

---

# **Documentación**

# **El Foráneo**

**versión 1.0**

**Preparado por:**

Angelo Ortega Ramírez - 2017080055  
José Evelio Castro Quesada - 2016086968  
José Gómez Casasola - 2016095929  
Alejandro Tapia Barboza - 2016167784

**22 de Enero del 2020**

# Propuesta de Proyecto

**Nombre del Proyecto:** El Foráneo

**Integrantes:** Team MaruTec

Angelo Ortega Ramírez 2017080055 [angeloale1@gmail.com](mailto:angeloale1@gmail.com)

Jose Evelio Castro Quesada 2016086968 [evelio08castro@gmail.com](mailto:evelio08castro@gmail.com)

José Gómez Casasola 2016095929 [jgomezcasasola@gmail.com](mailto:jgomezcasasola@gmail.com)

Alejandro Tapia Barboza 2016167784 [aletapiab28@gmail.com](mailto:aletapiab28@gmail.com)

## **Descripción:**

Se realizará una página web para el manejo de recetas, donde se podrán realizar varias acciones tales como búsqueda de recetas, calificación de recetas, ingreso de recetas nuevas, plan de comidas, los pasos para realizarlo entre algunas otras más.

## **Justificación:**

Debido a la necesidad prevista de muchos estudiantes en las labores de cocina, se quiere mejorar la calidad de vida de los estudiantes que residen en apartamentos y residencias en las cercanías de universidades, por su cuenta, con el fin de que dichos jóvenes puedan mejorar su alimentación, teniendo una página web con la cual puedan buscar recetas que se adecuen a sus necesidades y capacidades. De esta manera, el estudiante puede buscar recetas para todos los tiempos alimenticios, es decir, desayuno, almuerzo y cena.

## **Productos Esperados:**

### **Alcance:**

El proyecto tiene la meta de proveer una plataforma web orientada a estudiantes universitarios para el manejo de planificación de comidas. Se plantea ayudar a presupuestar, planificar y administrar comidas, mediante ingredientes y recetas. Cada usuario va a poder incluir sus propias recetas y encontrar nuevas recetas. La aplicación viene a solventar una necesidad latente entre los estudiantes, un espacio en donde conseguir nuevas recetas, guardarlas en lista de favoritos y poder determinar qué hacer con un conjunto de ingredientes. Las recetas van a poder ser consultadas con diferentes criterios de búsqueda. Entre algunas consideraciones, no se incluye el manejo de inventario de ingredientes ni la inserción de recetas por parte del equipo administrativo.

### **Requerimientos Funcionales:**

- Ingresar recetas.
- Guardar y compartir recetas.

- Calificar recetas.
- Búsqueda de recetas: por nombre, puntaje, ingredientes, precio, usuario.
- Generación de plan de comidas parametrizado (por semana, rango de precios, ingredientes).
- Lista de recetas favoritas.
- Desplegar pasos de receta.

#### **Cronograma:**

Fecha	Descripción
16/12/19	Anteproyecto
23/12/20	Prototipo
13/01/20	Especificación funcional
22/01/20	Documentación
22/01/20	Presentación Final

## **Lista de Funcionalidades Ofrecidas**

Para el desarrollo del proyecto se propusieron una serie de requerimientos funcionales. A continuación, se dará una explicación detallada de cada uno de ellos.

### **Registro de nuevos usuarios**

Los nuevos usuarios de la aplicación, que no poseen una cuenta en El Foráneo deberán llenar el formulario de creación de nuevas cuentas, en el cual el usuario debe ingresar datos personales, los cuales son:

- Nombre
- Apellidos
- Correo electrónico
- Username
- Contraseña

Con dicha información suministrada, el usuario dará click al botón “Regístrate” y el sistema se encargará de validar que no exista otro usuario con dicho correo o username, y en dicho caso registrará al nuevo usuario.

A continuación se presenta como, según el prototipo realizado del sistema el Foráneo, se visualizará dicho apartado:

Formulario de creación de cuenta con el título "Crea una cuenta" en una barra superior naranja. El fondo es oscuro verde. Hay cinco campos de entrada blancos con bordes redondeados: "Nombre", "Apellidos", "Username", "Password" y "Celular". Debajo de los campos hay un botón naranja con el texto "Registrate".

Figura 1

### Iniciar Sesión

Los usuarios al iniciar sesión deberán suministrar su username y contraseña con el objetivo de poder acceder a su cuenta. En caso de que los datos no sean correctos, el sistema desplegará un mensaje indicando al usuario la condición.

La siguiente imagen presenta como se verá dicho apartado en la aplicación:

Formulario de inicio de sesión con el título "Inicio de Sesión" en una barra superior naranja. El fondo es oscuro verde. Hay dos campos de entrada blancos con bordes redondeados: "Username" y "Password". Debajo de los campos hay un botón naranja con el texto "Iniciar Sesión".

Figura 2

### Ingresar recetas

Una vez ingresado los usuarios, estos podrán subir sus recetas personalizadas. Para esto el usuario deberá ingresar datos como el nombre de la receta, tiempo estimado, ingredientes, pasos para la elaboración, así como una imagen de la receta. Una vez listo los datos, se dará en agregar receta y está será ingresada al sistema, donde podrá ser vista por los demás usuarios, así como compartida y añadida a favoritos. La siguiente imagen, muestra cómo ha sido diseñado este apartado en el prototipo.

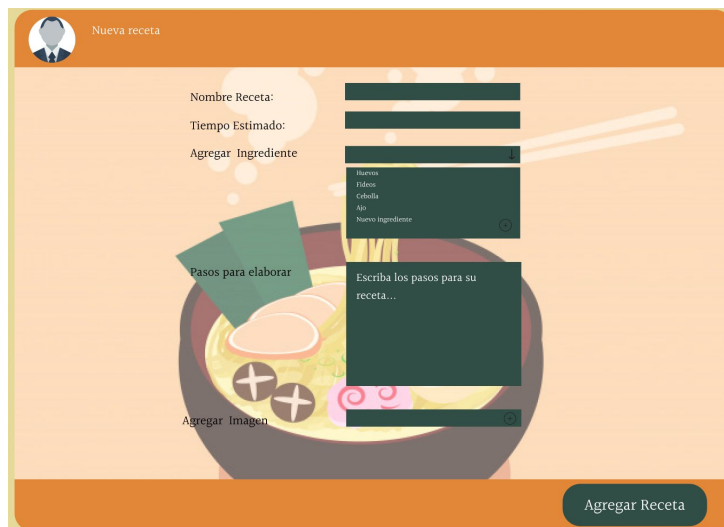


Figura 3

## Guardar y compartir recetas

Si el usuario ve una receta que le llama la atención, podrá guardar en la sección de favoritos, para después encontrarla de una manera más rápida. Además de esto, tendrá la opción de compartirla en su propio muro, para que los demás usuarios que lo siguen, puedan observar la receta también. Dichas funcionalidades se realizarán mediante los siguientes botones.



Figura 4

## Calificar recetas

Cuando usted ve una receta, está tendrá la opción de calificarla. Cada receta podrá tener puntuación de 1 a 5. El sistema de calificación será representado por el logo de la aplicación, y tendrá un comportamiento igual a la calificación con estrellas. La puntuación se verá reflejada en la parte inferior izquierda de la receta y será el promedio de calificación de todos los usuarios. Cada usuario puede calificar la receta según así lo convenga. Este sistema de calificación ayudará al usuario a ver la calidad de la receta, (según la opinión de los demás usuarios), así como en la búsqueda de nuevas recetas, donde se podrá filtrar por este aspecto. En la siguiente imagen, podemos ver que esta receta está calificada con una puntuación de 5.



Figura 5

## Búsqueda y filtrado de recetas

Los usuarios de la aplicación web El Foráneo podrán realizar búsquedas de recetas en la aplicación ingresando el nombre de la receta que desean buscar. Además podrán realizar búsquedas mediante el uso de filtrados por categoría de platillo, rangos de precios y puntaje como se observa en la siguiente imagen.



Figura 6

Al realizar la búsqueda o filtrado, las recetas que se obtienen según la búsqueda, se visualizan de la siguiente manera, como se definió anteriormente en el prototipo de la aplicación:

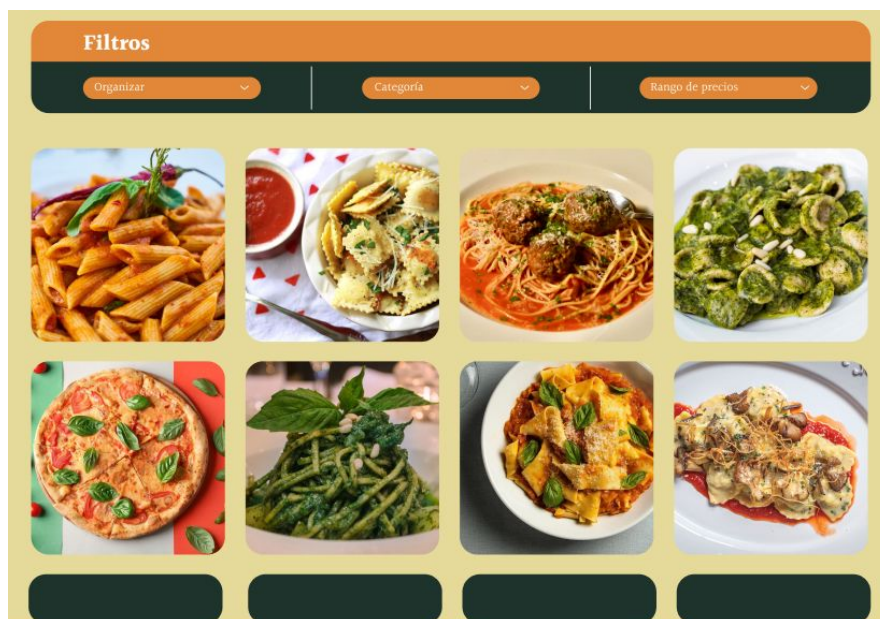


Figura 7

## Generación de plan de comidas parametrizado

En dicha sección, los usuarios podrán crear planes de comidas parametrizados según sus necesidades y las recetas que existan en la aplicación, donde el usuario podrá generar los planes según la cantidad de comidas que desee realizar, las categorías, y los rangos de precios que estén a su alcance. Dicha caja de selección para generar planes se visualizará de la siguiente manera.

**Generación de plan de comidas**

¿Qué desea comer?

Categoría ▼

¿A qué precio?

Rango de precios ▼

¿En cuántas comidas?

Cantidad

Generar

Figura 8

Al generar exitosamente su plan alimenticio, este se desplegará en pantalla con los respectivos platillos designados para el usuario, los cuales se mostrarán según lo establecido, en el prototipo de la aplicación, de la siguiente forma.

**Desayuno**

Usuario Apellido  
45 min

Ingredientes:  
-Huevos  
-Fideos  
-Cebolla  
-Ajo  
-Entre otros

Precio estimado:  
₡ 1.500

**Almuerzos**

Usuario Apellido  
45 min

Figura 9

Como último, aparecerá un resumen del plan alimenticio generado, donde se presentarán datos como el costo total, la cantidad de desayunos, almuerzos y cenas que conforman el plan y los ingredientes, como se presenta a continuación.



Figura 10

### Listado de recetas favoritas

En la sección principal, aparecerán, en un costado, el listado de recetas que el usuario haya marcado como favoritas. Así, este podrá acceder sus recetas favoritas de una manera fácil y sencilla. En la siguiente imagen, se puede observar cómo se desplegará el listado de favoritos



Figura 11

### Desplegar pasos de recetas

Al dar click sobre una receta, se desplegará en pantalla toda la información referente a la misma, con el fin de que el usuario pueda ver todos los detalles, como lo son los ingredientes, pasos a seguir, el precio estimado, una imagen de cómo se verá la receta finalizada y demás datos, por ejemplo, la cantidad de porciones que se obtienen al realizar la receta. Además, el usuario podrá ver quién publicó la noticia, podrá guardarla en favoritos, calificar la receta y compartirla.

En la siguiente imagen se puede observar como se verá dicho apartado, según lo estipulado en el prototipo:

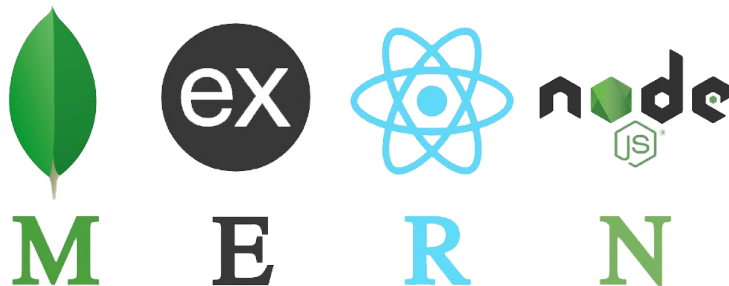




Figura 12

## Arquitectura

### Descripción y justificación



Para la creación de la aplicación se utilizó un web stack basado en MERN, sustituyendo Mongo por MySQL. MERN esta compuesto por el front-end de la aplicación realizado en ReactJS, framework de interfaces gráficas desarrollado por Facebook. Para el lado del back-end a nivel de base de datos se escogió una base de datos MySQL instalada mediante XAMPP, la manipulación de los datos de esta base de datos se realiza mediante Express y NodeJS, Express nos permite hacer el routing de la aplicación de servidor de forma que tenemos una estructura para manejar las peticiones que se reciben del cliente y responder a ellas de acorde a lo necesario, reflejando los cambios de manera permanente en la base de datos para permitir una aplicación extensible y dinámica. De esta forma se creó un API en Express para hacer que el servidor fuese lo más independiente posible de la solución de front-end utilizada. Esta es una práctica común ya que permite que con un mismo servidor se puedan generar múltiples soluciones de front-end que trabajen sobre los

misimos datos. MySQL fue utilizada ya que es una base de datos ligera relacional que cumple con todas las características necesarias para poder realizar una aplicación rápida y escalable. ReactJS fue utilizado ya que cuenta con una curva de aprendizaje corta en comparación con otros frameworks en front-end como lo es Angular ya que solo lidia con la parte de vistas y no se mete ni con el modelo ni con el controlador, dejando al usuario cómo se crearan estos. Esto en conjunto con la reutilización de componentes y el rápido y responsivo manejo del DOM permite que realizar aplicaciones web en react resulte sencillo y efectivo.

## Desglose de trabajo

Para la división del trabajo se optó por hacer un “split” de las funcionalidades del front-end y las del back-end para poder ir las desarrollando en paralelo de la forma más rápida posible. José Evelio Castro y José Gómez Casasola se encargaron de trasladar el prototipo a una solución escrita en React la cual utiliza componentes de forma que estos pudieran ser poblados con información de la base de datos. La creación del modelo de la base de datos, los procedimientos almacenados, los triggers y el acceso al mismo mediante express fue hecho por Angelo Ramírez y Alejandro Tapia. De esta forma se crearon tanto el front-end como el back-end de manera paralela y luego se trabajó en juntar su funcionalidad, lo cual resultó efectivo ya que todos tenían en mente lo que se necesitaba lograr y los pasos necesarios para lograrlo.

### División de tareas

Sabiendo que algunos de los siguientes elementos fueron manipulados por más de un usuario, entiéndase que las menciones duplicadas del mismo se deben a dicho acto.

#### Angelo Ortega


1. Creación del modelo de la base de datos el cual cubre todas las necesidades de manejo de datos de la aplicación.
2. Creación de procedimiento almacenado de generación de planes de comida.
3. Creación de triggers después de inserción para el manejo de notificaciones.
4. Creación de un API en Express para el control y manejo de datos.
  - a. Control de rutas.
  - b. Manipulación de recetas
  - c. Consultas sobre
    - i. Recetas
      1. Calificación
      2. Favoritos
      3. Compartidos
    - ii. Usuarios.
      1. Datos
      2. Notificaciones

- 3. Recetas favoritas
    - 4. Recetas creadas
    - 5. Recetas compartidas
  - iii. Categorías.
- 5. Enlace de la funcionalidad de las siguientes ventanas con sus respectivos controladores.
  - a. Página de perfil de usuario
  - b. Login y registro
  - c. Búsqueda de recetas
  - d. Filtrado de recetas
  - e. Pantalla principal
- 6. Documentación.

### **Alejandro Tapia**

- 1. Creación del modelo de la base de datos.
- 2. Creación de un API en Express para el control y manejo de datos.
  - a. Inserción de recetas
  - b. Consultas sobre
    - i. Recetas
      - 1. Calificación
      - 2. Favoritos
      - 3. Compartidos
    - ii. Usuarios.
      - 1. Datos
    - iii. Categorías.
- 3. Enlace de la funcionalidad de las siguientes ventanas con sus respectivos controladores.
  - a. Like a receta
  - b. Compartir receta
  - c. Calificar receta
  - d. Agregar receta
- 4. Documentación.
- 5. Especificación funcional.

### **Evelio Castro**

- 1. Elaboración general del prototipo.
- 2. Incorporación de imágenes e iconos.
- 3. Creación de los archivos:
  - a. busqueda.js: archivo concerniente a la pantalla que visualiza los resultado de una búsqueda (fig. 7)
  - b. busqueda.css: archivo encargado de estilizar busqueda.js
  - c. notificacion.js: archivo concerniente al cuadro de notificaciones desplegadas una vez se da clic sobre icono  ubicado en la barra de encabezado.
  - d. Notificacion.css: archivo encargado de estilizar notificacion.js

- e. nuevaReceta.js: archivo que contiene el formulario donde se ingresan nuevos productos (fig. 3)
- f. tabla.js: archivo encargado de generar una tabla en donde se pudiesen registrar ingredientes, precio y cantidad, que se incorpora dentro de nuevaReceta.js
- g. tabla.css: archivo encargado de estilizar tabla.js
- 4. Modificación de los archivos:
  - a. barra.js: header que se incorpora en todas las pantallas y que contiene los iconos de conexión entre diversas pantallas.
  - b. principal.css: archivo encargado de estilizar barra.js, perfil.js, y principal.js
- 5. Documentación.

### **José Gómez**

- 1. Elaboración general del prototipo.
- 2. Incorporación de imágenes e iconos.
- 3. Creación de los archivos:
  - a. app.js: siendo este el index, se presentan dos formularios correspondientes al *log In* y *sign up*.
  - b. app.css: archivo encargado de estilizar app.js.
  - c. barra.js: header que se incorpora en todas las pantallas y que contiene los iconos de conexión entre diversas pantallas.
  - d. infoReceta.js: expande el contenido de una publicación y muestra de manera completa la información de la susodicha.
  - e. infoReceta.css: archivo encargado de estilizar infoReceta.js.
  - f. principal.js: archivo que muestra la pantalla a modo de feed para las publicaciones.
  - g. principal.css: archivo encargado de estilizar principal.js y publicacion.js.
  - h. publicacion.js: archivo para mostrar el contenido concerniente en una receta publicada.
- 4. Especificación funcional.
- 5. Documentación

### **Experiencias de cada uno**

#### **Angelo Ortega**

Desarrollar el back-end del proyecto me permitió entender mejor cómo funcionan las aplicaciones web a nivel de servidor, específicamente aquellas basadas en APIs y los beneficios y desventajas de utilizar un modelo así para la capa de control de la aplicación. Con esto logré reforzar mis conocimientos sobre bases de datos utilizando profundamente por primera vez MySQL. Adicional a esto aprendí sobre como ligar la lógica de negocios de una aplicación web con un front-end basado en react y las ventajas que generan los componentes reactivos ya que permite refrescar la página de manera dinámica y asíncrona y desplegar y modificar datos on-the-go

con muchísima facilidad. También enfrente complicaciones a la hora de trabajar con frameworks nuevos como Express pero gracias a la extensa comunidad y documentación fue posible encontrar soluciones a todos los problemas particulares encontrados.

### **Alejandro Tapia**

Enfrentando el reto de desarrollar el back-end me dejó entender la lógica manejada en los servidores. La lógica no fue una dificultad mayor. Cuando manejé un poco del front-end, comprendí cómo funcionaban los componentes de React y cómo se aprovechaban incorporándose en otros componentes. Esto me ayudó para crear los servicios, obtener los datos desde la pantalla y hacer las conexiones pertinentes.

Express me detuvo en el avance del proyecto por no conocerlo, pero tomado el tiempo para superar la curva de aprendizaje, además de aprovechar la experiencia de mis compañeros, se convirtió en un herramienta verdaderamente útil.

### **Evelio Castro**

Dedicándome al front-end del proyecto en una herramienta que me era desconocida me permitió expandir mis entornos de desarrollo y descubrir mis flaquezas respecto al cambio de herramientas. También descubrí el alcance referente a la cantidad de ambientes en los que se pueden desarrollar distintos paquetes que ya me eran conocidos.

Comparando contra conocimientos anteriores de Angular, pude comprobar la velocidad en la recarga dinámica del servidor, siendo React superior a Angular en este aspecto; pues incluso en archivos de estilo el mismo no ocupa recargarse completamente para incorporar los cambios, como si lo hace Angular.

La mayoría de dificultades los hallé en el paso de parámetro entre componentes y la incorporación de un componente dentro de otro, pues la renderización de React me era nueva y las estructuras, aunque similar en aspecto a Angular, eran de implementación diferente.

### **José Gómez**

El haber realizado labores de front-end en el desarrollo de la aplicación web El Foráneo me permitió ampliar mis conocimientos de diseño de interfaces gráficas de páginas web, ya que React.Js brinda gran cantidad de elementos y funciones que permiten realizar un concepto más flexible y ágil en cuanto a front-end.

Para mi persona, React no era del todo desconocido ya que en un proyecto del ITCR había utilizado React Native, el cual me brindó la capacidad de poder desenvolverme de una más ágil en cuanto al desarrollo con React, además de permitirme reforzar mis conocimientos en cuanto a estándares y métricas para el desarrollo con dicha herramienta, lo cual no sólo fue enriquecedor para mí, ya que con esto, pude ayudar a mis compañeros y así compartir los conocimientos que poseía para realizar el proyecto en conjunto.

# Análisis de resultados

## Resultados esperados vs. resultados obtenidos

### Registro de nuevos usuarios

Compleitud: 100, los usuarios se agregan satisfactoriamente y pueden iniciar sesión, adicionalmente cuenta con las respectivas validaciones necesarias

### Iniciar Sesión

Compleitud: 100, Los usuarios registrados pueden iniciar sesión satisfactoriamente, adicionalmente cuenta con las respectivas validaciones necesarias

### Ingresar recetas

Compleitud: 100, un usuario registrado puede acceder a la ventana de creación de recetas y definir sus pasos, ingredientes, tiempo de preparación y demás atributos y agregarlo a la aplicación para que el resto de usuarios pueda verla, calificarla, agregarla a favoritos y compartirla en su perfil.

### Guardar y compartir recetas

Compleitud: 100, un usuario registrado puede ver todas las recetas y guardarlas dentro de favoritas al hacer click en el botón de favorita y compartirla en su perfil con la opción de compartir. Ambas opciones generan una notificación al usuario creador de la receta. Adicionalmente, si un usuario da clic una segunda vez a cualquiera de estas opciones las anula.

### Calificar recetas

Compleitud: 100, un usuario registrado puede ver todas las recetas y calificarlas en un rango del 1 al 5 al hacer click en la barra de rating. Esto genera una notificación al usuario creador de la receta diciéndole cuánto le han dado de calificación a su receta. Adicionalmente, si un usuario da clic una segunda vez actualiza la calificación.

### Búsqueda y filtrado de recetas

Compleitud: 100, un usuario registrado puede ver todas las recetas y filtrarlas sobre su tiempo estimado de preparación, categoría y precio, esto se puede hacer tanto en la pantalla de búsqueda como en la pantalla principal. El usuario puede ingresar el criterio de búsqueda en la barra y se desplegarán todas las recetas que cuenten con el criterio de búsqueda como parte de su cuerpo o de su nombre.

### Generación de plan de comidas parametrizado

Compleitud: 100, un usuario registrado puede ingresar a la pantalla de generación de plan de comidas y aquí se le desplegarán las opciones de categoría, precio y cantidad. Se consulta en la base de datos por recetas que cumplan con estos criterios y se genera un plan para llenar la cantidad de comidas necesarias. Al final

de la página se despliegan los datos referentes a cantidad de comidas y desglose en platillos y precio así como un listado de los ingredientes necesarios para prepararlos.

### **Listado de recetas favoritas**

Compleitud: 100, un usuario registrado puede ver todas las recetas a las cuales le ha dado “favorito” en su muro principal.

### **Desplegar detalles de recetas**

Compleitud: 100, un usuario puede consultar los detalles de una receta en la pantalla principal, así como hacer click sobre la imagen de una receta para ver los detalles en grande. Los detalles incluyen la categoría, el tiempo de preparación, el usuario, los ingredientes y sus cantidades y el precio final.

### **Desplegar notificaciones**

Compleitud: 100, un usuario puede hacer click en la barra de tareas sobre el botón de notificaciones y un popup será generado dónde se muestra las interacciones de distintos usuarios con sus recetas.

### **Desplegar perfil de usuario**

Compleitud: 100, un usuario puede consultar los detalles de su perfil en donde se despliega su nombre completo, las recetas las cuales ha compartido, las recetas de su autoría y sus recetas favoritas.

## **Opiniones personales del proyecto y recomendaciones para el curso**

### **Angelo Ortega**

El proyecto me dió la oportunidad de aprender nuevas tecnologías que no se tocan normalmente en el plan de estudios de la carrera. Logré reforzar lo ya aprendido en Bases de Datos sobre modelaje e interacción con la base de datos y además aprendí a realizar un API que recibe diferentes tipos de parámetros mediante requests HTTP y realiza validaciones y modificaciones sobre la base de datos. Adicional a esto aprendí sobre React, herramienta la cual se encuentra muy en demanda en el mercado laboral la cual nos ayudó a facilitar la creación de los distintos componentes para realizar una aplicación interactiva con múltiples usuarios de manera sencilla y eficaz. Todo esto me permitió entender más a fondo cómo funciona la web y a reforzar los conceptos fundamentales como lo son HTML, CSS y JavaScript. Con respecto al curso, me gustó que se empezó cubriendo las bases del desarrollo web y se nos contextualizara con respecto al mercado laboral mediante la exposición de diferentes herramientas de front-end y back-end. Esto me permitió generar un criterio propio acerca de las herramientas y entender mejor cómo funcionan las aplicaciones a nivel WEB.

### **Alejandro Tapia**

Respecto al proyecto me ayudó a retomar conceptos y prácticas para interactuar con bases de datos de tipo relacional, y aprendí a manipular parámetros mediante

requests HTTP. Gracias al proyecto, laboratorio 3 y el taller, junto con la exposición, aprendí sobre React, su historia, usos, características, ventajas, desventajas, funcionalidades y métodos. Además pude darme una noción de qué tan demandada es la librería por la cantidad de aplicaciones que la aprovechan.

Sobre el curso me pareció que dividir cada herramienta era una solución elegante para poder abarcar todas las herramientas y que los talleres permitían entender lo suficientemente bien si la tecnología que se explicaba era una tecnología afín a cada estudiante. Los laboratorios, con excepción del primero, fueron de mi agrado. Poder visualizar rutas era un trabajo que nunca había implementado. Además que aprendí sobre librerías y páginas dedicadas al routing.

### **José Evelio**

El proyecto me permitió comprender las subdivisiones posibles entre front-end y back-end y el cómo dividir el trabajo en grupo. Además me brindó acceso a tecnologías que son usadas laboralmente en la actualidad.

Sobre el curso me pareció excelente la exploración de distintas tecnologías, y los talleres para tener noción de las mismas. La estructura, en que se nos dio a conocer los elementos principales de desarrollo web (HTML, CSS y Javascript) para adentrarse, pero sin quedarse más de lo necesario en ellas.

El uso de PHP me resultó molesto (por el lenguaje en sí mismo), pero el que se nos diera a conocer me permitió tener nociones de su funcionamiento en caso de dar mantenimiento en un futuro.

Realmente aplaudo la idea de explorar la cantidad máxima de herramientas relativas al curso, puesto que el darlas a conocer nos permite no ser extraños en ambientes (principalmente laborales) donde sí se están utilizando.

### **José Gómez**

Acerca del proyecto, este me permitió trabajar de una manera acorde a lo que se vive en el ámbito laboral real, dividiendo las labores entre los compañeros en cuando a back-end y front-end, donde un subgrupo se encargó exclusivamente del front-end y otro del back-end. También, el curso en general, y el proyecto, cumplió con el objetivo de demostrar a los alumnos un gran conjunto de frameworks para trabajar, y así se logró escoger de manera acertada lo que era la escogencia de la tecnología a escoger, a pesar de no haber realizado los talleres para el momento del inicio del proyecto, al conocer qué tecnologías se expondrían, el grupo investigó sobre estas y eligió correctamente cuál utilizar, y con esto, me permitió ampliar mis conocimientos en cuanto a tecnologías web, la cual antes del curso eran pocos.

Me encantó la mecánica del curso ya que iniciamos con aplicaciones web básicas. donde se utilizaba sólo archivos html, js y css, luego PHP, hasta realizar aplicaciones con frameworks como React y Node.js / Express.js donde ya las aplicaciones manejan de una manera mucho más estructurada el manejo del back-end y el front-end por aparte donde los equipos pueden realizar sus tareas de una manera mucho más objetiva.