



Desarrollo de Proyectos en Arquitecturas Web 2.0  
Profesor: Mirko Gueregat Henríquez/Enrique Riquelme Bracho  
Ingeniería Civil Informática.  
Universidad Austral de Chile

# **Informe 02:**

# **Documento de Diseño:**

# **Fraternidad Wash**

## **Integrantes:**

José Luis Acuña Oyarce.

Francisco Bahamondes.

Eduardo Hopperdietzel.

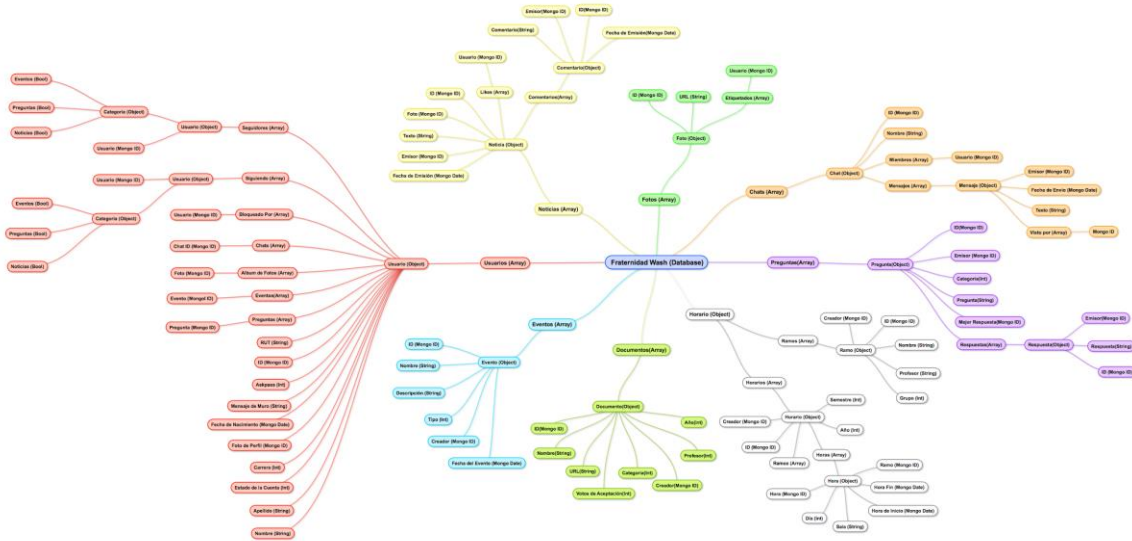
Nicolás Jiménez.

## **Fecha de Entrega:**

18 de octubre de 2016.

### 1) Diagrama de Datos:

Utilizaremos MongoDB para almacenar los datos. Trataremos a cada elemento (fotos, usuarios, noticias, etc) como un objeto, y dividiremos toda la base de datos en arreglos de estos mismos, como se muestra en la imagen:



Lo cual nos permite tener un mejor orden y manejo de permisos de los usuarios. Por ejemplo, si un usuario quiere eliminar una de sus fotos, simplemente elimina el ID de la foto dentro de su propio arreglo de fotos, pero no edita el arreglo de fotos global.

## 2) Lista de funcionalidades a desarrollar:

1. **Registro de usuarios:** Los usuarios al momento de registrarse, deberán entregar la siguiente información: Nombre, apellido, fecha de nacimiento, sexo, carrera, año de ingreso, contraseña y correo institucional, este último nos permitirá verificar que la persona sea miembro de la universidad, verificando que el dominio de su correo sea (@alumnos.uach.cl) y a la vez enviando un correo de comprobación para descartar el falsificamiento de usuario.
2. **Perfil de usuario:** Permitirá a los usuarios manejar su perfil (o visitar el de otros), agregar o cambiar datos personales tales como su foto de perfil, nombre, mensajes de muro, entre otros.
3. **Login de entrada:** Trataría de implementar un proceso limpio de ingreso de usuarios registrados con el correo institucional particular que posea cada uno de los estudiantes.
4. **Uso de Fraterno:** Esta es la nueva funcionalidad que tendrá la red social que lo diferencia de otras como Facebook, u otros. Esto consiste en que nadie que ingrese a la red social tendrá que solicitar ser amigo de nadie, ya que todos serán ya amigos de todos los demás usuarios de FraternidadWash, por eso el término de la relación entre usuarios es la relación entre “Fraternos”. Estos usuarios o frateros estarán disponibles para su búsqueda en la red social para interactuar.
5. **Creación de Grupos:** La creación de grupos de toda índole, siendo los más importantes grupos de tutorías, para el apoyo académico. Le da un toque más serio porque la aplicación nos deja inmerso en un ambiente universitario, hay que recordar que este proyecto es una herramienta de apoyo para los estudiantes. En especial porque esto deja el espacio abierto para aquellos que deseen inscribirse como tutores, usen este espacio para crear grupos de tutorías.
6. **Chat grupales o personales:** Estos chats o conversaciones serán tanto individuales como grupales, los primeros serán como el típico chat persona-persona, y los chats grupales serán aquellos realizados por varios frateros, esto será realizado con la ayuda de la biblioteca socket.io de node.js mencionada en los componentes de la plataforma en este informe. En este chat como el de las redes sociales como facebook y whatsapp dará la fecha de envío, quién vio el mensaje enviado y la opción de compartir documentos, links o fotos.
7. **Intercambio de archivos:** Ésta plataforma permite que los usuarios suban archivos, ordenados según año, asignatura y profesor, los cuales serán útiles para sus compañeros como, por ejemplo: diapositivas de materias, pruebas de años anteriores, el cual estará fijo en la web y disponible para la descarga de cualquier “Fraterno”. E incluso estará disponible el curriculum vitae de algunos compañeros, los cuales servirán como modelo

para el diseño del suyo propio a la hora de buscar empleo. E incluso se subirán fotos las cuales estarán asociados a un usuario en particular.

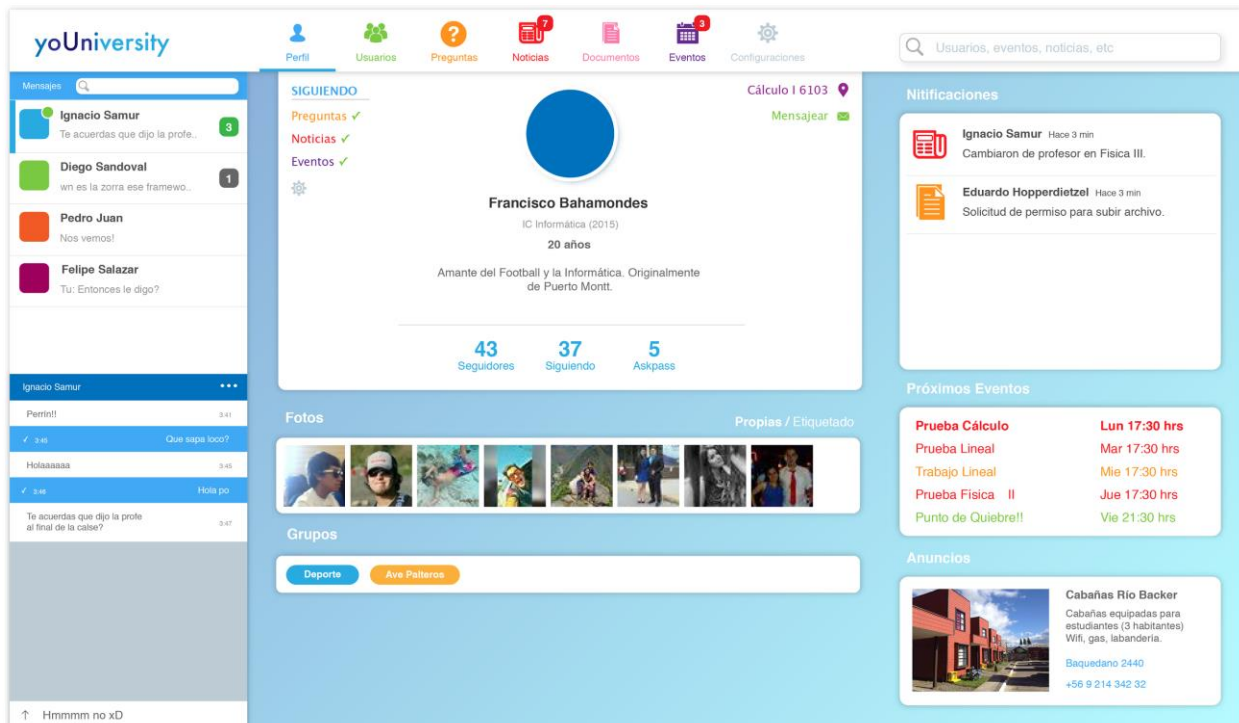
8. **Sección de eventos:** Se podrá crear eventos, categorizados por: Trabajos, pruebas, actividades, fiestas, etc. Esto puede ser hecho por cualquier “Fraterno” de la red social. Se dará la opción de seguir los eventos de alguien en específico, lo cual es útil cuando sabes de un compañero que es ordenado y mantiene todas las fechas de sus pruebas al día.
9. **Sección de Preguntas:** Será creada con el objetivo de responder distintas dudas de los estudiantes sobre distintos temas académicos de su elección, los “Fraternos” podrán responder estas preguntas, y de forma similar a “stackoverflow.com” se escogerá la mejor respuesta para la duda planteada por el estudiante. Cada usuario tendrá una cantidad de puntos inicial, que lentamente perderá al preguntar, inhabilitando de hacer más preguntas, pero podrá recuperarlos respondiendo las de otros.
10. **Sección de Noticias:** Se creará una sección de noticias con respecto a asuntos relacionados con la universidad como seminarios, resultados de tesis, premiaciones u otra noticia concerniente al ambiente universitario, donde los alumnos puedan realizar alguna reseña al respecto. Se dará la opción de ver las noticias a nivel completo de Universidad (siendo las noticias más votadas y comentadas las primeras en aparecer) o solo de los usuarios que sigas.  
Se podrá seguir las noticias, eventos y preguntas de un usuario por separado.
11. **Sección de Ajustes:** Aparecerá en el perfil de cada usuario, permitiendo cambiar datos como contraseña, fecha de nacimiento, carrera, etc. Además de la opción de reportar y bloquear a algún usuario.
12. **Sección de anuncios:** Se dará la opción de mostrar anuncios al costado derecho de la página a quienes quieran pagar por ello. (Tales como cabañas, departamentos, pensiones, tiendas, etc).

### 3) Diseño de Interfaz:

#### 1. Menú principal:

Está compuesto por 4 sectores principales:

- Barra superior, donde podrás acceder a las distintas secciones, cada una con un contador de notificaciones propio, además de un buscador de usuarios, noticias o eventos.
- Sector izquierdo, compuesto por el chat, donde se muestran las últimas conversaciones primero y la ventana para mensajear.
- Sector derecho, compuesto por una ventana de notificaciones, otra de eventos próximos, siendo el color el indicador de importancia. Y una última ventana donde se mostrarán anuncios pagados.
- Sector central, es el que mostrará la sección actual en la que te ubicas (perfil, usuarios, preguntas, etc). Este sector a diferencia del resto (que se actualiza usando sockets) se actualizará utilizando ajax.



## 2. Pantalla de Login:

El login diseñado está compuesto de tres partes:

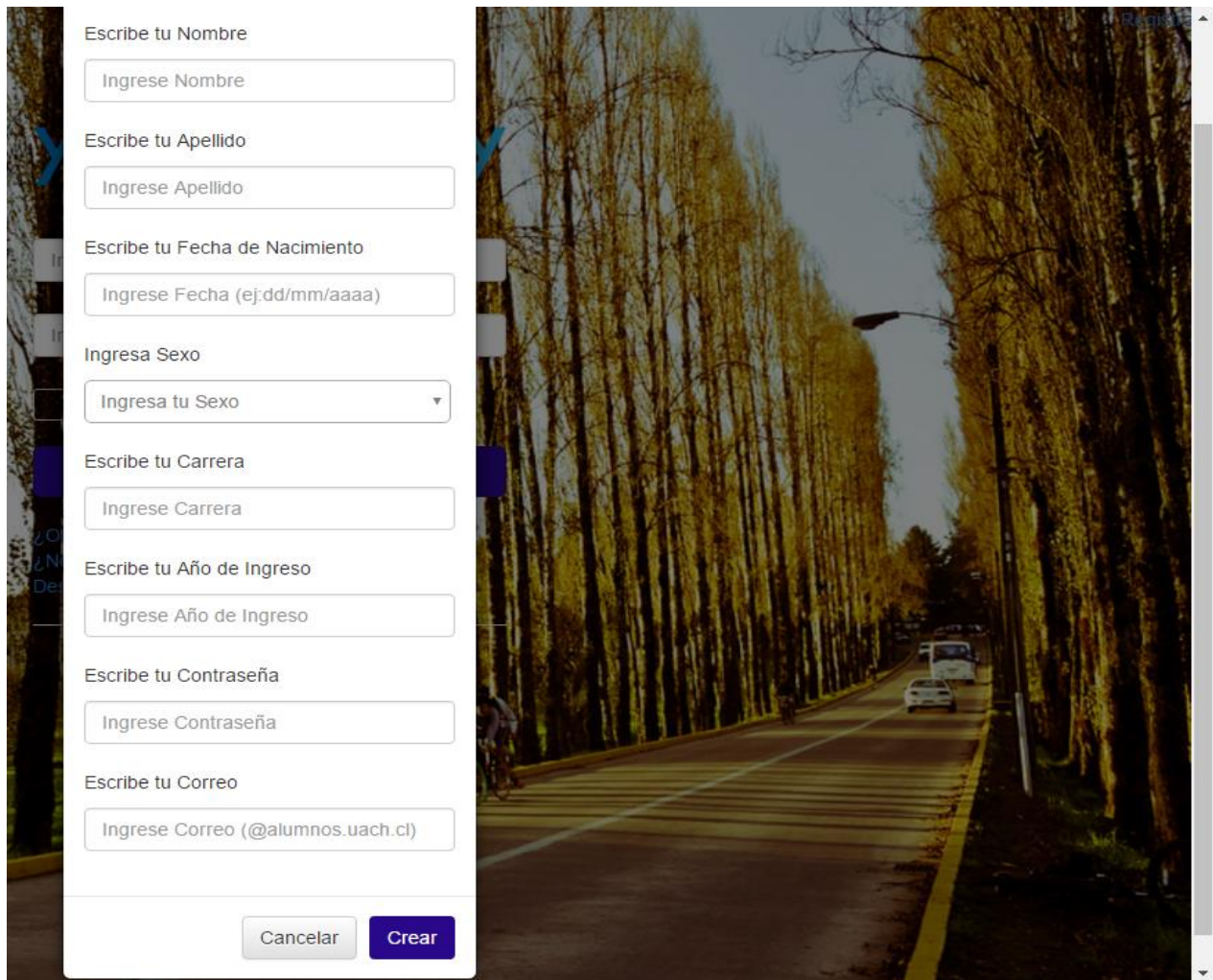
- La primera parte está compuesta por el login en sí, que será una pantalla que permitirá al usuario ingresar su correo electrónico y también su contraseña para poder acceder a la plataforma. Pero si desea ingresar al sistema sin usuario solo ingresa por medio del link “Descubre los temas del momento” (similar a lo de tumblr).
- La segunda parte del login sería el registro de usuario donde éste registrará por primera y única vez a un usuario por su correo electrónico institucional de la Universidad (@alumnos.uach.cl) y una contraseña definida donde se corresponderá con los registrados en la base de datos de la universidad para aceptar y registrar al usuario ingresado.
- La tercera parte estaría compuesta por la pantalla de recuperación de contraseña, que al ingresar un usuario se debería recuperar la contraseña, la funcionalidad aún no es definitiva y podría cambiar.

### Login Principal:





## Login de Registro:



Registro

Escribe tu Nombre

Escribe tu Apellido

Escribe tu Fecha de Nacimiento

Ingresa Sexo

Escribe tu Carrera

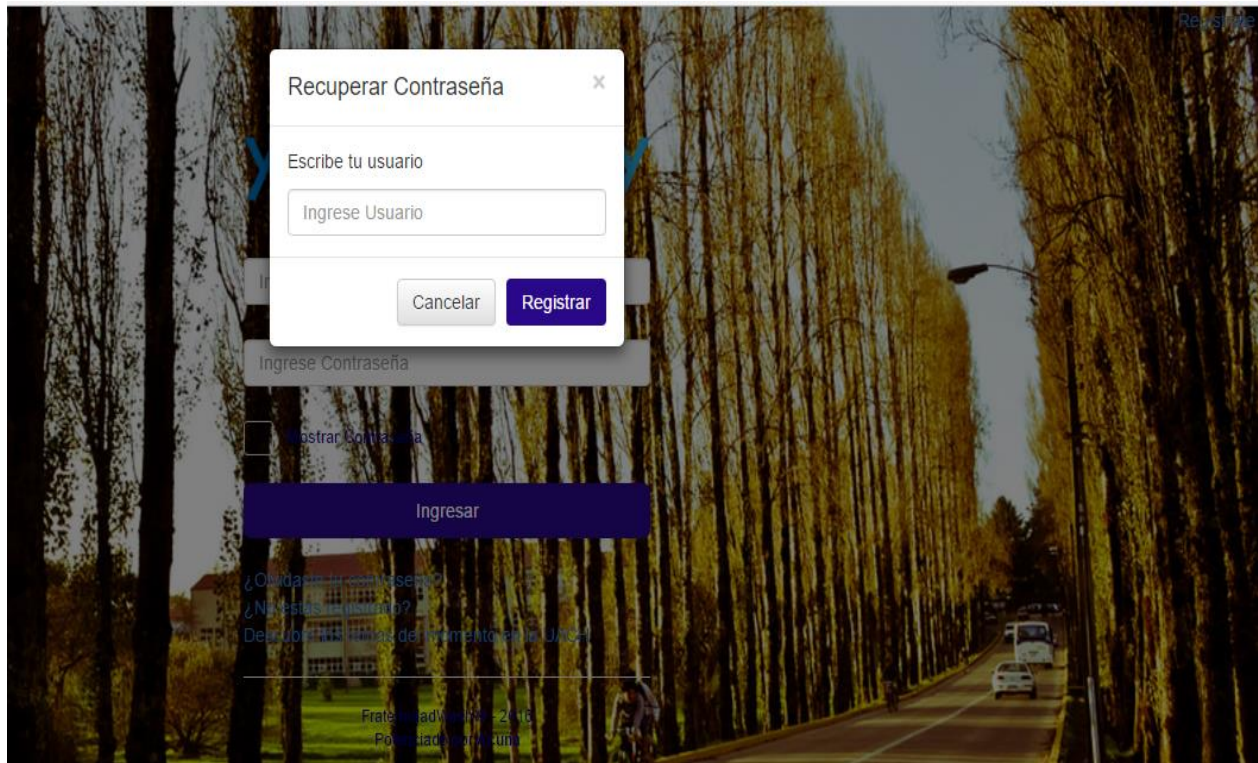
Escribe tu Año de Ingreso

Escribe tu Contraseña

Escribe tu Correo

Cancelar Crear

## Recuperación de Contraseña:



### 4) Diagrama de Componentes:

Utilizaremos Heroku que es una plataforma que ofrece servicios de computación en la nube, esta soporta distintos lenguajes de programación, y que en nuestro caso nos permite utilizar Node.js.

En Node.js utilizaremos la librería Express para facilitar la programación, la librería Socket.io para comunicación instantánea (Sobre todo para el chat). Y la base de datos MongoDB como se explicó en un inicio.

Los usuarios tendrán derecho a editar sólo su propio objeto de usuario de la base de datos a través de la interfaz, pero el resto de la información solo será accesible si el servidor lo aprueba.

Si un usuario sube una foto, se añadirá un objeto foto, en el array de fotos global de la base de datos y solo se añadirá el ID de ese objeto al array de fotos del usuario en específico. Y si el usuario quiere eliminar esa foto, solo se eliminará el ID de su propio array de fotos, pero el array de fotos global se mantendrá intacto. Esto permitirá que muchos usuarios tengan en común el mismo elemento, y se aplicará de forma similar, a las noticias, preguntas, eventos, etc.



Crearemos una carpeta llamada “public”, dónde irá todo el código del frontend, donde los usuarios tendrán acceso total. Y almacenaremos el código del backend en un directorio restringido.

Para no sobrecargar el servidor, enlazaremos el sitio con Imgur, una aplicación web que permite almacenar imágenes ilimitadamente y de forma gratuita, que además posee una API, la cual nos devuelve el link de las imágenes, luego de ser subidas, el cual almacenaremos en la base de datos.

Por último, en el frontend, solo utilizaríamos JQuery y Bootstrap.