes similars i els problemes a resoldre. S'estudien, també, les tecnologies, llenguatges i metodologies disponibles per al disseny d'una aplicació web de les carac-

terístiques descrites. Per últim, s'esmenten algunes consideracions que cal tenir en compte a l'hora de configurar el panell tàctil.

Existeixen multitud de quiosc multimèdia similars al projecte que es pretén desenvolupar. A continuació es llisten alguns exemples de projectes similars:

- Kiosk multimèdia per a gestions acadèmiques [1]. Aquest quiosc, però, no és responsive i està dissenyat per a una finalitat molt específica. Les característiques són molt diferents i els projectes no són assimilables.
- Kiosk multimèdia per a un punt d'informació d'un centre comercial (multitouch) [2]. Aquest projecte analitza amb profunditat la interacció entre els usuaris i el panell multitàctil. No profunditza, però, en el desenvolupament web de l'aplicació. Les característiques d'aquest projecte són més assimilables al nostre projecte que no pas l'anterior referència. Tot i així, tant les fonts d'on extreuen les dades com el concepte en sí s'allunya del nostre projecte.

Aquests sistemes poden servir com a base i referència a l'hora de desenvolupar el projecte però, com s'ha vist, les seves característiques són molt diferents al projecte que es vol realitzar.

Una opció per reduir els costos de desenvolupament del projecte podria ser l'ús d'un gestor de continguts (CMS) [3]. Existeixen diferents CMS com, per exemple, *Joomla* o *Drupal*. Els CMS permeten una alta independència entre els continguts (dades de l'aplicació) i la forma en que aquestes es gestionen i visualitzen (controlador i vista). El seu objectiu, per tant, és dotar de transparència la publicació de continguts web. Al *dEIC Kiosk* però, no cal una gestió periòdica dels continguts donat que aquests, o bé s'extreuen d'una base de dades mantinguda pel departament (dades de professors, horari laboratoris...) o bé s'extreuen de webs de tercers (mapes, horaris transport, notícies...). Per aquest motiu la solució que ofereixen els CMS no està enfocada al problema que es pretén resoldre i no aporta cap millora substancial al projecte.

Descartades les alternatives de quiosc presentades i els gestors de continguts cal analitzar els diferents frameworks i llenguatges que poden ser útils en el desenvolupament d'un projecte de les característiques esmentades.

Un dels aspectes més innovadors en quant a tecnologies web és la necessitat d'adaptar-los a diferents dispositius amb diferents mides de pantalla (dispositius mòbils, pantalles tàctils, pantalles d'ordinador...). Cal diferenciar i estudiar les diferents alternatives [8] [11]. Existeixen nombrosos frameworks i llibreries de classes que faciliten el desenvolupament d'aplicacions web responsive, un dels més utilitzats és *Bootstrap* [6].

En quant als llenguatges de programació de la banda del servidor hi trobem, segons la web *Tiobe* que els més utilitzats són: Java, C, C++, Python, C# i PHP [7]. Respecte als llenguatges de la banda del client destaca clarament Javascript.

Un altre aspecte important del punt d'informació serà la gestió dels mapes i les rutes. En aquest punt es fa indispensable requerir a la tecnologia de Google Maps i a la seva API de Javascript [12].

És important, també, establir una bona metodologia de desenvolupament [4] i uns patrons adequats per a la confecció de l'aplicació i la interfície gràfica [5].

Per últim, cal parar atenció a la forma en què s'executarà el quiosc a la pantalla tàctil. Existeixen diverses formes d'executar un quiosc en una pantalla. En el cas dels quioscs desenvolupats en format web aquests han d'executar-se en un navegador. Existeixen, però, multitud de plugins per als diferents navegadors que faciliten la visualització de webs en quioscos [9] [17] executant la web en pantalla completa i millorant la usabilitat del navegador per aquest tipus d'ús. Existeixen plugins, també, per simular un teclat a la pantalla [10]. Aquesta característica és molt important en quioscos que requereixen que l'usuari introdueixi text i no disposen de teclat físic.

En conclusió, s'han analitzat alternatives similars al projecte desenvolupat però aquests tenen unes característiques que no els fan assimilables. Els gestors de continguts tampoc resolen el problema plantejat donat que estan més pensats per publicar continguts de manera transparent. Caldrà, per tant, desenvolupar una aplicació web pròpia; per fer això cal posar atenció als llenguatges, frameworks i metodologies de desenvolupament més adequades i que estan marcant tendència actualment. Per últim, s'han analitzat els diferents plugins i aspectes a tenir en compte en la configuració i posada del marxa del quiosc a la pantalla tàctil.

### 3 ANÀLISI

En aquesta secció es realitza un anàlisi del projecte. Inicialment apareixen enumerats els requisits funcionals i no funcionals. A continuació es pot trobar un anàlisi dels recursos disponibles per a la consecució del projecte. Per últim es presenta la infraestructura disponible pel projecte.

#### 3.1 Requisits

En aquesta secció s'enumeren els requisits funcionals i no funcionals del projecte. Aquests requisits s'estableixen fruit dels objectius definits anteriorment.

El projecte ha de satisfer els següents requisits funcionals:

- Consultar el directori de la UAB amb les dades dels professors, despatxos...
- Enviar missatges als professors del departament. L'adreça de destinació ha de ser la institucional de la universitat.
- Enllaçar amb els fòrums i continguts de les diferents assignatures del departament disponibles a la pàgina web del departament.
- Consultar els horaris i reserves dels laboratoris així com els horaris i grups de pràctiques.
- Mostrar el llistat d'estudiants top segons l'aplicació que desenvolupa un altre estudiant en paral·lel. Aquesta aplicació elabora un rànquing de mèrits del estudiants que hi participen. Els mèrits s'obtenen realitzant entregues, fent treballs...

- Permetre buscar rutes i plànols de carreteres per arribar a diferents destinacions des del quiosc.
- Mostrar horaris de trens (Renfe i FGC) i línies de bus. Predefinir la UAB com a punt de sortida.
- Carrousel d'imatges on es puguin anar consultant els diferents pòsters publicats.
- Permetre llegir notícies científiques i tecnològiques relacionades amb els continguts que s'imparteixen a l'escola.
- Permetre llegir notícies generals i d'actualitat.

El projecte ha de satisfer els següents requisits no funcionals:

- L'aplicació es desenvoluparà en PHP donat que el servidor web està preparat per suportar-lo.
- Es podran realitzar consultes sobre la base de dades del departament però, perseguretat no es poden fer insercions ni eliminacions. Es podran crear noves instàncies de BD en cas de ser necessari.
- L'aplicació web serà visualment agradable i estarà pensada per utilitzar-se en dispositius tàctils amb facilitat, fluïdesa i de forma dinàmica.
- Es respectaran els colors corporatius del departament (tons taronges, marrons i grocs).
- L'aplicació ha de poder visualitzar-se en dispositius mòbils i ordinadors.

Els requisits funcionals i no funcionals exposats anteriorment defineixen gran part de les funcionalitats que ha d'implementar l'aplicació web així com algunes de les tecnologies que han hagut d'utilitzar-se al projecte.

#### 3.2 Recursos

En aquesta secció es definiran els recursos disponibles per portar a terme el projecte. Inicialment es fa esment als recursos humans per desenvolupar el projecte i, posteriorment, als recursos materials i d'infraestructura.

Aquest projecte té assignades 300 hores (incloent planificació, reunions, documentació, desenvolupament, test...). Aquestes 300 hores estan assignades íntegrament a un únic projectista. Addicionalment es disposa del suport del tutor del treball que el supervisarà i orientarà. Si és necessari, també es pot comptar amb el tècnic del departament per realitzar petites gestions relacionades amb la infraestructura.

Es disposa dels següents recursos materials per implementar el projecte:

- **Pantalla tàctil:** ViewSonic TD2220. És una pantalla tàctil multi-touch de 2 punts amb 22" i una resolució de 1920x1080. Disposa de 2 altaveus integrats i disposa de sistema VESA amb el qual està collat a la paret del passadís.
- **Servidor kiosk:** es tracta d'una màquina situada a la sala Turing. Disposa d'un sistema Ubuntu 16.04 LTS

amb la versió del kernel 4.4 de 64 bits. Es troba al domini `abra.uab.es`. La pantalla està connectada a aquesta màquina, que executa un navegador de manera permanent i que no és possible minimitzar per tal d'assegurar que els usuaris no l'utilitzen fora de l'aplicació web.

- **Servidor deic:** conté la base de dades del departament. L'accés a aquesta màquina és molt limitat per raons de seguretat. Aquesta base de dades està implementada en PostgreSQL.

En aquest apartat hem pogut determinar quins recursos s'utilitzaran per l'execució d'aquest projecte; tant humans com materials.

En conclusió, el *deIC Kiosk* ha de disposar de múltiples funcionalitats i poder visualitzar-se en qualsevol tipus de dispositiu. S'haurà de desenvolupar en PHP i tindrà accés a la base de dades del departament. Per assolir el projecte es disposa de 300 hores d'un únic projectista i del hardware necessari.

## 4 METODOLOGIA

En aquest apartat es definirà la metodologia de desenvolupament del projecte, que serà incremental i iterativa. L'objectiu de la metodologia que es presenta és ajustar-se al màxim a les característiques del projecte analitzades al punt anterior.

Per tal de definir la metodologia de desenvolupament és necessari clarificar quina estructura ha de tenir l'aplicació. Aquesta consta d'un marc amb un menú que es mostra sempre. A l'espai reservat per contingut, que és la major part de la pantalla, inicialment es mostra la pantalla principal. La pantalla principal consta de diversos enllaços a funcionalitats de l'aplicació encabits en aquesta a mode de *widgets*.

Atenent a aquesta estructura és necessari definir inicialment el menú i la pantalla principal, és a dir, l'estructura de l'aplicació web, i tractar les diferents funcionalitats com a mòduls independents que s'hauran d'adaptar a aquest marc.

S'ha utilitzat, per tant, una metodologia de treball incremental i iterativa [4]. És a dir, en termes generals el projecte es desenvolupa en cascada ja que trobem tasques que depenen de que es finalitzin tasques anteriors (per desenvolupar les funcionalitats és necessari haver desenvolupat prèviament el menú i la pantalla principal). D'altra banda, el gran nombre de funcionalitats que cal implementar fa adequada una metodologia iterativa per tal de desenvolupar un mínim de funcionalitat de cada mòdul i millorar-la en futures iteracions.

Per aquests motius es divideixen les tasques en: mòdul d'estructura (menú i pantalla principal) i mòduls de funcionalitats. S'avança el projecte iterant mòdul a mòdul fins a tenir una funcionalitat mínima. Un cop aconseguida aquesta funcionalitat mínima es tornen a iterar els mòduls posant més o menys èmfasi dependent del resultat obtingut.

En conclusió, s'ha escollit una metodologia de desenvolupament incremental i iterativa en què s'avança ràpidament en l'obtenció de resultats i posteriorment s'itera cada mòdul fins a dotar-lo de la qualitat requerida.

## 5 DISSENY

En aquest apartat s'analitza el model de disseny de l'aplicació web parant atenció a l'estructura i paradigma de desenvolupament. Posteriorment es presenta l'estructura de la interfície d'usuari i es justifiquen les decisions preses en aquest disseny.

### 5.1 Disseny web

En aquesta secció s'analitza el model MVC que es segueix en el disseny de l'aplicació web així com la relació amb la base de dades. El Model-Vista-Controlador és un paradigma de desenvolupament web que separa la capa de dades, la lògica de negoci i la interfície de l'usuari en diferents components.

MVC és un dels paradigmes de desenvolupament més utilitzats actualment per la quantitat d'avantatges que suposa [13]. El principal avantatge d'MVC és que aïlla les diferents capes de la aplicació fent-les independents. D'aquesta manera es pot substituir alguna de les parts sense haver de modificar les altres. Per aquest motiu s'ha optat per desenvolupar la web seguint aquest paradigma. D'aquesta manera es disposa d'un model que controla les dades de sessió, accés a base de dades, enviament de correus electrònics... Un controlador que gestiona les vistes que s'han de mostrar, l'idioma... I un conjunt de vistes que permeten visualitzar tota la informació.

La base de dades no requereix cap disseny donat que les dades que es requereixen ja es troben en una base de dades pròpia del departament. Aquesta base de dades està implementada en PostgreSQL.

En conclusió, el patró MVC permet simplificar el desenvolupament de les aplicacions web aïllant els components en diferents capes independents. Per aquests motius es desenvoluparà el *deIC Kiosk* seguint el model MVC. L'aplicació web consulta informació d'una base de dades implementada en PostgreSQL.

### 5.2 Disseny de la interfície d'usuari

En aquesta secció s'explica i justifica l'estructura del portal web i les diferents pantalles incloent imatges. Primer es presenta el menú i les àrees on es mostra la informació. Seguidament es presenta la pantalla principal. Per últim es tracta la manera en que s'estructura la informació a les diferents pantalles de funcionalitat.

L'aplicació web disposa d'un menú desplegable a la part superior, que, en cas d'executar-se al quiosc es mostra a la part inferior. La resta de l'espai s'utilitza per mostrar el contingut que apareix sempre centrat verticalment.

Un bon disseny ha de complir diversos criteris, un d'ells és que sigui el més *estàndard* possible i segueixi el patró de les webs similars [5]. Per aquest motiu, i observant altres webs i portals s'ha situat el menú horitzontalment a la part superior de la web. Aquest criteri, però, és diferent quan es visualitza al dispositiu quiosc donat que es mostra a la part inferior per tal d'assegurar l'accés a persones de baixa estatura. La resta de l'espai queda destinat al contingut de la pantalla tal i com es mostra a la figura 1.

El menú incorpora els colors del departament i és de tipus desplegable per tal d'encabir totes les funcionalitats. Es pot

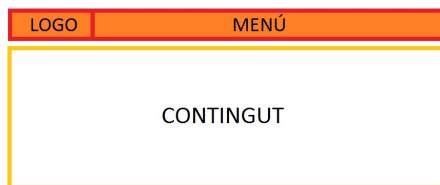


Fig. 1



Fig. 2

visualitzar una imatge del menú en a la figura 2

La pantalla principal és la primera pantalla que es visualitzarà i ha de ser atractiva, cridar l'atenció i mostrar les principals funcionalitats que pot oferir el quiosc. Per aquest motiu, com es pot veure a la figura 3, la pantalla principal disposa de diferents panells que referencien a cadascuna de les funcionalitats del punt d'informació.

Mantenir diferents panells per cada funcionalitat és necessari per assegurar la modularitat de l'aplicació definida a la metodologia i que facilitarà la incorporació o retirada de continguts.

La resta de pantalles de l'aplicació segueixen totes la mateixa estructura. Disposen del títol del mòdul que ofereix la funcionalitat a la part superior. Utilitzen panells per separar els continguts i, en cas d'haver de gestionar massa informació, s'organitzen en pestanyes o *tabs*.

Els colors utilitzats segueixen la mateixa línia que al menú i pantalla principal i es mostren centrades al mig de la pàgina en la divisió de contingut. A la figura 4 es pot visualitzar un exemple.

És a dir, es disposa d'un menú principal i una divisió per a mostrar els continguts. Inicialment, en la divisió de continguts es mostra una pantalla principal on es poden visualitzar les diferents funcionalitats del punt d'informació. Tota l'aplicació segueix el mateix estil i la mateixa línia gràfica.

En conclusió, l'aplicació web està dissenyada seguint el model MVC i es connecta a una base de dades ja implementada en *PostgreSQL*. El disseny de la interfície d'usuari pretén assimilar-se a altres tipus de webs similars i disposa de dos zones: menú i contingut.

## 6 IMPLEMENTACIÓ

En aquesta secció s'analitza la feina realitzada en la implementació de l'aplicació a partir de l'anàlisi i el disseny realitzats anteriorment. Primer s'analitza quina configuració ha requerit la màquina que executa el quiosc, seguidament es parla de l'estructura de la web i, per últim, de les tecnologies utilitzades per al seu desenvolupament.

### 6.1 Configuració del maquinari

La correcta configuració de la màquina *kiosk* que executa el portal és un dels punts que cal tenir en compte per assolir el projecte amb èxit. És necessari que la web es visualitzi correctament, sense permetre que l'usuari pugui sortir de l'aplicació web i aquesta s'inicialitzi de manera automàtica.



Fig. 3

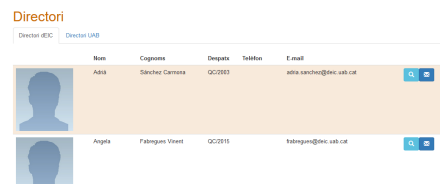


Fig. 4

Un altre aspecte a tenir molt en compte és la usabilitat i la gestió dels gestos tàctils.

Donat que l'ús de gestos tàctils, especialment en entorns d'escriptori, és relativament recent cal posar atenció als drivers i mòduls de control al sistema operatiu. És per aquest motiu que s'ha realitzat una instal·lació d'*Ubuntu 16.04 LTS* a la màquina *kiosk* que ja incorpora aquesta funcionalitat. Per tal d'executar l'aplicació web de manera autònoma a l'arrancada de la màquina ha sigut necessari definir un usuari sense contrasenya amb inici de sessió automàtic. Es requereix configurar el navegador, en aquest cas Google Chrome, perquè s'inicialitzi a l'engegar la màquina. És necessari, també, que quan s'executi el navegador aquest s'iniciï ocupant tota la pantalla i sense possibilitat de tancar-lo. Existeixen multitud de plugins que ofereixen aquesta funcionalitat; per aquest projecte s'ha optat pel plugin *Kiosk 5.7.6* [9].

Per últim, cal afrontar la problemàtica d'introduir text sense disposar de teclat físic. Per solucionar-ho s'ha instal·lat el plugin *Touch Kiosk Keyboard* [10]. És necessari seleccionar l'idioma *Spanish (QWERTY)* per tal de poder utilitzar el caràcter *ç*. A la figura 5 es pot visualitzar el teclat en execució.

En conclusió, és important posar atenció als drivers i configuració de la màquina que executarà el quiosc per evitar problemes de seguretat de la pròpia màquina i assegurar el servei. Existeixen, també, plugins per als navegadors que faciliten aquestes tasques.

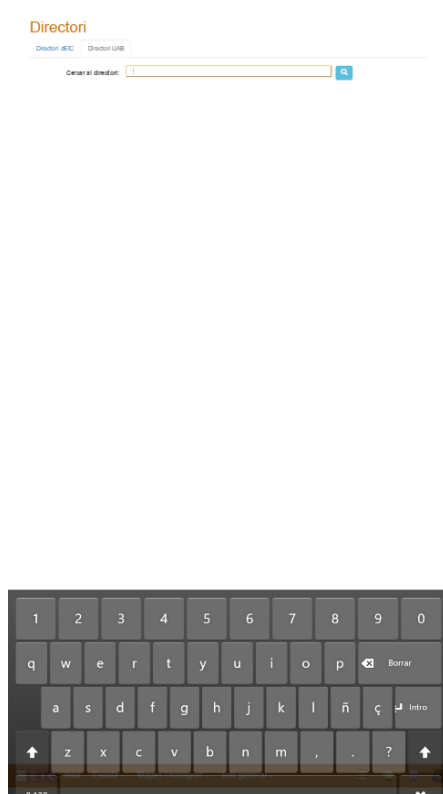


Fig. 5

## 6.2 Tecnologies web

En aquest apartat s'analitzen i justifiquen les principals tecnologies i tècniques utilitzades per al desenvolupament de l'aplicació. Les tecnologies i llenguatges esmentats són: PHP, Bootstrap, jQuery, web scraping i Google Maps.

- **PHP:** PHP és un llenguatge interpretat que s'executa a la banda del servidor i genera codi HTML que envia al client. El cos principal de l'aplicació web funciona amb PHP. Hi trobem implementat amb PHP:
  - Localització, gestió dels idiomes.
  - Gestió de les vistes.
  - Consultes a la base de dades.
  - Enviament de mails.
  - Web Scraping.
  - Gestió de continguts RSS

Existeixen diversos motius pels quals s'ha escollit PHP en comptes d'un altre llenguatge interpretat a la banda del servidor. En primer lloc, l'aplicació web que executava el quiosc anteriorment estava desenvolupada en PHP. Això ha permès reaprofitar part del codi (especialment part del model i controlador). En segon lloc, el servidor web disposava del mòdul de PHP. Utilitzar un altre llenguatge hagués suposat haver d'instal·lar un nou mòdul a la màquina servidor. Per últim, PHP permet utilitzar la tècnica de *web scraping* amb facilitat enviant peticions a webs externes i tractant les respostes.

En definitiva, l'ús de PHP ha permès reaprofitar el màxim de codi possible de l'anterior aplicació i ha su-

posat nombrosos avantatges per determinats mòduls de l'aplicació que han de rebre dades de webs externes.

- **Bootstrap:** Bootstrap és un framework per al desenvolupament web. S'utilitza al projecte per aconseguir:
  - Comportament responsive i adaptable als diferents dispositius.
  - Elements predefinitos com panells, *carousels*...
  - Menú de navegació.

Bootstrap s'utilitza, en primer lloc, perquè ofereix un comportament responsive sense quasi cap cost de desenvolupament donat que el framework s'encarrega d'oferir aquesta funcionalitat. A més, ofereix diversos elements com els panells, menús de navegació, *carousels*... molt útils en el desenvolupament web. El fet de disposar d'aquests elements ja desenvolupats simplifica la implementació i el disseny. En resum, Bootstrap permet dotar d'un comportament responsive a l'aplicació i ofereix elements ja desenvolupats que simplifiquen considerablement l'esforç de desenvolupament.

- **jQuery:** jQuery és una biblioteca Javascript que permet modificar l'estructura del DOM d'una web sense recarregar-la sencera. En aquest projecte s'utilitza per carregar les vistes internes de cada mòdul. Mentre la càrrega dels diferents mòduls de funcionalitat és controlada pel controlador PHP, el control de vistes interior de cada mòdul es realitza amb jQuery. També s'utilitza jQuery per modificar la posició del menú de navegació en funció del dispositiu donat que permet afegir i eliminar classes als objectes. El principal avantatge d'utilitzar jQuery és que es redueix la quantitat d'informació a enviar en cada petició donat que només es retorna la vista de contingut que es requereix.

En definitiva, jQuery permet carregar vistes enviant la informació mínima necessària i millorant el rendiment tant de la xarxa com del propi navegador del client que només haurà de recarregar una part de la web.

- **Web Scraping:** Una característica determinant del present projecte és la necessitat d'obtenir informació externa (directori de la UAB, horaris del transport públic...). Aquesta necessitat pot esdevenir un problema complex si el proveïdor de la informació no en facilita l'accés.

La forma més senzilla d'obtenir dades de tercers és consumint *webservices* [14]. És a dir, realitzant peticions a un servidor que les processa i retorna les dades en un format determinat. Per utilitzar aquesta tecnologia, però, el proveïdor de la informació ha de disposar d'un *webservice*. Els proveïdors d'informació que necessita aquest projecte no en disposen. Es fa necessari, llavors, trobar una alternativa per obtenir la informació externa. La solució implementada és el *Web Scraping*. Aquest mètode consisteix en realitzar una petició HTTP a la web que disposa de la informació i tractar la resposta per tal d'extreure la informació en qüestió. PHP està preparat per realitzar aquest tipus de tasques. Aquesta tècnica, però, disposa dels següents inconvenients:

- El contingut de la resposta HTTP pot tenir diversos formats i incloure la informació de maneres diverses. No sempre és possible separar aquesta informació de forma estàndard.
- Pot suposar un risc de seguretat si s'incrusta un script en la web externa que es parseja i aquest passa el procés de filtrat.
- Si el proveïdor canvia l'estructura de la seva web o la vista que serveix les dades el sistema deixarà de funcionar.

Per aquests motius és recomanable evitar el *Web Scrapping* sempre que sigui possible. Tot i així pot haver ocasions en què no quedin més alternatives. En conclusió, la metodologia més segura, eficient i recomanable per obtenir dades de tercers és l'ús de *webservices*. Aquesta tecnologia, però, no l'implementen tots els organismes que serveixen dades i per tant s'han de cercar altres alternatives com el *Web Scrapping* tot i els inconvenients que hi poden suposar.

- **Google Maps API:** Google ofereix diverses APIs per accedir als seus mapes des de webs externes i suficient documentació [12].

En aquest projecte s'utilitza l'API Javascript de Google Maps per mostrar un mapa de la universitat i calcular rutes en cotxe des del quiosc fins a la ciutat que l'usuari esculli.

Per fer ús d'aquesta API és necessari enregistrar-se a la web de Google i obtenir una clau. Es pot obtenir una clau gratuïta si l'ús dels mapes no excedeix un certs límits.

Utilitzar l'API de mapes de Google facilita l'ús de mapes a l'aplicació donat que permet mostrar-los, calcular i mostrar rutes... sense suposar un gran esforç de desenvolupament.

Es pot afirmar, per tant, que existeixen diverses tecnologies i frameworks que faciliten la feina de desenvolupament web. L'ús d'aquestes llibreries redueix el nombre de recursos necessaris per desenvolupar aquest tipus de projectes i suposa una certa estandardització que pot contribuir a millorar la usabilitat de les webs.

En conclusió, ha sigut necessari configurar la màquina física amb diferents plugins que faciliten la usabilitat de pàgines web en aquest tipus de dispositius. També s'han hagut de combinar diferents llenguatges i tecnologies per oferir les diferents funcionalitats i característiques del *dEIC Kiosk*.

## 7 PROVES I QUALITAT

En aquesta secció s'expliquen les diferents proves i test que s'han realitzat per assegurar el correcte funcionament del sistema així com les validacions i accions que s'han dut a terme per assegurar la qualitat del codi.

### 7.1 Proves

El testeig és una activitat important en el disseny i desenvolupament web ja que permet solucionar possibles errors i

vulnerabilitats en el codi. En aquest apartat s'expliquen les proves i test que s'han realitzat a la web.

Les proves realitzades han sigut de tipus manual. Han sigut descartats altres tipus de test més complexos com l'*automated test* per la seva complexitat en relació a l'abast del projecte. Durant les iteracions realitzades a cada mòdul (veure punt 4. Metodologia) s'han realitzat proves de caixa blanca per assegurar-ne l'estabilitat del mòdul. Aquestes proves han estat efectuades pel desenvolupador de l'aplicació web. Posteriorment, un cop integrats tots els mòduls, s'han realitzat proves de caixa negra ignorant el funcionament intern dels mòduls i assegurant-ne el correcte funcionament tant dels mòduls com del conjunt de l'aplicació. Per últim s'ha realitzat la posada en marxa de l'aplicació web en l'entorn real per tal de provar-la amb usuaris dels quals s'ha recollit feedback.

En conclusió, s'han realitzat diferents tipus de testeig a l'aplicació web que han servit per corregir alguns error i millorar la qualitat i fiabilitat del quiosc.

## 7.2 Qualitat

Per tal d'assegurar la qualitat del codi, que aquest compleix els estàndards i reduir el nombre de possibles errors s'han realitzat diverses validacions sobre el codi.

- **Validació HTML:** S'ha validat el codi de cadascun dels mòduls de l'aplicació web mitjançant el validador del *World Wide Web Consortium* [15].
- **Validació CSS:** S'ha validat el full d'estils *styles.css* amb l'eina de validació del *World Wide Web Consortium* per a CSS3 [16].

El codi, per tant, compleix amb els estàndards definits pel *World Wide Web Consortium* i compleix les guies d'estil definides pel consorci.

En conclusió, s'han realitzat proves manuals de caixa blanca sobre cada mòdul i de caixa negra sobre el conjunt de l'aplicació. Addicionalment s'ha adequat el codi per a què superi les validacions dels estàndards que utilitza (HTML5 i CSS3).

## 8 RESULTATS

En aquesta secció es presenten els diferents materials que s'han generat com a resultat del projecte. Aquest resultat és el *dEIC Kiosk*: un quiosc tàctil configurat per visualitzar i interactuar amb una web i una aplicació web que permet les funcionalitats requerides.

### 8.1 Aplicació web

En aquest projecte s'ha aconseguit desenvolupar una aplicació web responsive, és a dir, capaç d'adaptar-se a les diferents resolucions de pantalla. Aquesta aplicació web obté dades, tant internes de la base de dades del departament, com de webs externes (horaris de trens, notícies RSS...) La web, a més, compleix el format Model-View-Controller i ha passat les validacions d'HTML i CSS del *World Wide Web Consortium*.

## 8.2 dEIC Kiosk

Es disposa d'un quiosc que executa l'aplicació web anteriorment esmentada a l'iniciar l'equip en pantalla completa i permet fer zoom mitjançant un plugin del navegador. Per tal de permetre l'entrada de text a la web es disposa d'un altre plugin que mostra un teclat tàctil a la part inferior de la pantalla sota demanda, és a dir, cada cop que el focus de l'aplicació es situa sobre un quadre de text. Per últim, s'assegura el correcte funcionament dels gestos tàctils de polsar, arrossegar i fer zoom amb una versió recent del sistema operatiu Ubuntu.

Els resultats, per tant, són: un quiosc tàctil on es visualitza una aplicació web que proporciona informació diversa als usuaris i, addicionalment, es pot visualitzar en dispositius mòbils o d'escriptori. Aquest conjunt ha sigut anomenat *dEIC Kiosk*.

## 9 CONCLUSIONS

En aquest projecte s'ha assolit l'objectiu de desenvolupar un quiosc en format web que s'adapti als diferents dispositius i sigui capaç de mostrar, tant informació pròpia com externa.

Per desenvolupar un projecte d'aquestes característiques cal parar atenció a les diferents metodologies que existeixen. Existeixen, també, multitud de plugins per a navegadors que faciliten la usabilitat i resolen problemes com la possible manca de teclat.

El projecte portava associat uns determinats requisits. Aquest ha d'oferir informació de la seva base de dades però també de proveïdors externs. La web, a més, ha d'adaptar-se a la mida de pantalla dels diferents dispositius que la poden visualitzar i seguir els colors corporatius del departament. Per complir aquests requisits s'ha disposat d'una pantalla tàctil on s'executa el quiosc i 300 hores de desenvolupament del projecte assignades a un únic projectista.

Fruit de l'anàlisi dels requisits s'ha decidit seguir una metodologia de desenvolupament incremental i iterativa. S'ha dividit el projecte en diferents mòduls: un mòdul principal de menú i pantalla principal i diferents mòduls que ofereixen funcionalitats concretes. Cada mòdul s'ha desenvolupat de manera independent, realitzat una primera iteració sobre cada un d'ells per obtenir una funcionalitat bàsica i, augmentat després, el nombre d'iteracions per millorar i polir el funcionament.

L'estructura escollida per l'aplicació ha sigut un menú de navegació superior, seguint els principis d'usabilitat i una pantalla principal que dona accés a les diferents funcionalitats. El menú principal, en cas d'executar-se a la pantalla de quiosc es mostrarà a la part inferior de la pantalla assegurant l'accés per a persones de baixa estatura. La resta de pantalles seguiran una estructura similar a la pantalla principal, disposant d'un títol i organitzant el contingut en diferents pestanyes.

Un cop finalitzades les etapes d'anàlisi i disseny ha sigut necessari configurar la màquina que executarà el quiosc per a que reconegui els gestos tàctils i solucionar el problema de l'entrada de text sense teclat; han sigut molts útils alguns dels plugins del navegador. La web s'ha desenvolupat seguint el paradigma MVC que permet separar l'accés a les dades de la lògica de negoci i de la interfície de l'usuari. S'han utilitzat diferents tecnologies web per al desenvolupament



Fig. 6

del quiosc. L'estructura principal de la web està feta en PHP donat que permet gestionar les vistes, recursos d'idiomes, dades de sessió... La gestió de les vistes de cada funcionalitat, en canvi, utilitza jQuery que optimitza el tràfic de xarxa enviant únicament els objectes del DOM que s'han de refrescar.

Per últim s'han realitzat proves de caixa blanca durant les iteracions de cada mòdul i proves de caixa negra un cop cohesionats els mòduls. Per millorar la qualitat del codi s'ha adequat per passar les validacions del *World Wide Web Consortium*.

Com a resultat de la feina anterior s'ha aconseguit una aplicació web que s'executa al quiosc i s'ha generat la documentació necessària per millorar i continuar el desenvolupament d'aquest projecte.

### 9.1 Línies de millora

En aquesta secció s'analitzen algunes de les línies de millora en les quals es podria treballar per tal de millorar el producte. Algunes no han pogut realitzar-se ja que depenen de tercers i d'altres per manca de recursos assignats al projecte.

- **Ús de webservices:** La principal línia de millora que es pot aplicar en l'aplicació és l'obtenció de dades dels següents serveis per webservice:
  - Horaris Renfe
  - Horaris FGC
  - Directori UAB

El motiu de que en la primera versió no s'hagi realitzat d'aquesta manera és que aquestes fonts no disposen de webservice, per aquest motiu, actualment, es *parsejen* les webs respectives. Aquest mètode comporta una sèrie de riscos, el més important dels quals és que si canvien la web deixarà de funcionar el mòdul en qüestió del quiosc. Cal posar atenció al moment en que aquests serveis publiquin un webservice per fer les modificacions pertinents als mòduls afectats.

- **Google Maps Indoor:** Google Maps ha implementat, des de fa alguns anys, la possibilitat d'explorar mapes interiors dels edificis. L'Escola d'Enginyeria és un d'aquests edificis 6. Aquesta funcionalitat permet, fins i



tot, generar rutes entre despatxos o aules. Aquesta funcionalitat podria ser molt interessant per a l'aplicació web del quiosc però, no obstant, a data de publicació de la primera versió encara no s'ha alliberat cap API per manegar aquest tipus de funcionalitats. Cal prestar una mica d'atenció a la publicació de noves versions d'API de Google Maps per implementar, quan sigui possible, aquesta funcionalitat.

- **Inhabilitar pantalla completa al visor de PDF:** Per visualitzar PDF l'aplicació web del quiosc utilitza un iframe on carrega el Google PDF Viewer. Aquest visor porta implementat un mode en pantalla completa. Tot i que no es tracta d'una situació crítica no és adequat disposar d'aquest botó en la visualització del quiosc donat que es podria arribar a utilitzar per sortir de la web. Cal explorar diferents alternatives per evitar aquest risc.

Existeixen, per tant, diverses millores que es poden realitzar. La majoria, però, depenen de que terceres entitats comparteixin informació donant més facilitats per reproduir-la i utilitzant tecnologies estàndards.

En conclusió, s'ha assolit l'objectiu principal de desenvolupar un kiosk basat en web. També s'han assolit els subobjectius del treball; relacionats amb la investigació i estudi de les diferents tecnologies i metodologies per al desenvolupament del quiosc.

## REFERÈNCIES

- [1] M. Toledo, "Kiosko Multimedia para Consulta y Emisión de Certificados Académicos de la Universidad Politécnica Salesiana.", Universidad Politécnica Salesiana, Cuenca, Ecuador, 2009
- [2] C. L. García i J.M. Libreros, "Sistema interactivo de información para centros comerciales.", Universidad Autónoma de Occidente, Santiago de Cali, Colombia, 2014
- [3] Y. C. Pérez LLa gestión de contenidos en portales Web", Acimed, Ciudad de la Habana, Cuba, 2007
- [4] P. Cáceres i E. Marcos, "Procesos Ágiles para el Desarrollo de Aplicaciones Web", Universidad Rey Juan Carlos, Madrid, España, 2001
- [5] J. Nielsen, "Designing Web Usability: The Practice of Simplicity", Thousand Oaks CA: NRP, 1999
- [6] WebDesign Like. (11/12/2014) *16 Best Free Responsive HTML5 Frameworks 2016* [Online]. Disponible: <https://www.sitepoint.com/5-most-popular-frontend-frameworks-compared/> Accedit el 21/06/2016
- [7] Tiobe. (06/06/2016) *TIOBE Index for June 2016* [Online] Disponible: [http://www.tiobe.com/tiobe\\_index](http://www.tiobe.com/tiobe_index), 2016 Accedit el 21/06/2016
- [8] Ó. Campos. (28/09/2011) *Diferencia entre layout fluid y responsive* [Online]. Disponible: <http://www.genbetadev.com/desarrollo-web/diferencia-entre-layout-fluid-y-responsive>
- [9] Zebradog. (14/06/2016) *Kiosk 5.7.2* [Online]. Disponible: <https://chrome.google.com/webstore/detail/kiosk/afhcomalholahp1bj> Accedit el 05/06/2016
- [10] Kore Innovations. (27/06/2014) *Touch Kiosk Keyboard* [Online]. Disponible: <https://chrome.google.com/webstore/detail/touch-kiosk-keyboard/lfaiipcbikbnfcgcmaldlacamgekmb?hl=es> Accedit el 06/06/2016
- [11] T. Esteban. (19/11/2012) *Diferencias entre diseño líquido y diseño responsive* [Online]. Disponible: <http://visibilidad-trafico-conversion.com/2012/11/19/diferencias-entre-diseno-liquido-y-diseno-responsive/>
- [12] Google Inc., Google Maps Javascript API V3.24, Maig 2016
- [13] Yenisleidy Fernández Romero i Yanette Díaz González, "Patrón Modelo-Vista-Controlador", Revista Telem@tica, Vol. 11, pp. 47-57, Gener 2012
- [14] Carlos Andrés Morales Machuca, "Estado del Arte: Servicios Web", Universidad Nacional de Colombia, Bogotá, Colombia, 2008
- [15] World Wide Web Consortium, "HTML 5.1 Standard", <https://www.w3.org/TR/html/introduction.html>, Accedit el 06/06/2016
- [16] CSS Working Group of World Wide Web Consortium. (06/07/2016) *CSS Current Work* [Online]. Disponible: <https://www.w3.org/Style/CSS/current-work> Accedit: 06/06/2016
- [17] Firefox Add-Ons. (02/05/2016). *mKiosk: Modo de kiosko para Firefox* [Online]. Disponible: <https://addons.mozilla.org/es/firefox/addon/mkiosk/>