

Desenvolupament d'una xarxa social d'esquí per a mòbil: SocialSKI

Marc Ferret Font

Resum — A Internet existeixen diversos portals web que permeten conèixer l'estat de les pistes d'esquí, cadascun d'ells amb un funcionament similar, però alhora diferents entre ells. L'objectiu d'aquest projecte és desenvolupar una xarxa social per a dispositius mòbils que a part de donar la possibilitat als usuaris de consultar un conjunt de dades sobre les diferents pistes d'esquí, també puguin aportar una opinió personal seguint uns criteris dins d'un formulari. A partir d'aquestes dades aportades, el sistema calcula una conclusió a través de l'algorisme de lògica difusa. Aquest algorisme permet obtenir transparència i interpretabilitat sobre les opinions dels usuaris. Aquest projecte ha nascut després de l'anàlisi de les diferents aplicacions relacionades amb el camp de l'esquí i la necessitat per part de la societat de poder accedir a totes aquestes dades mitjançant una aplicació instal·lada en el dispositiu mòbil. Un cop finalitzada aquesta primera versió d'aquest projecte, els usuaris poden accedir a una xarxa social des de qualsevol de les diferents plataformes existents per a dispositius mòbils (Android, iOS, entre altres) i, així poder conèixer en exactitud l'estat de les pistes i millorar la seguretat en aquest esport.

Paraules clau — Android, AngularJS, Aplicació, CSS3, Framework, HTML5, iOS, Mòbil, Multi-plataforma, Muntanya, Neu, SKI

Abstract— On the Internet there are several websites that provide an insight into the conditions of the ski resorts, each of them with similar functions, but different at the same time. The aim of this project is to develop a social network for mobile devices that in addition to providing to the final users the opportunity to know a range of data about the different ski slopes, it will also give a personal opinion to the resort conditions provided by the other users. From the data fed by the users, the system calculates an outcome through a fuzzy logic algorithm. This algorithm allows a better transparency and understanding of users feedback. This project was born from the analysis of various applications within the field of skiing and the need of society to be able to access all this data via a single application downloaded to a mobile device. Once this first version of the project is finalized, the users will be able to access a social network from any of the currently existing platforms for mobile devices (Android, iOS, etc.) and therefore, be able to know exactly the status of the tracks and improve the overall sport's safety.

Index Terms — Android, AngularJS, Application, CSS3, Framework, HTML5, iOS, Mobile, Multi-platform, Mountain, Snow, SKI



1 INTRODUCCIÓ

Avui en dia per a molts amants de la pràctica de l'esquí i altres modalitats d'esports de neu la jornada comença consultant quin és l'estat de les pistes per tal de decidir quin serà el seu destí. Actualment consultar l'estat de les pistes es pot fer en diferents portals web o aplicacions per a dispositius mòbils, gestionades per la mateixa estació d'esquí o per portals que mostren dades de múltiples estacions d'esquí. Aquests portals faciliten un conjunt de dades proporcionades per les mateixes estacions, així els usuaris poden decidir el que més els convé. En termes generals, les estacions d'esquí acostumen a aportar les següents dades: temperatura, el gruix de la neu, el nombre de pistes obertes, quilòmetres esquiables, la velocitat de les ràfegues de vent, l'estat de la neu i per curiositat, el preu del forfait per tal de poder-se adaptar a tots els perfils dels usuaris.

- Email de contacte: marcferret@gmail.com
- Menció realitzada: *Tecnologies de la Informació*.
- Treball tutoritzat per Joan Bartrina
- Curs 2015/2016

Sovint es dona el cas que la informació que donen les estacions d'esquí no representa la realitat.

Per exemple, el gruix de la neu moltes vegades es mesura sobre un punt concret de l'estació d'esquí. Però, és aquest gruix uniforme per a tot el domini esquiable? en moltes ocasions la resposta és no. El mateix passa amb la qualitat de la neu.

Per aquests motius, es va veure la necessitat de crear una aplicació mòbil que aportés informació el més ajustada possible a l'estat real de les pistes. Aquesta informació no pot ser aportada pels responsables de les estacions d'esquí de forma oficial, sinó que ha de ser subministrada pels propis usuaris finals que visiten els resorts.

SocialSKI és una xarxa social que neix amb l'objectiu de centralitzar les opinions personals de tots els usuaris sobre l'estat de les estacions d'esquí. En aquesta primera versió ens centrem en les estacions Catalanes i Andorranes.

SocialSKI permetrà comparar i unificar totes les pistes d'esquí en una sola aplicació.

Actualment, cada pista d'esquí disposa d'una pàgina web i d'una aplicació pròpia on l'estació hi actualitza les seves dades a l'inici de la jornada.

Les pistes que es poden trobar, de moment, en aquesta xarxa social són: La Molina, Vallnord, GrandValira i Baqueira Beret.

SocialSKI obté informació de les pistes d'esquí a partir de les dades subministrades pels usuaris.

Una de les característiques principals de SocialSKI és que la informació que s'hi mostra és aportada pels propis usuaris a través dels seus comentaris d'opinió que es realitzen mitjançant un formulari amb unes respostes predeterminades. Aquestes respostes són processades pel sistema mitjançant l'algorisme de lògica difusa. L'ús de la lògica difusa permet obtenir transparència i interpretabilitat sobre les opinions aportades pels usuaris.

Després d'una recerca exhaustiva de diversos serveis web i aplicacions per a dispositius mòbils, tenint en compte les característiques que es consideren necessàries per a conèixer l'estat de les pistes i proporcionar-ne una opinió. Infonieue [1] seria considerada la capdavantera pel que fa a la relació entre l'estat real de les pistes i el fet de proporcionar informació sobre aquestes. Infonieue [1] és una plataforma web que també informa de l'estat de les pistes i dóna la possibilitat als usuaris d'aportar opinions personals. Tot i així, es pot observar a la seva plataforma que no hi ha cap comentari.

Com s'ha dit anteriorment, SocialSKI pretén aportar aquella informació per la pràctica de l'esquí que no es subministra pels canals actuals i que realment interessa als esquiadors. Per exemple, 1) independentment del gruix de neu, el que interessa a l'usuari final és si hi ha pedres o altres elements no coberts per la neu que puguin obstaculitzar la pràctica de l'esquí. 2) Si la neu que hi ha permet la pràctica de l'esquí fora pista. 3) Si el tipus de neu és uniforme per tota l'estació o és canviant... A més a més, una cop l'usuari està a la pista, aquest pot comentar sobre l'estat d'aquesta amb total comoditat, ja que només ha d'iniciar l'aplicació instal·lada al dispositiu, que es comporta com una aplicació nativa, i aportar la seva pròpia opinió.

D'aquesta manera l'ús de SocialSKI permetrà aconseguir una millora en la comunicació durant la jornada d'esquí, un increment de la seguretat, l'obtenció de dades fiables i més rapidesa en la presa de decisions a l'hora d'escollir la pista desitjada segons les necessitats.

La resta de treball es troba estructurat de la següent manera: La secció 2 fa un recull d'algunes aplicacions existents i la millora que aporta SocialSKI. A més a més, es contemplen les possibles tecnologies per a desenvolupar l'aplicació. La secció 3 descriu l'aplicació SocialSKI pel que fa a objectius, requeriments funcionals i no funcionals.

2 ESTAT DE L'ART

Actualment, hi ha diverses plataformes web que permeten consultar l'estat de les diverses pistes d'esquí en un mateix lloc, com Infonieue [1], Tele-SKI [2], Esqui.Sport [3], SkiInfo [4], entre d'altres. Tots aquests portals proporcionen diferents dades sobre les pistes d'esquí, com són: la temperatura, velocitat del vent, gruix de la neu, nombre de pistes obertes, quilòmetres esquiabls i fins i tot,

càmeres en directe en diferents punts de l'estació. Aquests serveis web són *online* i, per tant, és necessari accedir-hi a través d'un navegador amb connexió a Internet. Cadascun d'aquests portals tenen característiques diferents, tot i que la seva funcionalitat principal és mostrar la informació proporcionada per l'empresa que gestiona cada pista d'esquí als usuaris, sense possibilitat de comparar amb altres dades, com per exemple, les proporcionades pels usuaris.

A més a més, al món on vivim gairebé tothom té a les seves mans un telèfon mòbil. Tot i així, aquests dispositius utilitzen diferents plataformes/sistemes operatius i aquests han estat desenvolupats amb llenguatges de programació diferents, és a dir, les aplicacions que es publiquen al mercat depenen del dispositiu a que es vulguin adreçar. Per tant, per cada plataforma s'haurà de desenvolupar una aplicació nativa, que es dissenya i es programa específicament per a cadascuna d'elles. Alguns exemples dels llenguatges més utilitzats en el procés de desenvolupament d'una aplicació per un dispositiu mòbil són:

- **Objective-C i Swift:** aquests dos llenguatges de programació són dues opcions per desenvolupar aplicacions natives (específiques i a mida) pel sistema operatiu iOS. Aquest sistema és gestionat per *Apple* i ocupa gairebé el 17% de participació dins del mercat [5].
- **C#:** aquest llenguatge de programació s'utilitza per desenvolupar aplicacions natives pel sistema operatiu Windows Mobile. Aquest sistema és gestionat per *Microsoft* i ocupa gairebé el 6% del mercat [5].
- **JAVA:** s'utilitza per desenvolupar aplicacions natives pel sistema operatiu Android, gestionat per *Google Inc.* Amb aquest llenguatge de programació, es desenvolupen les aplicacions més populars, Android ocupa aproximadament el 68% de participació del mercat [5].
- **HTML5+CSS3+AngularJS:** utilitzats en la programació web i aplicant aquests llenguatges al desenvolupament d'una aplicació nativa per cada plataforma, formen un framework open-source anomenat Phonegap. A partir d'aquest framework podem desenvolupar una aplicació híbrida, és a dir, una aplicació que s'instal·la al dispositiu amb la mateixa forma o aspecte que una aplicació nativa, però aquesta conté una capa intermèdia que incorpora funcionalitats que fan ús d'altres llenguatges de programació. El nucli de les aplicacions desenvolupades amb el framework PhoneGap està format per llenguatge web. A partir d'aquest i utilitzant l'API (Application Programming Interface) del framework es pot accedir mitjançant AngularJS a característiques natives del dispositiu. Així, un desenvolupador pot accedir i controlar elements nadius del dispositiu des d'una aplicació híbrida, és a dir, no desenvolupada específicament per aquella plataforma.

Objectiu	Descripció de l'objectiu
Proporcionar uns criteris bàsics als usuaris per a determinar l'estat de les pistes d'esquí.	Mostrar a l'usuari un formulari basat en uns criteris concrets (si hi ha pedres a la pista, si n'hi ha a fora pista, l'estat de la neu, etc) per a determinar l'estat en que es troben les pistes. A partir d'aquests criteris, el sistema podrà calcular mitjançant un algorisme, una conclusió final que es proporcionarà als usuaris.
Fomentar que els usuaris siguin els que proporcionin una opinió a la xarxa social.	Mitjançant un formulari amb unes respostes predefinides i ràpides de respondre que permeten al sistema calcular-ne una conclusió. A més a més, els usuaris en poden obtenir un detall de la conclusió calculada a partir de totes les opinions.
Algorisme de lògica difusa.	Mitjançant els comentaris d'opinió proporcionats pels usuaris el sistema en calcula una conclusió mitjançant aquest algorisme, executant-lo amb els paràmetres d'entrada els comentaris dels usuaris.
Donar a l'usuari la possibilitat de comparar l'estat real de les pistes amb la conclusió calculada.	A partir de les opinions proporcionades pels usuaris, el sistema n'obtindrà una conclusió. L'usuari podrà comparar amb les dades reals proporcionades per la pròpia pista d'esquí.
Crear una aplicació que permeti conèixer en temps real diferents detalls i aspectes rellevants per poder gaudir d'una bona jornada d'esquí.	Com ho són la temperatura, el gruix de la neu, el nombre de pistes obertes, la velocitat de les ràfegues de vent, l'estat de la neu i per curiositat, el preu del forfait per tal de poder-se adaptar a tots els perfils dels usuaris.
Proporcionar més facilitat en la presa de decisions sobre una jornada d'esquí.	Proporcionar unes dades clares que ajudin a planificar una jornada d'esquí segons les preferències dels usuaris.
Ser multiplataforma.	L'aplicació serà compilada només una vegada perquè es pugui utilitzar per a les diverses plataformes (Android, IOS, etc).
Aplicació nativa.	L'aplicació instal·lada en les diferents plataformes ha de ser nativa, és a dir, s'ha d'adaptar de la millor manera possible al sistema operatiu que utilitzi el dispositiu mòbil.
Aconseguir un increment de seguretat en la pràctica de l'esquí	Gràcies als comentaris actualitzats proporcionats pels usuaris, s'aconseguirà un increment de seguretat en les jornades d'esquí.
Iniciar sessió amb Facebook.	Mitjançant l'API de la xarxa social podem demanar i comprovar les credencials proporcionades per l'usuari i així accedir a SocialSKI amb aquestes dades.

Taula 1: Objectius Específics

Davant la gran oferta de dispositius que hi ha al mercat i alhora, la gran necessitat d'oferir l'aplicació al màxim nombre de plataformes, principalment per Android i iOS, desenvolupar aquesta aplicació per cadascuna de les plataformes suposa començar-lo de nou tantes vegades com plataformes es vulguin oferir.

Per aquest motiu, aquest projecte es desenvolupa utilitzant llenguatge web, HTML5+CSS3+AngularJS. La utilització d'aquest framework permet dur a terme un sol desenvolupament i compilar-lo per cadascuna de les plataformes desitjades, i així, convertir-la en una aplicació nativa.

3 L'APLICACIÓ SOCIALSKI

3.1. OBJECTIU PRINCIPAL

L'objectiu principal del projecte és desenvolupar una aplicació per a dispositius mòbils, que es converteixi en una xarxa social. A més a més, un aspecte bàsic de l'aplicació és unir totes les aplicacions que es troben actualment al mercat, cadascuna d'una única pista d'esquí en una de sola. D'aquesta forma, a través de SocialSKI es podrà accedir als detalls de totes les pistes d'esquí des d'una sola aplicació.

3.2. OBJECTIUS ESPECÍFICS:

Els objectius específics d'aquest projecte es recullen en la Taula 1. Aquests estan ordenats de forma decreixent i d'aquesta manera, és representa la seva prioritat.

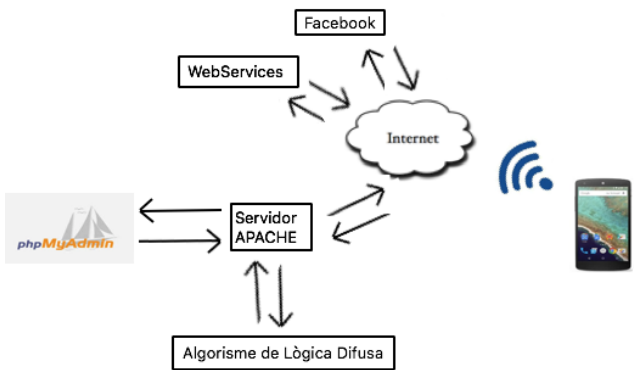


Figura 1: Representació del funcionament de l'aplicació.

3.3. ESQUEMA DE FUNCIONAMENT DE L'APLICACIÓ

A la figura 1 es mostra l'esquema de funcionament de l'aplicació. Un dispositiu mòbil té instal·lada l'aplicació SocialSKI de forma nativa, a l'executar-la cal iniciar sessió a la xarxa social. Aquest dispositiu està connectat per WiFi, o bé per dades de telefonia mòbil (3G, 4G, ...). En el cas que l'usuari no tingui la possibilitat d'accedir a Internet podrà accedir a l'aplicació de forma anònima on només tindrà la possibilitat de consultar les dades de les pistes d'esquí actualitzades anteriorment quan l'usuari tenia Internet. Si té connexió a Internet, l'usuari podrà iniciar sessió a l'aplicació utilitzant les credencials de Facebook o bé, les de la mateixa aplicació emmagatzemades en el servidor phpMyAdmin.

L'usuari pot consultar totes les dades. Aquestes s'actualitzen mitjançant els webServices. També, l'usuari, pot proporcionar la seva opinió a la xarxa social, un cop emmagatzemada, s'executarà l'algorisme de lògica difusa. Aquest algorisme està desenvolupat en un fitxer PHP allotjat en un servidor Apache.

3.4. REQUERIMENTS FUNCIONALS

- 3.4.1. **Log-in de l'usuari:** l'usuari a través d'un formulari iniciarà sessió a l'aplicació i podrà utilitzar totes les funcions que s'ofereixen sense necessitat d'iniciar sessió de nou.
- 3.4.2. **Registre a la Base de Dades:** l'usuari s'ha de registrar a l'aplicació a través d'un formulari per poder participar a la xarxa social. Aquest requisit no s'ha pogut implementar per falta de temps. Ara bé, el controlador ja conté tots els serveis preparats per poder fer les peticions al servidor i així, completar aquest requisit.
- 3.4.3. **Log-in a l'aplicació mitjançant Facebook:** l'usuari, mitjançant un clic a l'enllaç, entrarà a l'aplicació de Facebook on es descarregarà la seva credencial per poder accedir a la informació personal i així participar dins la xarxa social.
- 3.4.4. **Llistat general de les pistes d'esquí:** l'usuari en una sola vista ha de poder veure el llistat de pistes d'esquí que s'ofereixen en la xarxa. En aquesta versió només es contemplen estacions d'esquí catalanes i andorranes.

3.4.5. **Detalls sobre la pista d'esquí:** en el llistat de les pistes, l'usuari ha de poder visualitzar de forma clara la informació que s'ofereix, com per exemple: temperatura, nombre de pistes obertes, velocitat del vent i cost del forfait. De la mateixa manera, l'usuari ha de poder accedir a uns detalls més extensos i als diferents menús per donar una opinió.

3.4.6. **Botó back:** l'usuari donant clic en el botó "back" situat a l'extrem superior esquerre podrà tornar a la finestra anterior.

3.4.6.1. **Botó físic back d'Android:** Alhora, el sistema operatiu Android incorpora un botó "back" físic o incorporat a la pantalla que permet a l'usuari utilitzar-lo de la mateixa forma que el botó explicat en el punt anterior. En aquest cas, no s'ha aconseguit aquest requisit. El botó propi del sistema Android treballa de forma independent a l'aplicació i en fer-hi clic la tanca.

El motiu d'aquest error és causat per la diferència existent entre l'Activitat Principal, programada en Java, i la navegació dins l'aplicació que es duu a terme mitjançant el llenguatge AngularJS [6] (variant de JavaScript propi de Google). El botó "back" del sistema detecta que l'usuari està a la pantalla principal i en donar-li clic tanca l'aplicació. En aquesta versió s'han realitzat diferents proves per solucionar aquest problema, tot i que de moment sense èxit.

3.4.7. **Donar opinió sobre la pista d'esquí en concret:** l'usuari proporciona la seva opinió mitjançant un simple formulari. Aquesta resposta proporciona els criteris necessaris per conèixer de forma aproximada l'estat de la pista d'esquí desitjada.

3.4.8. **Mostra de les conclusions:** l'usuari podrà consultar un resultat de les opinions d'altres usuaris ja tractades. A partir d'aquestes opinions, podrà extreure'n unes conclusions personals. Les opinions dels usuaris són tractades mitjançant l'algorisme de lògica difusa que permet obtenir una conclusió sobre aquestes. Aquest algorisme està desenvolupat al punt 3.2.3.

3.4.9. **Aplicació multi-plataforma:** l'aplicació haurà de ser compatible i interactiva en les plataformes: Android i iOS.

3.5. REQUERIMENTS NO FUNCIONALS

3.5.1. **Comprovació de les dades d'usuari a la Base de Dades:** una vegada l'usuari hagi enviat les dades en el formulari, el controlador recollirà les dades de la vista per enviar-les al model i així comprovar-les. Finalment, rebrà una resposta a la petició per poder donar-li accés i iniciar la sessió. Si les dades no són correctes, no s'obrirà la sessió.

3.5.2. Emmagatzemar les dades a la Base de Dades: una vegada l'usuari hagi omplert el formulari d'opinió sobre una pista d'esquí aquesta s'emmagatzemarà a la Base de Dades.

3.5.3. Algorisme de Lògica Difusa: s'utilitza per calcular una conclusió a partir dels comentaris proporcionats pels usuaris a través del formulari de l'aplicació i del nombre de comentaris fets per cada usuari. És a dir, l'algorisme dóna més pes als comentaris fets pels usuaris que han fet més comentaris en total a l'aplicació [7]. D'aquesta forma, podem suposar que els usuaris que més comentaris han aportat a l'aplicació són més experts. Per tant, les seves opinions, segurament, seran més encertades que un usuari que fa proves a l'aplicació. Per aquest motiu, els comentaris dels usuaris més experimentats dins l'aplicació tenen més pes sobre les opinions d'un altre usuari.

S'utilitza aquest algorisme de lògica difusa amb l'objectiu d'aconseguir transparència i interpretabilitat sobre les opinions aportades pels usuaris. [8]

Els models de lògica difusa es poden classificar en dos tipus:

- Dependent del nombre de variables d'entrada i sortida:
 - o **SISO** (Single Input, Single Output): formats per una única variable d'entrada i una de sortida.
 - o **MIMO** (Multiple Input, Multiple Output): formats per un conjunt de paràmetres d'entrada i de sortida.
 - o **MISO** (Multiple Input, Single Output): format per un conjunt de paràmetres d'entrada i una sola variable de sortida.
- Dependent de les variables involucrades:
 - o **Regles completes:** totes les regles que formen el model difús estan involucrades les mateixes variables d'entrada.
 - o **Regles incompletes:** són regles formades per subconjunts de dades, no necessàriament iguals, del conjunt total de variables.

L'algorisme de lògica difusa aplicat en aquest projecte està basat en el model difús MISO. Aquest calcula la sortida, és a dir, la conclusió, amb un conjunt de paràmetres d'entrada proporcionats per les opinions dels usuaris i el nombre de comentaris realitzats per cadascun [8].

A més a més, l'algorisme es basa en regles incompletes, ja que aquest es repeteix diverses vegades per cada un dels subconjunts de dades aportades pels usuaris. La fórmula utilitzada en aquest algorisme i així, extreure'n una conclusió és:

$$y = \frac{\sum_{i=1}^c w_i R_i(x)}{\sum_{i=1}^c R_i(x)}$$

Mitjançant aquesta fórmula obtenim la conclusió (y) amb el càlcul del sumatori del producte de l'opinió (w_i) pel nombre de comentaris realitzats (R_i) de cada usuari dividit pel sumatori del nombre de comentaris (R_i) realitzats pels diferents usuaris.

Aquest càlcul es repeteix diverses vegades, tantes com variables es contestin al formulari per obtenir la informació proporcionada per l'usuari sobre l'estat de les pistes d'esquí. [9-11]

3.5.4. Petició de Mostra dels Resultats: cada vegada que l'usuari demani la mostra dels resultats, es farà una petició al model perquè proporcionin aquestes dades. Les dades, tractades i actualitzades, es mostraran a l'usuari.

3.5.5. Obtenir dades actuals: es faran peticions cada 5 minuts per aconseguir dades més exactes sobre la meteorologia, estat de pistes, etc. Aquest requisit no s'ha pogut aconseguir, ja que els serveis que proporcionen aquesta informació són privats i tenen un cost. S'ha contactat amb diferents proveïdors que permeten accedir a la seva API de forma gratuïta per desenvolupadors i així obtenir les dades necessàries, com per exemple InfoNieve¹, MountainNews², WeatherUnlocked³, entre altres. Tot i haver contactat amb les diferents empreses a través de diferents vies (mail, xarxes socials, formularis, etc.) encara s'espera una resposta. Per simular aquestes dades, en aquesta primera versió s'han utilitzat constants.

3.5.6. Límit de Comentaris: l'usuari podrà comentar com a màxim 2 cops al dia. Aquesta restricció és necessària per evitar un cas d'ús fraudulent de l'aplicació.

3.5.7. Temps de resposta: establim com a màxim, amb la capacitat actual, un temps màxim de 5 segons entre la petició i la resposta.

¹ <http://www.infonieve.es/recursos/>

² <http://mountainnews.com/>

³ <https://developer.weatherunlocked.com/skiresort>



Figura 2: Representació d'un webservice que interactua amb el servidor.

3.5.8. Escalabilitat: tot i que AngularJS ens proporciona bons índexs d'escalabilitat i rendiment, s'està treballant en aquest aspecte per millorar l'escalabilitat en la base de dades i en el servidor, on s'hi executa el model. Tot i haver implementat aquest projecte amb la versió 1 d'AngularJS, desenvolupada per Google. També, en versió beta, existeix la versió 2 d'aquest llenguatge que conté una millora considerable en aquest aspecte.

3.5.9. Actualitzar el registre diàriament: diàriament s'ha d'actualitzar el registre numComentariosDiaris a valor 0 perquè al començar un nou dia els usuaris puguin donar de nou les seves opinions a la xarxa social.

3.6. FRONTEND

Per implementar aquesta aplicació és necessari un *front*, és a dir, la capa on s'implementen i es controlen totes les accions que es mostren a l'usuari.

En el moment de desenvolupar la part del client, era necessari utilitzar una eina que fos compatible amb Phonegap. Aquest ens dona la possibilitat de transformar l'aplicació d'híbrida a nativa per cada dispositiu. A més a més, es necessari comptar amb un codi *opensource* que donés una certa flexibilitat per modificar els elements d'aquesta. Per aquest motiu, es va seleccionar el framework OnsenUI per a implementar el FrontEnd de l'aplicació. Aquest framework compleix totes les característiques necessàries, com poden ser: *opensource*, totalment compatible amb Phonegap, Còrdova (nucli de Phonegap), JavaScript, jQuery i AngularJS. Aquesta eina també incorpora components que permeten dissenyar la interfície d'usuari de forma més fàcil. [12]

3.7. BACKEND

Per implementar aquesta aplicació és necessari un *back*, on s'implementen i s'executen les funcions que corresponen al servidor. Algunes característiques que es duen a terme en el servidor són:

3.7.1. WebServices

El servidor es comunicarà amb diferents webServices per obtenir-ne dades, aquests estan allotjats al *Cloud*. A la figura 2 es representa com un webservice té la capacitat d'interactuar amb un servidor. Un webservice és similar a una API (Application Programming Interface), ja que es pot considerar un tipus d'interfície



Figura 3: Esquema de peticions POST al Servidor.

que permet accedir als mètodes i objectes de forma remota. Un webservice utilitza diferents protocols de comunicació, com poden ser el SOAP o el REST que proporcionen les dades al servidor. Un dels avantatges d'aquesta capa és que s'hi pot accedir des de qualsevol punt i a més a més, és independent del sistema operatiu.

En aquesta aplicació, els webservice s'utilitzen per demanar dades que són necessàries a l'aplicació. Com per exemple previsions meteorològiques, estat de les pistes d'esquí, imatges de les càmeres, etc. Aquest punt s'ha preparat, tot i que no s'ha pogut implementar, ja que aquests serveis no són gratuïts. [13-14]

3.7.2. Peticions POST

L'aplicació està desenvolupada seguint l'estructura model-vista-controlador. El controlador desenvolupat en AngularJS es comunica amb el model desenvolupat en llenguatge PHP. S'utilitza PHP en el servidor per la facilitat, fiabilitat i velocitat que ens proporciona aquest llenguatge i així, complir els requisits per realitzar la comunicació entre el controlador i el model. Per dur a terme aquesta comunicació entre codi que s'executa al client i el model en el servidor s'utilitzen peticions POST. Per tant, el controlador conté un servei que és l'encarregat de fer les peticions al servidor i esperar-ne una resposta, en format JSON, per retornar-la a la funció. Així tractar-la per poder proporcionar-la a la vista.

A la figura 3 es pot observar una representació de peticions que realitza el controlador al servidor a l'espera d'una resposta.

3.7.3. Base de Dades

La base de dades allotjada en el servidor és *mysql*. S'ha utilitzat una base de dades relacional que contempla les relacions entre les taules i a més a més emmagatzema les files de les taules amb l'ordre exacte al generat.

3.8. EINES DE COMPILACIÓ

Phonegap va ser adquirit per Adobe. Tot i així aquest manté tot el codi i totes les llibreries *opensource*, amb el suport d'Adobe al darrere [15].

Adobe ens proporciona una eina anomenada Adobe Phonegap Build, online i gratuïta [16]. Aquesta eina és gratuïta amb unes certes limitacions. Permet compilar l'aplicació per les diferents plataformes. En cas de voler superar els límits, com pot ser desenvolupar més d'una

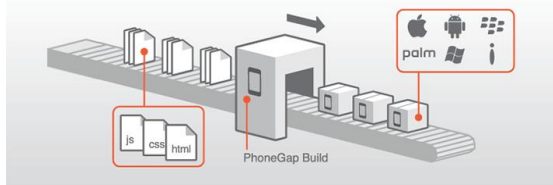


Figura 4: Representació de les entrades i sortides de l'eina Adobe Phonegap Build.

aplicació privada, és necessari contractar un pla de desenvolupament a partir de 9,99\$/mensuals.

Dins de l'aplicació web que ens ofereix la Adobe, només cal importar el projecte (fitxers *.html*, d'estils (*css*) i *.js*) en un fitxer comprimit en format *.zip*. Aleshores, només cal esperar que la plataforma compili l'aplicació i descarregar l'aplicació nativa per cada dispositiu i distribuir-la. A la figura 4 es representa la funcionalitat de Phonegap Build que permet compilar els fitxers desenvolupats en llenguatge web i transformar-los en aplicacions natives per les diferents plataformes existents pels dispositius mòbils. Tot i així, al compilar aquest projecte utilitzant aquesta eina de forma gratuïta apareixen diversos problemes. El principal problema apareix en els dispositius mòbils que utilitza el sistema operatiu Android, on el sistema no concedeix els permisos necessaris a l'aplicació perquè aquesta accedeixi a la connexió del dispositiu. És a dir, l'aplicació no té accés ni a la connexió Wi-Fi ni a la de dades. Per aquest motiu, l'aplicació no pot realitzar totes aquelles funcionalitats que necessiten l'ús del BackEnd de l'aplicació, com per exemple, el Login o actualitzar les dades, entre altres.

Davant d'aquest problema es proposen diferents solucions, com poden ser: cerca d'altres mètodes de compilació, planteig de contractar un pla (assumint el cost que aquest comporta), etc. A la vegada, s'intenta contactar amb la companyia on no s'obté cap resposta.

Finalment, es decideix compilar l'aplicació de forma manual i per cadascuna de les plataformes. Es compilarà l'aplicació mitjançant *Eclipse* per la plataforma Android. Definint els permisos necessaris en el fitxer *AndroidManifest.xml* aconseguim que el dispositiu no bloquegi les funcionalitats necessàries i així, aconseguir el òptim funcionament de l'aplicació.

Per compilar l'aplicació per la plataforma iOS s'utilitzaria XCode que ens permet codificar l'aplicació i convertir-la a nativa per tots els dispositius distribuïts per *Apple*.

4 METODOLOGIA

A l'hora de plantejar la metodologia a seguir per afrontar aquest desenvolupament ràpidament es va veure que eren necessàries dues fases. En la primera fase s'ha utilitzat la metodologia en cascada [17], ja que era necessari aprendre la tecnologia des de zero. És a dir, era necessari observar els requisits necessaris per començar a crear el projecte amb PhoneGap, seguidament crear-ne un de prova i analitzar que tots els passos s'havien fet de forma correcta.

Així, ja es pot començar a desenvolupar el projecte. Per la segona fase, la fase de desenvolupament del projecte, és necessari aplicar una metodologia àgil [18]. La idea és anar desenvolupant petites parts del projecte, analitzant prèviament l'estat actual per poder planificar i començar a desenvolupar el següent Sprint Backlog.

D'aquesta manera es pot aconseguir un entorn de treball dinàmic, on tant la planificació com les tasques poden canviar d'estat molt ràpidament, tant per problemes en el procés de desenvolupament com per una incorrecta estimació del temps necessari per completar aquella tasca.

Tot i així, aplicar una metodologia àgil no és fàcil. Per això, des del primer contacte amb el projecte es va crear un tauler de *Trello* [19]. *Trello* és una eina online que permet organitzar les tasques, seguint el model de Kanban [20]. Aquest model el va desenvolupar la companyia Toyota sobre els anys 40. El mètode de Kanban neix per millorar la gestió dels projectes. Transforma cada tasca en *kanbans*, és a dir, en senyals visuals o targetes. Aquestes passen a través de les tres columnes dibuixades en una pissarra que representen l'estat actual d'aquesta tasca: TO DO, DOING i DONE. D'aquesta forma es pot tenir un coneixement general de l'estat de totes les tasques d'un projecte. *Trello* substitueix la pissarra física per un tauler d'una plataforma online on les targetes es poden arrossegar per les diferents columnes del tauler, i així marcar el seu estat dins del projecte. A més a més, *Trello* permet crear tantes columnes com siguin necessàries i així adaptar el model de Kanban a tots els estats possibles de les tasques dins d'un projecte. D'aquesta manera, el temps invertit per pensar en quina tasca s'està treballant i quines depenen de les altres és molt més fàcil. A més, existeix la possibilitat de comentar les tasques per deixar més clars diferents aspectes que hi puguin afectar.

5 FEINA FUTURA

La primera versió de l'aplicació conté la funcionalitat descrita al llarg d'aquest treball. Un cop presentada aquesta, es plantegen unes propostes i uns objectius per a complir-se més endavant, que seran explicats a continuació.

Partint de la base de què una xarxa social té l'objectiu de proporcionar funcionalitats al màxim nombre de persones possible, en podem extreure la idea que és necessari que aquesta millori, mantingui i amplii funcionalitats per així aconseguir augmentar el nombre d'usuaris existents. Així, a partir de la finalització d'aquesta primera fase, comença immediatament una segona. Primer, caldrà investigar sobre algunes funcionalitats que no s'han pogut acabar d'implementar, com ser el botó físic del sistema operatiu Android per anar a la pantalla anterior. Alhora, també és important potenciar la compilació de l'aplicació pel sistema operatiu iOS.

Abans de continuar amb l'expansió i publicació de l'aplicació als principals mercats, com poden ser: Play Store (disponible per Android) i App Store (disponible pels usuaris que utilitzin el sistema operatiu iOS) és necessari contractar a un dissenyador web perquè millori el

disseny de l'aplicació i així potenciar l'accessibilitat i la facilitat d'ús d'aquesta.

Una vegada ja s'ha aconseguit una total compatibilitat en els diferents sistemes operatius per a dispositius mòbils, és important iniciar una fase de desenvolupament de l'aplicació que inclogui diferents idiomes i la possibilitat de consultar un nombre més elevat de pistes d'esquí.

Seguidament, en una tercera fase, començaria el desenvolupament d'una nova funcionalitat que no existeix en cap altra aplicació. Consisteix a introduir un mapa de Google Maps dins de l'aplicació. Google Maps inclou el recorregut de les diferents pistes d'esquí del món, a partir d'aquests registres i aprofitant el sistema de geolocalització que ens proporcionen tant els telèfons mòbils dels usuaris com Google es podria aconseguir una estimació del nombre d'esquiadors que hi ha en temps real a cada pista d'esquí. És a dir, coneixent la posició real dels usuaris i de cada pista d'esquí podem diferenciar en quin punt de l'estació es troba. D'aquesta manera, la xarxa social té la capacitat d'indicar quins són els punts de l'estació on hi ha més persones en un moment determinat. Així, un usuari que té la intenció de gaudir d'unes pistes amb poca gent, pot consultar l'aplicació i saber en temps real quines són les que l'hi interessa.

6 CONCLUSIONS

En aquest projecte s'ha desenvolupat una xarxa social anomenada *SocialSKI*. Aquesta aplicació ha estat desenvolupada a través de llenguatge web, la qual es converteix en una aplicació híbrida. D'aquesta manera, aquesta aplicació es pot compilar per cada sistema operatiu sense necessitat de desenvolupar específicament l'aplicació per cadascun d'ells. Així, sense cap cost de desenvolupament addicional es pot descarregar i instal·lar l'aplicació per diferents sistemes operatius, com per exemple Android i iOS. Aquest punt pot ser molt una millora molt gran per les petites i mitjanes empreses que tenen la necessitat de publicar al mercat una aplicació compatible amb diferents sistemes operatius sense dedicar-hi molts recursos. El fet de desenvolupar-la només una vegada i distribuir-la diverses vegades disminueix el cost de desenvolupament i augmenta la rendibilitat d'aquesta aplicació.

Actualment, el framework utilitzat per desenvolupar aquesta aplicació no ofereix molta informació, tot i que la base d'aquest és llenguatge web: HTML5+CSS3+AngularJS. Aquests són uns llenguatges molt potents i proporcionen al desenvolupador una capacitat de disseny gairebé infinita i compatible per tots els dispositius mòbils dels usuaris. Per tant, utilitzant aquest framework un desenvolupador pot crear qualsevol idea per a qualsevol dispositiu.

A nivell de metodologia, penso que a partir d'aquest treball he après a organitzar-me molt millor i a pensar com gestionar el temps dedicat per cada part d'aquest projecte. El fet d'haver-lo dividit en petites parts penso que ha estat més fàcil afrontar-lo des de bon principi. Començant des de la base del framework fins a desenvolupar l'aplicació passant per les fases intermèdies, fent evolucionar així l'aplicació i el projecte en general.

Personalment, la valoració del projecte és 100% positiva, ja que penso que s'han completat tots els objectius d'aquesta primera versió de l'aplicació i a més a més, s'ha preparat tot el que és necessari per continuar aquest projecte i publicar una segona versió d'aquesta en breus.

A més a més, com a esquiador i senderista de les grans muntanyes del nostre Pirineu crec que una aplicació com aquesta és necessària al nostre telèfon mòbil. Arriba un moment on un usuari habitual pot arribar a tenir desenes d'aplicacions, cadascuna d'una pista o d'una part del Pirineu. En canvi, utilitzant SocialSKI es podrien unificar totes aquestes en una de sola i així guanyar eficiència, accessibilitat a les dades i reducció de l'espai ocupat a la memòria del dispositiu.

Evidentment aquesta aplicació no l'hem llançat al mercat perquè encara necessita diversos canvis, millores i noves funcionalitats. Penso que el temps que hi dedica un usuari a determinar si l'aplicació es queda al seu dispositiu o la desinstal·la és de gairebé d'uns segons. Així mateix, és important llançar l'aplicació al mercat i per tots els dispositius quan sobre aquesta versió se n'hagin generat una segona o tercera. Tot i així destaco que aquest camp és nou, ja que no hi ha cap aplicació d'aquestes característiques i per tant, SocialSKI té un camp buit i on tot és possible. Per tant, personalment continuaré aquest projecte amb les mateixes ganes que el primer dia per poder aconseguir que en algun moment, SocialSKI, lideri com a xarxa social d'esquí en els dispositius mòbils.

7 AGRAÏMENTS

A la meua família, amics i companys de classe per donar-me suport en cadascuna de les parts d'aquest repte.

Als meus companys de Mango per donar-me consells i bones pràctiques en el procés de desenvolupament d'aquesta aplicació.

Al meu tutor, Joan Bartrina, per guiar-me durant aquests 4 mesos i ajudar-me a buscar les millors solucions per aconseguir l'èxit d'aquest projecte.

REFERÈNCIES

- [1] Infonieue: detall de les pistes d'esquí, <http://www.infonieue.es/estaciones-esqui/>, Març 2016.
- [2] TeleSki: detall de l'estat de la neu, <http://www.teleski.com/Inicio.html>, Juny 2016.
- [3] Esqui.sport: detall de les diferents pistes d'esquí espanyoles: <http://esqui.sport.es/>, Juny 2016.
- [4] Skiinfo: detall i comentaris sobre les pistes d'esquí, <http://www.skiinfo.es/>, Abril 2016.
- [5] Smartphone market share: Comparacions de la lluita entre els diferents plataformes per a dispositius mòbils, <http://money.cnn.com/2012/08/08/technology/smartphone-market-share/>, Maig 2016.
- [6] AngularJS: Superheroic JavaScript Framework, <https://angularjs.org/>, Març 2016.
- [7] D.Guzman, V.M.Castaño, "Lógica Difusa aplicada a la Ingeniería", <http://revistas.ucr.ac.cr/index.php/cienciaytecnologia/article/download/2640/2591>, Abril 2016.

- [8] Juan A. Botía Blaya y Mercedes Valdés Vela, "Lógica difusa y sistemas difusos al análisis inteligente de datos", http://webs.um.es/juanbot/miwiki/lib/exe/-fetch.php?id=tiia&cache=cache&media=clase_tiia5.pdf, Abril 2016.
- [9] J. Garzón Sanabria, "Clusterización utilizando lógica difusa", <http://orinoquia.unillanos.edu.co/index.php/orinoquia/articloe/view/179/641>, Maig 2016.
- [10] Algoritme de lógica difusa: Introducción a la teoría, <https://www.mql5.com/es/articles/1991>, Abril 2016.
- [11] Oscar G. Duarte, "Sistemas de lógica difusa - fundamentos", <https://www.mql5.com/es/articles/1991>, Maig 2016.
- [12] OnsenUI: Framework and UI components for Phonegap, <https://onsen.io/>, Febrer-Juny 2016.
- [13] Diferencia entre API y servei web, <https://ar.answers.yahoo.com/question/index?qid=20090817212317AAfaBxt>, Juny 2016.
- [14] API Web Services, https://docs.moodle.org/-dev/Web_services_API, Juny 2016.
- [15] Carlos Solis, "Adobe compra Phonegap... y ahora que?", <http://revolucion.mobi/2011/10/04/adobe-compra-phonegap-y-ahora-que/>, Maig 2016.
- [16] Adobe Phonegap Build: Documentació sobre l'eina de compilació de Phonegap, http://docs.build.phonegap.com/en_US/
- [17] Raul Martinez, "Método en cascada", http://www.academia.edu/6362716/METODO_EN_CASCADA, Febrer-Abril 2016.
- [18] Ophelia Pastrana, "Aplicar metodologías ágiles en el desarrollo del Software", <http://www.i2btech.com/blog-i2b/tech-deployment/5-beneficios-de-aplicar-metodologias-agiles-en-el-desarrollo-de-software/>, Febrer-Juny 2016.
- [19] Gestió del projecte de forma gratuïta, flexible i visual, <https://trello.com/>, Febrer-Juny 2016.
- [20] "Método de Kanban: gestión de proyectos y tareas para equipos, <http://iberzal.com/metodo-kanban/>, Juny 2016.

ANNEX: CAPTURES DE PANTALLA DE L'APLICACIÓ

En aquest annex es recullen diverses figures que representen les diferents pantalles que formen l'aplicació SocialSKI.

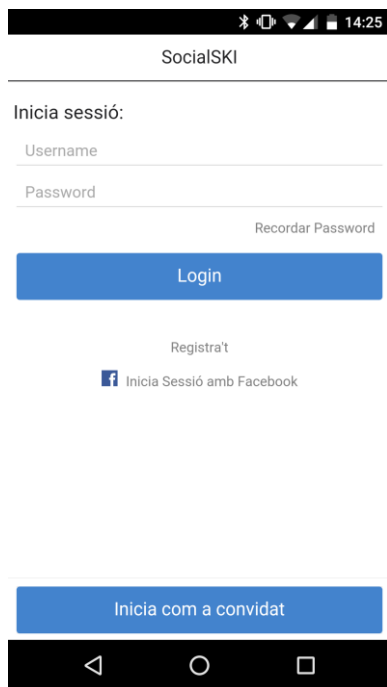


Figura 1: Login de l'aplicació

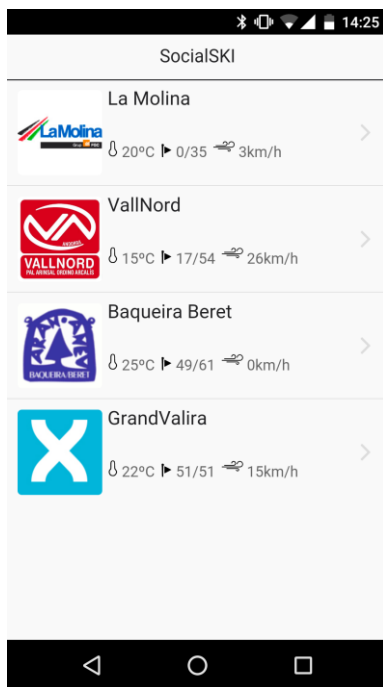


Figura 2: Llistat de les pistes d'esquí

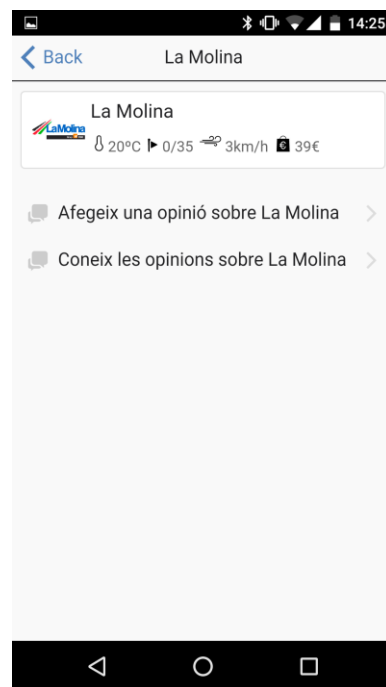


Figura 3: Detalls de la pista

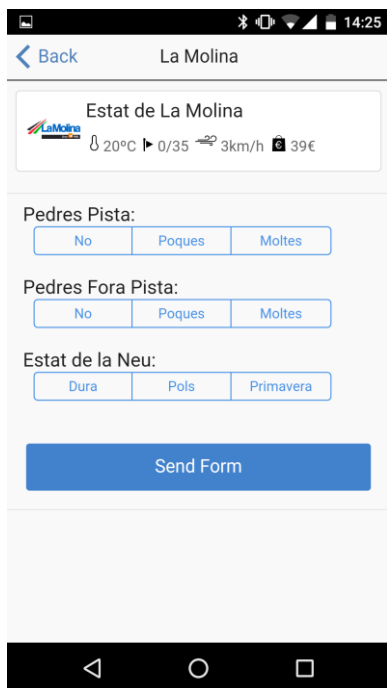


Figura 4: Formulari per aportar una opinió sobre la Molina

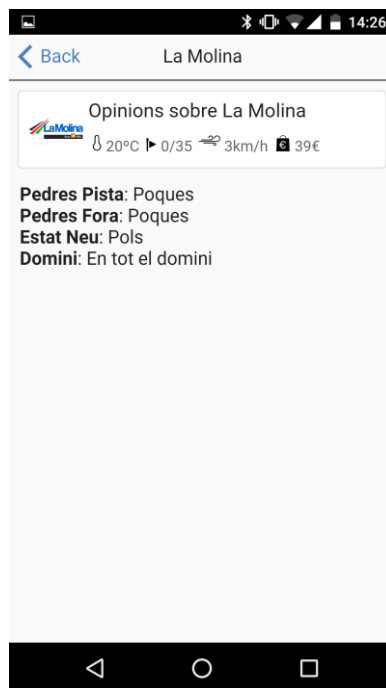


Figura 5: Mostra de la opinió resultant sobre la Molina