

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

APLICACIÓN HÍBRIDA PARA CONFERENCIAS

PABLO DAVID VILLARROEL PEÑA

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN
INFORMÁTICA

DICIEMBRE, 2018

Pontificia Universidad Católica de Valparaíso
Facultad de Ingeniería
Escuela de Ingeniería Informática

APLICACIÓN HÍBRIDA PARA CONFERENCIAS

PABLO DAVID VILLARROEL PEÑA

Profesor Guía: **Ismael Figueroa Palet**

Profesor Co-referente: **Leslie Pérez Cáceres**

Carrera: **Ingeniería de Ejecución en Informática**

Diciembre, 2018

A Dios por haberme acompañado y guiado por el camino correcto.

A mi Familia, quien me han apoyado incondicionalmente cada vez que lo he necesitado.

A todas las personas que me apoyaron y me aconsejaron.
Todo lo que he logrado ha sido gracias a ellos.

Índice

Resumen	iii
Abstract	iii
Lista de Figuras	iv
Lista de Tablas	v
1 Introducción	1
2 Marco general del proyecto	2
2.1 Situación en estudio – Descripción general	2
2.1.1 Caracterización de Connfa	3
2.2 Objetivos	3
2.2.1 Objetivo General	3
2.2.2 Objetivos Específicos	3
3 Situación de Estudio	4
3.1 Descripción de la situación actual	4
3.2 Identificación de los problemas detectados	4
3.3 Estado del arte.....	4
4 Propuesta de Solución	6
4.1 Descripción general de la solución	6
4.2 Modelo de proceso de desarrollo	6
4.2.1 Incrementos.....	6
4.3 Arquitectura de la solución	6
5 Desarrollo del software	8
5.1 Identificación de requerimientos	8
5.1.1 Historias de usuario primer incremento	8
5.1.2 Historias de usuario segundo incremento	8
5.1.3 Historias de usuario tercer incremento	9
5.2.1 Interfaz	10
5.2 API.....	13
5.3 Gestor	17
6 Implementación	19
6.1 Implementación primer incremento	19
6.2 Implementación segundo incremento	20

6.3	Implementación tercer incremento	21
7	Alcances y limitaciones.....	23
7.1	Alcances	23
7.2	Limitaciones	23
8	Conclusiones.....	24
9	Referencias	25

Resumen

El siguiente documento describe la problemática actual que desempeña un gestor de conferencia de código abierto llamada Connfa, que con el estado actual no permite realizar mejoras o actualización de manera práctica a las aplicaciones móviles porque se encuentran en distintos lenguajes de programación.

Para solucionar las problemáticas encontradas, se propone desarrollar una aplicación híbrida el cual puede ser distribuida en distintas plataformas y además que la información necesitada se solicite a través de una API.

Palabras-claves: connfa, gestor de conferencia, aplicaciones híbridas

Abstract

This document describes the current problems that an open-source conferencing manager named Connfa, which with the current state does not allow improvements or updates to mobile applications in a practical way because they are in different programming languages.

To solve the problems encountered, it is proposed to develop a hybrid application which can be distributed on different platforms and also that the information needed is requested through an API.

Keywords: connga, conference manger, hybrid application

Lista de Figuras

Figura 2. 1 Screenshot de la aplicación	2
Figura 4. 1 Arquitectura de la solución	7
Figura 5. 1 Menu principal	11
Figura 5. 2 Pantalla Sesiones.....	11
Figura 5. 3 Pantalla expositores.....	12
Figura 5. 4 Pantalla evento	12
Figura 5. 5 Pantalla orador	13
Figura 5. 6 Llamada API	13
Figura 5. 7 Gestor	18
Figura 6. 1 Vista Sesiones	19
Figura 6. 2 Filtro Tracks	20
Figura 6. 3 Vista Expositores	20
Figura 6. 4 Vista mapas	21
Figura 6. 5 Notificaciones	21
Figura 6. 6 Vista Notificaciones	22

Lista de Tablas

Tabla 5. 1 Tabla getSession	14
Tabla 5. 2 Tabla getSpeakers.....	14
Tabla 5. 3 Tabla getTypes	15
Tabla 5. 4 Tabla getLevels	15
Tabla 5. 5 Tabla getTracks	15
Tabla 5. 6 Tabla getInfo title	15
Tabla 5. 7 Tabla getInfo info	16
Tabla 5. 8 Tabla getFloorsPlans	16
Tabla 5. 9 Tabla getLocations	16
Tabla 5. 10 Tabla getSettings	16
Tabla 5. 11 Tabla checkUpdates	17

1 Introducción

Una de las problemáticas que existe a la hora de difundir el itinerario de alguna conferencia, es que la información a desplegar es demasiada al mostrarse en una página o un cartel, es por esto que se ve en la necesidad de ocupar las tecnologías de la información para que esta se distribuya de manera rápida y completa. Añadiendo a lo anterior, a veces ocurren cambios de último minuto que se quieren difundir de manera masiva pero no se puede hacer llegar a la gente que lo necesita. Es por esto que la necesidad de crear una aplicación móvil es fundamental cuando se gestiona alguna conferencia.

Particularmente, en la Pontificia Universidad Católica de Valparaíso (PUCV), se hacen conferencias y seminarios constantemente para exponer distintos temas de diferentes asuntos en todas las facultades. Por lo anterior, cierta facultad se ha acercado a la Escuela de Ingeniería Informática para solicitar una solución a su problemática. La solución más económica que se ha encontrado es consumir un gestor de conferencia de código abierto (Connfa) el cual no es probable ya que contiene muchos problemas. En el pasado estudiantes de la Escuela han solucionado algunos de estos, pero sigue habiéndolos.

Según Lemberg, la empresa detrás de Connfa, las aplicaciones para cada sistema operativo móvil (Android e iOS) están subidas para el uso libre, el problema es que al momento de mejorar o evolucionar las aplicaciones se deben realizar los cambios para las dos plataformas, teniendo que consumir más recursos en el intento. Es por esto que se tomó la decisión de atacar esta problemática utilizando las aplicaciones híbridas, lo cual consiste en crear una sola base para ejecutar en multiplataforma, siendo esta alimentada a través de una API. La aplicación híbrida será desarrollada con una metodología incremental la cual es un método ágil para el desarrollo de software y sus requerimientos fueron obtenidos mediante la técnica de historias de usuarios, además se realizaron mockups para visualizar como sería la interfaz de la aplicación móvil.

2 Marco general del proyecto

A continuación, se describe la situación general del proyecto, los objetivos principales y los específicos

2.1 Situación en estudio – Descripción general

La Pontificia Universidad Católica de Valparaíso en Chile (PUCV), participa organizando múltiples conferencias al año, es por esto que es recurrente que se acerquen a la Escuela de Informática solicitando realizar aplicaciones para dispositivos móviles con el objetivo de desplegar la información de forma ordenada e interactiva. Esto ya ha ocurrido para la conferencia FELAFACS 2017, organizada por la Escuela de Periodismo [1].

Para satisfacer dicha solicitud se trabajó con la plataforma Connfa [2], una aplicación de código abierto creada por la empresa desarrolladora de software, Lemberg, para gestionar los distintos ámbitos que abarcan las conferencias, tales como invitados, horarios, tipos de charlas y sesiones. En un proyecto de título de la Escuela de Ingeniería en Informática de la Pontificia Universidad Católica de Valparaíso, realizado por Omar Vera, se desarrolló el Backend Administrativo, implementando las API utilizadas por esta aplicación.

Actualmente Connfa cuenta con dos aplicaciones móviles, una para el sistema operativo Android y otra para iOS. Sin embargo, aunque las aplicaciones se comunican de la misma manera, sus implementaciones son diferentes, la primera está implementada en Objective-C y la segunda en Java, y tienen distintos problemas y bugs, lo que hace que sea complicado de mantener y evolucionar ambas a la vez. A modo de solución, este proyecto consiste en desarrollar una aplicación híbrida utilizando el framework Ionic, y así mantener una sola base de código para los dos sistemas operativos, u otras plataformas, además siendo compatible exclusivamente con el backend administrativo ya mencionado.

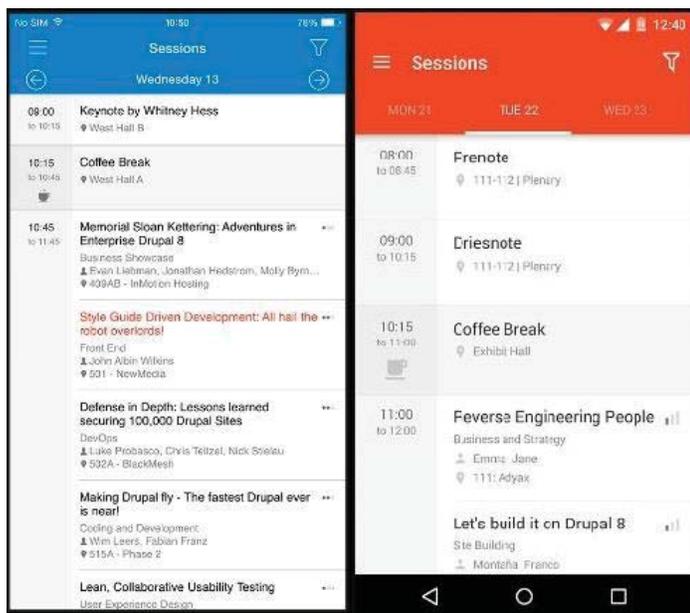


Figura 2. 1 Screenshot de la aplicación

2.1.1 Caracterización de Connfa

1. **Sesión (Session):** Una sesión es la presentación sobre un tema en específico dictada por uno o varios oradores.
2. **BOF (Birds of a Feather):** Es una expresión en inglés para referirse a una discusión en grupo de carácter informal donde los asistentes se agrupan en base a un interés compartido y llevan a cabo discusiones.
3. **Evento (Social Event):** Un evento refiere a un conjunto de personas específicas que generalmente realizan actividades grupales.
4. **Orador (Speaker):** Un orador es aquella persona que realiza una charla durante una sesión, BOF o un Evento. Podría haber varios oradores para una sola sesión. Se compone de varios atributos tales como: nombre, apellido, avatar, nombre de organización, título profesional, descripción, Twitter y sitio web.
5. **Categoría (Track):** Es una clasificación para identificar sesiones, BOF y eventos sociales. Algunos ejemplos de estos son: “Gestión y Administración”, “Gestión del Riesgo”, “Medicina Natural” o “Programación”.
6. **Nivel de Experiencia (Experience Level):** Es una clasificación que sirve para aconsejar el nivel requerido para participar adecuadamente de una sesión particular, BOF o evento. Algunos ejemplos de estos son: Principiante, Medio y Avanzado.
7. **Lugares (Location):** Sirve para identificar el lugar donde se realiza la conferencia.
8. **Información (Information):** Sirve para conocer los detalles del evento tales como: Acerca del evento, Acerca de la aplicación, detalles del contacto, etc.

2.2 Objetivos

A continuación, se presentan el objetivo general y los objetivos específicos.

2.2.1 Objetivo General

- Desarrollar una aplicación híbrida para los sistemas operativos Android e iOS, en función de facilitar la gestión de conferencias, basándose exclusivamente en el proyecto de título Backend Administrativo para gestor de conferencias Connfa.

2.2.2 Objetivos Específicos

- Investigar la lógica de la gestión de conferencias, identificando sus categorías para el desarrollo de la aplicación híbrida.
- Diseñar la aplicación híbrida a partir exclusivamente del proyecto de título Backend Administrativo para gestor de conferencias Connfa.
- Crear e implementar la aplicación híbrida para la gestión de conferencias, por medio del framework Ionic, verificando y validando su funcionamiento en los sistemas operativos Android e iOS.

3 Situación de Estudio

En la siguiente sección se presenta una pequeña descripción de la situación actual en cuanto al uso de Connfa, los problemas que conlleva el uso de éste actualmente y el estado del arte en cuanto al uso de aplicaciones para conferencias a nivel de nacional.

3.1 Descripción de la situación actual

Connfa nació a manos de la empresa Lemberg, una empresa de software fundada por un grupo de jóvenes ucranianos en el año 2007, quienes realizaron cuatro aplicaciones distintas para las conferencias oficiales de Drupal. DrupalCon es un evento educativo internacional que reúne a las personas que utilizan, desarrollan, diseñan y respaldan la plataforma Drupal. Este evento presenta docenas de sesiones y paneles de algunas de las personas más influyentes de la comunidad de Drupal. Tras lanzar la aplicación móvil para Android la comunidad de Drupal vio pertinente que la aplicación fuese también para iOS, por lo anterior en la conferencia de Ámsterdam en septiembre de 2014 se lanzó al público para ambas plataformas, con un total de dos mil descargas.

3.2 Identificación de los problemas detectados

Actualmente existen dos aplicaciones para los dos sistemas operativos móviles principales del mercado que son Android e iOS, cada uno de ellas presentan problemas con sus propios bugs. Esto genera una divergencia, duplicando el esfuerzo y los recursos en caso de querer hacer mejoras a la aplicación o responder a nuevos requerimientos. Limitar las opciones a dos sistemas operativos es generar problemas al momento satisfacer la necesidad de difundir la información de la conferencia ya que los usuarios, potencialmente, podrían estar utilizando uno distinto.

3.3 Estado del arte

En el siguiente apartado se menciona la historia de Connfa y se mencionara de otros gestores de conferencia que cuentan con alguna aplicación para dispositivos móviles.

La aplicación de código abierto Connfa, la desarrolló una empresa llamada Lemberg [3], esta empresa se fundó el año 2007 en el país de Ucrania y se caracteriza por ser una consultora de software y hardware. La aplicación nació por la necesidad de una conferencia que se realiza en distintos países llamada DrupalCon. Este congreso une expertos de todo el mundo para mostrar las tendencias en la experiencia digital que se realizan en docenas de sesiones y paneles de las personas más influyentes de la comunidad de Drupal [4].

AdianteApps es una plataforma web para crear aplicaciones Android de manera sencilla y sin saber programar, esta se utiliza frecuentemente para conferencias, eventos, ferias, seminarios y congresos. Los problemas de esta plataforma es que solo las aplicaciones se pueden utilizar en Android y iOS, limitando su expansión a otros sistemas y es de pago mensual [5].

EvenTwo es una plataforma web que desarrolla aplicaciones para conferencias muy completas teniendo una sede en Chile con grandes clientes en todo el mundo, entre sus principales funciones esta la participación de los asistentes mediante votaciones instantáneas,

notificaciones push, gamificación y análisis de datos. El problema de la plataforma es que solo se puede utilizar para Android e iOS y los precios son elevados al querer tener más funcionalidades para tu evento [6].

Attendify es una plataforma completa que se puede utilizar para gestionar la conferencia de manera gratuita siempre y cuando el evento sea gratuito. Los problemas de esto es que la aplicación es en general, es decir, que se debe buscar el evento dentro de la aplicación e incorporarse a esto. Esto se puede considerar un problema ya que con tanta información puede confundir a los usuarios de la inmensa cantidad de eventos que existe [7].

4 Propuesta de Solución

A continuación, se presenta una descripción de la solución al problema, junto al modelo de desarrollo seleccionado con su justificación, además de la arquitectura y de las herramientas que se utilizará para llevar a cabo el desarrollo del sistema.

4.1 Descripción general de la solución

Se desarrollará una aplicación híbrida a través de una interfaz simple donde se mostrará la información de las sesiones, oradores, categorías, etc. El usuario podrá informarse descargando una aplicación tanto para los sistemas operativos móviles Android e iOS y visualizarlo a través de la web. Además, podrá guardar las sesiones, eventos sociales y BOF que le parezcan más llamativas para revisarlas en una sola categoría. Inicialmente se buscará replicar la interfaz de usuario de las aplicaciones originales, pero, eventualmente se podrá evolucionar a un nuevo diseño mediante una sola base de código fuente.

4.2 Modelo de proceso de desarrollo

Se contempló que para el diseño del sistema se utilizará la metodología de desarrollo incremental [8], ya que este sostiene un desarrollo modular el cual consiste en suministrar partes del software funcional llamados “incrementos” que son seleccionados por prioridad siendo los primeros lo cruciales. Cabe destacar que esta metodología permite ampliar y mejorar funcionalidades en cada incremento, refinando el producto de manera ágil, es decir, cada incremento agrega una nueva funcionalidad o cubren nuevos requisitos, además que es posible realizar entregas de un sistema.

4.2.1 Incrementos

Durante el desarrollo del sistema se presentarán al menos 3 incrementos los cuales son fundamentales para reflejar la información en la aplicación. A continuación, se presentan los incrementos.

1. **Información de las sesiones:** Incremento que cubre todo lo relacionado con sesiones, BOF y eventos sociales. Tiene la particularidad que los tres comparten las mismas propiedades y tipos de datos.
2. **Información del evento:** Incremento que considera la información sobre el evento tales como los oradores, lugares e imágenes.
3. **Mis Eventos:** Incremento que agrega una funcionalidad a la aplicación donde se puede guardar todas las sesiones, BOF y eventos sociales en una sola categoría para mostrar la información relevante que el usuario estime conveniente en un solo lugar.

4.3 Arquitectura de la solución

La solución propuesta comprende que el software será desarrollado con el patrón de arquitectura Modelo-Vista-Controlador (MVC) un patrón propuesto por Reenskaug [9], para la solución de aquellos problemas donde se deben tratar con grandes volúmenes de datos, que consiste en tres principales capas las cuales se describen a continuación. Primero está la capa llamada Modelo donde se trabaja con los datos, la Vista es la interfaz con la que el

usuario interactúa, esta nos permite mostrar la salida y el Controlador que se encarga de ser enlace entre las vistas y los modelos. Por lo tanto, la aplicación híbrida sería la capa de la vista quien realizaría llamadas mediante un servicio API, que este es el encargado de proveer de información a la aplicación y además es el encargado de manipular las tuplas en la base de datos, en este caso el servicio API sería la capa del controlador y la base de datos la capa modelo.

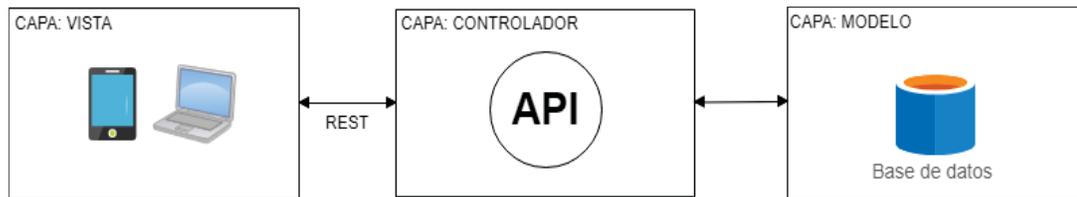


Figura 4 1 Arquitectura de la solución

4.4 Herramientas de desarrollo

Dado que se utilizará el patrón de arquitectura MVC o modelo vista controlador, se utilizará el framework para el desarrollo de aplicaciones híbridas Ionic [10], que se enfoca en dicho patrón. Ionic es de uso gratuito y de código abierto para desarrollar aplicaciones híbridas multiplataforma que utiliza HTML5, CSS (generado por SASS) y Cordova como base. Es uno de los framework del momento por utilizar AngularJS para gestionar las aplicaciones, lo que asegura aplicaciones rápidas y escalables. Para la administración de los archivos se utilizará SourceTree (<https://www.sourcetreeapp.com>) para interactuar con los repositorios de Git. Para el diseño de mockups se utilizará Moqups en su versión web (<https://www.moqups.com>).

5 Desarrollo del software

En esta sección se presenta las partes de cómo se realizará el desarrollo de la aplicación, mostrando los requerimientos capturados mediante la técnica de Historias de Usuario, así como los mockups y los métodos de la API a implementar.

5.1 Identificación de requerimientos

A continuación, se presentan los requerimientos mediante la técnica de Historias de usuario. Esta técnica es utilizada en las metodologías de desarrollo ágil, el cual tiene como objetivo describir las funcionalidades del sistema en un lenguaje cotidiano para facilitar su comprensión.

5.1.1 Historias de usuario primer incremento

HU1: Como usuario, quiero consultar las sesiones de la conferencia para encontrar los detalles de la información de cada sesión.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. Si presiono el icono “Menu” se desplegará las opciones desde el lado izquierdo.
- C. Si presiono el botón “Sesiones”, se muestra una pantalla con una o distintas pestañas en la parte superior con las fechas en que se realizará las sesiones, debajo de las pestañas se encuentran las sesiones agrupadas por hora de inicio.
- D. Si presiono una sesión se despliega una nueva pantalla con el título, descripción, hora de inicio, hora de termino, ubicación, track y expositores si existen asociados.
- E. Si presiono el icono “Atrás” volverá a la pantalla anterior.

HU2: Como usuario, quiero filtrar los tracks de la conferencia para los visualizar solo aquellas sesiones que me interesan.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. Si presiono el icono “Menu” se desplegará las opciones desde el lado izquierdo.
- C. Si presiono el botón “Sesiones”, se muestra una pantalla con una o distintas pestañas en la parte superior con las fechas en que se realizará las sesiones, debajo de las pestañas se encuentran las sesiones agrupadas por hora de inicio.
- D. En la parte superior existe un botón con un icono que al presionar aparecerá un listado en forma de checkbox con los todos tracks que existen en la conferencia.
- E. Al seleccionar los de mi interés y presionar el botón filtrar deberá refrescar la página y mostrarme sola aquellas sesiones que cumplan con la solicitud de filtro.

5.1.2 Historias de usuario segundo incremento

HU3: Como usuario, quiero consultar los expositores de la conferencia para encontrar los detalles de la información de cada expositor.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. Si presiono el botón “Menu” se desplegará las opciones desde el lado izquierdo.
- C. Si presiono el botón “Expositores” aparecerá una lista ordenada alfabéticamente y agrupadas por la primera letra del nombre.
- D. Si presiono a un expositor se despliega una nueva pantalla con el nombre completo, trabajo y organización, twitter, correo personal, sitio web, descripción y las sesiones relacionadas si existen asociados.
- E. Si presiono el icono “Atrás” volverá a la pantalla anterior.

HU4: Como usuario, si es que está disponible, quiero consultar las ubicaciones de la conferencia para visualizarlas en un mapa.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. Si presiono el botón “Menu” se desplegará las opciones desde el lado izquierdo.
- C. Si presiono el botón “Mapas” aparecerá el primer mapa que se ingresó con un título y la dirección.
- D. Si existe más de un mapa aparecerá una lista desplegable con los títulos de los distintos mapas.
- E. Al seleccionar un título diferente aparecerá un mapa relacionado a este.

5.1.3 Historias de usuario tercer incremento

HU5: Como usuario quiero guardar las sesiones de la conferencia para visualizarlas en una sección diferente.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. Si presiono el botón “Menu” se desplegará las opciones desde el lado izquierdo.
- C. Si presiono el botón “Sesiones”, se muestra una pantalla con una o distintas pestañas en la parte superior con las fechas en que se realizará las sesiones, debajo de las pestañas se encuentran las sesiones agrupadas por hora de inicio.
- D. Si presiono una sesión se despliega una nueva pantalla con la información asociada y un botón con un icono de estrella.
- E. Si presiono el botón con el icono de estrella deberá cambiar de color y este cambio deberá ser permanente.

HU6: Como usuario quiero consultar las sesiones de la conferencia que haya guardado para visualizarlas en una sección diferente.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. Si presiono el botón “Menu” se desplegará las opciones desde el lado izquierdo.
- C. Si presiono el botón “Mi Calendario”, se muestra una pantalla con una o distintas pestañas en la parte superior con las fechas en que se realizará las sesiones guardadas, debajo de las pestañas se encuentran las sesiones agrupadas por hora de inicio, en caso de ser eliminado después de haber sido guardado se mostrara un mensaje advirtiendo esto.
- D. Si presiono una sesión se despliega una nueva pantalla con la información asociada.

HU7: Como usuario quiero ser advertido si existe cambios en las sesiones de la conferencia para enterarme de las modificaciones si es que se han hecho.

Criterios de aceptación:

- A. Ingreso a la aplicación.
- B. En las pantallas principales deberá existir un icono de campana.
- C. Al haber una notificación deberá advertirme colocando algún distintivo en el icono de la campana.
- D. Si presiono el icono de la campana deberá desplegar una pantalla mostrando las modificaciones de las sesiones tales como el nombre, la descripción, la fecha, la ubicación y si es que se elimina.

5.2 Diseño

En la siguiente sección se presenta el modelo mostrando los principales mockups del sistema, creados con la herramienta online moqups, con los cuales se define de una forma aproximada la interfaz con la que el usuario estará interactuando al utilizar el sistema, además se presenta con los métodos a manejar de la API utilizada.

5.2.1 Interfaz

A continuación, se presenta los mockups de la aplicación a desarrollar.

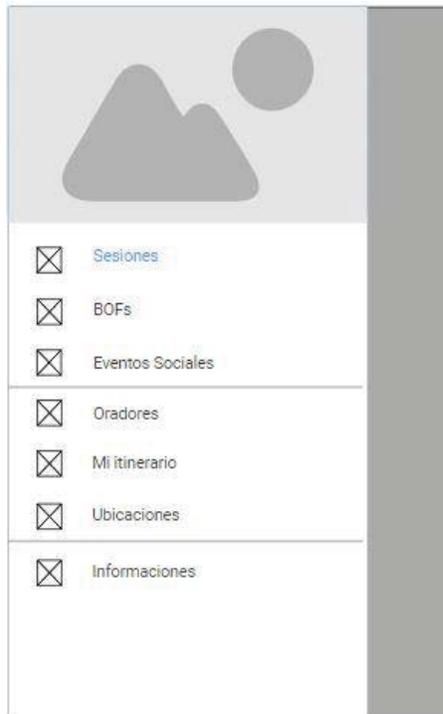


Figura 5. 1 Menu principal



Figura 5. 2 Pantalla Sesiones

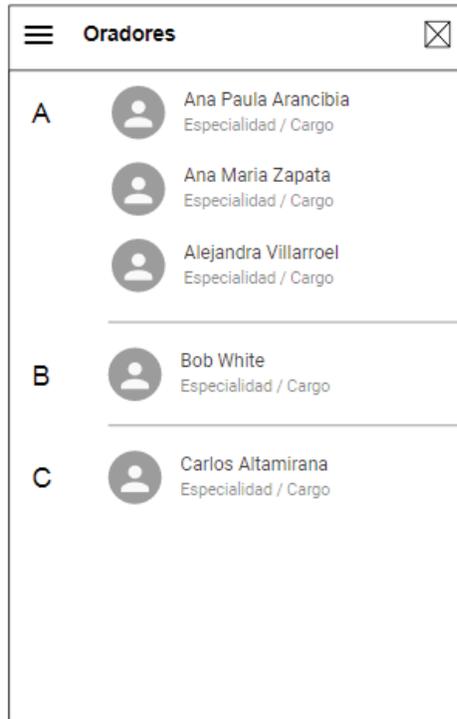


Figura 5. 3 Pantalla expositores

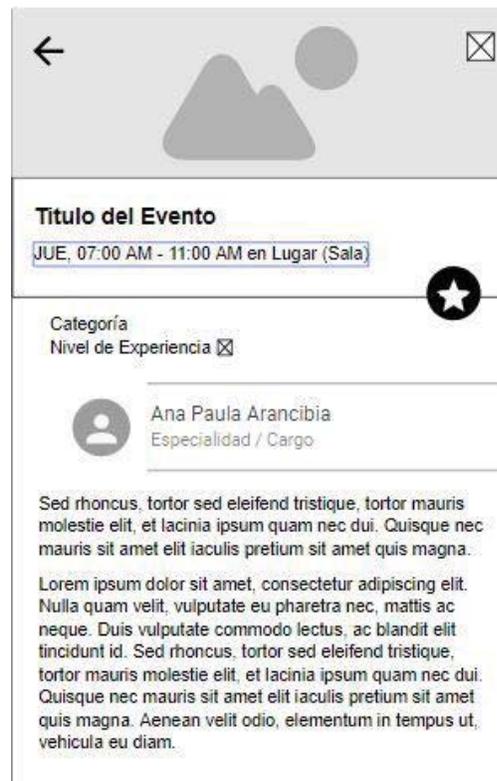


Figura 5. 4 Pantalla evento



Figura 5. 5 Pantalla orador

5.2 API

Para llamar el servicio API se debe hacer de la manera en como se muestra en la imagen a continuación (Figura 5.6).

BASE_URL + /api/ + <id_conferencia> + / + <método>

Figura 5. 6 Llamada API

El cual “BASE_URL” es la raíz del sitio web en donde se encuentra instalada la API, id_conferencia es el identificador de la conferencia y método es la palabra clave para solicitar la información deseada. A continuación, se describen los métodos que consumirá la aplicación híbrida con una breve descripción y una tabla con la propiedad, el tipo de dato y una breve descripción.

getSessions: Este método retorna una lista de todas las sesiones de una conferencia con la siguiente estructura: Un arreglo con un identificador “days” el cual contiene dos identificadores. El primero es “date” el que tiene formato DD-MM-YYYY. El segundo es un arreglo llamado “events” el cual contiene las distintas sesiones con el siguiente formato:

Tabla 5. 1 Tabla getSession

Propiedad	Tipo	Descripción
eventId	Integer	Identificador de la sesión, BOF o evento
From	String	Fecha y Hora de Inicio en formato ISO8601
To	String	Fecha y Hora de Terminación formato ISO8601
Text	String	Descripción del evento puede estar en formato text/html
Name	String	Nombre de la sesión, BOF o evento
Place	String	El lugar donde se realizará la sesión, BOF o evento
experienceLevel	integer	ID que identifica el nivel de la sesión, BOF o evento
Type	integer	ID que identifica a el tipo de la sesión, BOF o evento
Track	integer	ID que identifica categoría/temática de la sesión, BOF o evento
speakers	Arreglo	Arreglo con los IDs de los Oradores que dictaran en la sesión, BOF o evento
Link	String	URL de la sesión, BOF o evento
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación
Order	Real	Usado para ordenarlo

getBofs: Este método retorna una lista de todas las BOFs, tiene la misma estructura que el método getSession.

getSocialEvents: Este método retorna una lista de todos los eventos sociales, tiene la misma estructura que el método getSession.

getSpeakers: Este método retorna un arreglo con todos los oradores asociados a una conferencia y contiene el siguiente formato:

Tabla 5. 2 Tabla getSpeakers

Propiedad	Tipo	Descripción
speakerId	integer	Identificador del orador
firstName	string	Nombre del orador
lastName	string	Apellido del orador
avatarImageURL	string	Url de una imagen para el orador
organizationName	string	Nombre de la organización del orador
jobTitle	string	Cargo del orador
characteristic	string	Descripción del orador o reseña de su trayectoria
twitterName	string	Cuenta de twitter del orador
webSite	string	Url del sitio web del orador
order	real	Usado para ordenarlo
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getTypes: Este método retorna una lista de todos los tipos de una conferencia con la siguiente estructura: Un arreglo con un identificador “types” el cual contiene el siguiente formato:

Tabla 5. 3 Tabla getTypes

Propiedad	Tipo	Descripción
typeID	integer	Identificador para el tipo
typeName	string	Nombre de tipo
Order	real	Usado para ordenarlo
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getLevels: Este método retorna una lista de todos los niveles de una conferencia con la siguiente estructura: Un arreglo con un identificador “levels” el cual contiene el siguiente formato:

Tabla 5. 4 Tabla getLevels

Propiedad	Tipo	Descripción
levelID	integer	ID único para el nivel
levelName	string	Nombre de nivel
order	real	Usado para ordenarlo
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getTracks: Este método retorna una lista de todas las categorías de una conferencia con la siguiente estructura: Un arreglo con un identificador “tracks” el cual contiene el siguiente formato:

Tabla 5. 5 Tabla getTracks

Propiedad	Tipo	Descripción
trackID	integer	Identificador para la categoría
trackName	string	Nombre de categoría
Order	real	Usado para ordenarlo
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getInfo: Este método retorna dos listas sobre información de una conferencia con la siguiente estructura: El primer arreglo tiene un identificador “title” el cual contiene distintos tipos de conferencia con el siguiente formato:

Tabla 5. 6 Tabla getInfo title

Propiedad	Tipo	Descripción
titleMajor	string	Título de la conferencia
titleMinor	string	Subtitulo de la conferencia

El segundo arreglo tiene un identificador “info” el cual contiene la siguiente estructura:

Tabla 5. 7 Tabla getInfo info

Propiedad	Tipo	Descripción
infoId	integer	ID único para la información
infoTitle	string	Título para la información
Html	string	Descripción de la información
Order	real	Usado para ordenarlo
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getFloorPlans: Este método retorna una lista de todos los planos de una conferencia con la siguiente estructura: Un arreglo con un identificador “floorPlans” el cual contiene el siguiente formato:

Tabla 5. 8 Tabla getFloorsPlans

Propiedad	Tipo	Descripción
floorPlanId	integer	ID único para los planos
floorPlanName	string	Título para los planos
floorPlanImageURL	string	URL de una imagen de los planos
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getLocations: Este método retorna una lista de todas las direcciones de una conferencia con la siguiente estructura: Un arreglo con un identificador “locations” el cual contiene el siguiente formato:

Tabla 5. 9 Tabla getLocations

Propiedad	Tipo	Descripción
locationId	integer	Identificador para la dirección
longitude	real	Número de longitud de la dirección
latitude	real	Número de latitud de la dirección
locationName	string	Nombre de la dirección (como un título)
address	string	Dirección real
Order	real	Usado para ordenarlo
deleted	booleano	Booleano que si su estado es verdadero no se mostrará en la aplicación

getSettings: Este método retorna una lista con los títulos, el hashtag y la zona horaria de una conferencia con la siguiente estructura: Un arreglo con un identificador “settings” el cual contiene el siguiente formato:

Tabla 5. 10 Tabla getSettings

Propiedad	Tipo	Descripción
titleMajor	string	Título de la conferencia
titleMinor	string	Subtitulo de la conferencia
twitterWidget	string	Hashtag que se utilizara para la conferencia
timezone	string	Zona horaria en donde se realiza la conferencia

checkUpdates: Este método retorna una lista con un identificador “idsForUpdate” de identificadores únicos que están asociados a aquellos métodos que se pueden actualizar. En la siguiente tabla se muestra los métodos con sus respectivo identificadores.

Tabla 5. 11 Tabla checkUpdates

Método API	ID
getSettings	0
getTypes	1
getLevels	2
getTracks	3
getSpeakers	4
getLocations	5
getFloorPlans	6
getSessions	7
getBofs	8
getSocialEvents	9
getInfo	11

5.3 Gestor

A continuación, se describirá a grandes rasgos el funcionamiento del gestor realizado por Omar Vera.

El gestor cuenta con un menú con distintas opciones para editar los variados aspectos de la conferencia tales como eventos, conferencistas, locaciones, planos de pisos, páginas y conferencias (Figura 5.6). En la sección de eventos, se puede crear, editar o eliminar los niveles, tracks, tipos, sesiones, sociales y bofs de la conferencia. En la sección conferencista se puede crear, editar o eliminar expositores. En la sección locaciones, se puede crear, editar o eliminar las direcciones que cuentan la conferencia colocando la latitud, longitud y su dirección. En la sección planos de piso, se puede crear, editar o eliminar imágenes de las conferencias que generalmente son planos. En la sección páginas se puede agregar, editar o eliminar la información adicional que tiene la conferencias.

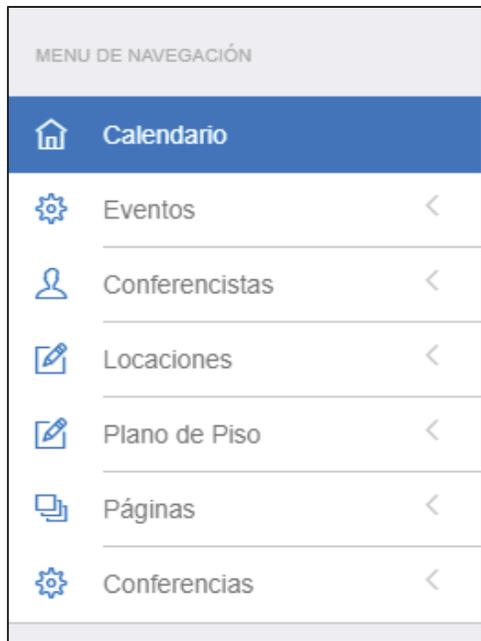


Figura 5. 7 Gestor

6 Implementación

Como se mencionó anteriormente el desarrollo del sistema se separó en tres incrementos los cuales se verán a continuación de forma más detallada.

6.1 Implementación primer incremento

El primer incremento consistió en desarrollar todas las funcionalidades relacionadas con el despliegue de las sesiones y sus detalles



Figura 6. 1 Vista Sesiones

Tal como se puede apreciar en la Figura 6.1 en la parte superior existen las pestañas con las fechas correspondiente a la conferencia y una lista con las sesiones que se realizaran en ese día. Están agrupadas por la hora de inicio de las sesiones, cuenta con un título la hora de inicio y termino, la ubicación, los expositores y el track si existiera.

Como se puede apreciar en la Figura 6.1 en la parte superior existe un icono en forma de embudo (funnel) el cual al presionarlo aparecerá una ventana con los tracks que están asociadas a la conferencia.

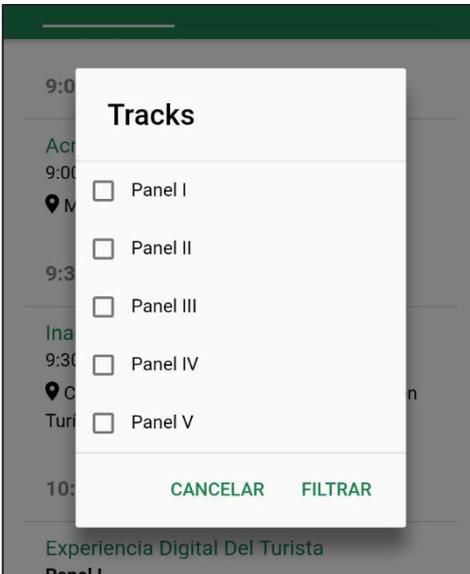


Figura 6. 2 Filtro Tracks

En la Figura 6.2 se aprecia la ventana con los tracks asociados a la conferencia y una casilla a su lado izquierdo que al presionarlo se marcará con un ticket, al presionar el botón “FILTRAR” regresará a la pantalla anterior (Figura 6.1) y solo mostrará aquellas sesiones que contenga los tracks seleccionados.

6.2 Implementación segundo incremento

El segundo incremento consistió en desarrollar todas las funcionalidades relacionadas con los expositores, sus detalles y los mapas.

Los expositores están agrupados por la letra inicial de su nombre, cada expositor aparecerá con una foto asociada, su nombre, apellido, cargo y nombre de la organización que representa.

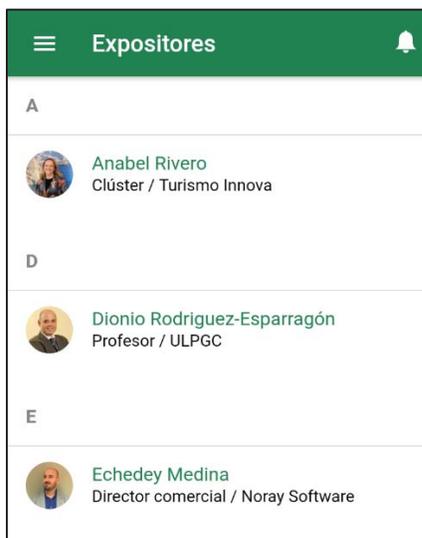


Figura 6. 3 Vista Expositores

En la figura 6.4 se puede apreciar la vista de los mapas el cual contiene un título, la dirección y un mapa y en el caso de que hubiera más locaciones, existe un listado despegable con los títulos de las diferentes ubicaciones.

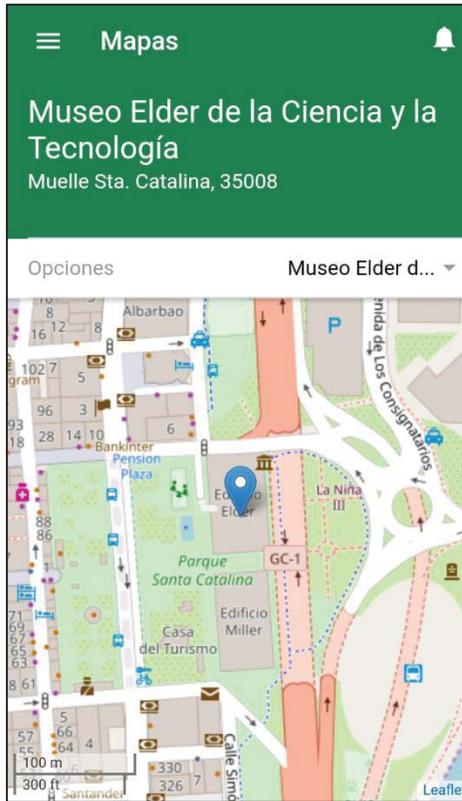


Figura 6. 4 Vista mapas

6.3 Implementación tercer incremento

El tercer incremento consistió en desarrollar todas las funcionalidades relacionadas con que el usuario pueda guardar las sesiones que el mismo seleccione como favoritas y notificaciones en caso de que cambie la descripción, el título, la fecha/hora de inicio y termino de las sesiones.

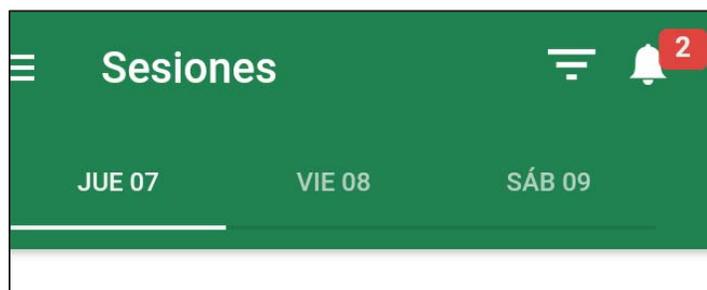


Figura 6. 5 Notificaciones

Como se puede apreciar en la Figura 6.1, Figura 6.3, Figura 6.4 existe en la parte superior derecha un icono en forma de campana el cual se agrega una insignia de color rojo con el número de modificaciones de las sesiones tal como aparece en la Figura 6.5. Al tocar el icono de la campana se desplegará una ventana nueva el cual aparecen las notificaciones con el nombre de la sesión, track, fecha/hora de inicio y termino, y además incluyendo si la sesión se creó, si se modificó la descripción, fecha/hora de inicio y/o termino y el nombre de la sesión.



Figura 6. 6 Vista Notificaciones

7 Alcances y limitaciones

En ésta sección se documentan los alcances del proyecto, las limitaciones de éste y las propuestas de trabajo a futuro sobre la aplicación desarrollada.

7.1 Alcances

Este proyecto tiene como alcance realizar una aplicación híbrida para que en el caso de nuevos requerimientos o arreglar fallos no detectados sea más sencillo realizar estos ya que se maneja un solo código para ser ejecutando en multiplataforma y no como está actualmente en dos distintos lenguajes de programación.

7.2 Limitaciones

Este proyecto se encuentra limitado en cuanto a nuevos requerimientos que se hubieran querido agregar, ya que, si la solución que se tendría con un nuevo requerimiento no solo se pueda cumplir modificando la interfaz, se tendría que modificar el gestor y/o la comunicación (API) que se tiene con la aplicación.

7.3 Trabajo a futuro

Como trabajo a futuro, debido a que funcionalmente el sistema está completo, se propone:

- Migrar el gestor desde Angular 1.5.9 a un framework más actual.
- Agregar nuevas funcionalidades a la interfaz con nuevos requerimientos que se han descubierto o se podrían eventualmente encontrar, tales como los mapas se abran con la aplicación favorita de ubicación del usuario, mensajes masivos, alerta de que la sesión esta por empezar, entre otros.
- Probar la aplicación en una conferencia real.

8 Conclusiones

Para este informe sobre el proyecto, se concluye que primordialmente se ha logrado cumplir gran parte del diseño de la propuesta de la solución. Aun así, se cree que siempre se puede mejorar, incluyendo los nuevos requerimientos que surgieron cuando se realizaba la aplicación híbrida.

En esta parte final del proyecto, se logró exitosamente realizar la aplicación, pero no se pudo implementar la aplicación en una situación real como en algún seminario o conferencia para evaluar el desempeño. Se espera como trabajo a futuro mejorar la aplicación, implementar nuevos requerimientos, probarlo en una situación real y tener una retroalimentación de los usuarios para encontrar nuevos problemas y proponer soluciones para seguir construyendo una aplicación más robusta. El proyecto queda abierto a expandirse en el caso de que se necesite mejorar alguno de los otros sistemas como es la API o el Gestor, poniendo en énfasis en este último ya que se encontraron ciertos problemas de usabilidad al tratar de ingresar información.

Por último, dejar establecido en esta misma sección que el proyecto se seguirá mejorando y testeándola para suplir las necesidades de la mayoría de conferencias y hacer un producto completo para quienes lo necesiten.

9 Referencias

- [1] Sitio web de la programación general de FELEFACS XVII de la PUCV, disponible en: <http://www.pucv.cl/uuaa/programacion-academica/2016-04-01/153143.html>
- [2] Sitio web oficial de la aplicación connfa, disponible en: www.connfa.com
- [3] Sitio web oficial de Lemberg, disponible en: <https://lembergsolutions.com/about>
- [4] Sitio web oficial de DrupalCon, disponible en: <https://events.drupal.org/>
- [5] Sitio web oficial de AdianteApp, disponible en: <https://www.adianteapps.com/info/hacer-aplicaciones-moviles-para-eventos>
- [6] Sitio web oficial de AdianteApp, disponible en: <https://eventwo.com/>
- [7] Sitio web oficial de Attendify, disponible en: <https://attendify.com/>
- [8] Pressman, R. (2010). Ingeniería del software un enfoque práctico (7ª ed.). Disponible en <http://cotana.informatica.edu.bo/downloads/ldIngenieria.de.software.enfoque.practico.7ed.Pressman.PDF>
- [9] Reenskaug, T. (2007). The original MVC reports. Disponible en http://heim.ifi.uio.no/~trygver/2007/MVC_Originals.pdf
- [10] Sitio web oficial del framework Ionic, disponible en: <https://ionicframework.com/docs/>