

# Capítulo 1

## Descripción de la aplicación

En el capítulo anterior ?? se ha introducido tanto los antecedentes como se describió brevemente la aplicación. En este capítulo explicaremos toda las funcionalidades y sus características con detalle.

Partimos de qué tipo de aplicación queremos crear. Se trata de desarrollar un software que ayude a las personas a gestionar el costo de producción de las recetas, dependiendo del precio de sus ingredientes y el número de raciones. Ésta se llamará **ChefManagement** y sus características iniciales son:

- Crear, ver, editar y eliminar recetas.
- Listar todas las recetas.
- Crear backup de las recetas y cargarlo si es necesario.
- Calcular el precio de una determinada receta para un determinado número de comensales.

Para esta primera versión, además de ser capaz de realizar esta serie de tareas, debe cumplir con las necesidades y expectativas del usuario y facilitar su interacción con la aplicación. Éstos son sus requisitos iniciales:

- Dos entornos de producción: la aplicación puede ser usada desde dos nubes diferentes.
- Tener en cuenta aspectos de usabilidad: adaptación, entendimiento y facilidad de uso.
- Importar y exportar archivos en formato json: almacenar copias de las recetas en el equipo cliente y restaurarlas si es necesario.
- Acceso mediante registro o haciendo uso de APIs de distintas redes sociales como Google+ o Facebook. Hay que facilitar el acceso a los usuarios.

Tanto las características como los requisitos de la aplicación serán aplicadas a la versión 1.0, pudiendo ser mejorados o añadidos más en futuras versiones ??.

## 1.1. Acceso a la aplicación

Para usar esta aplicación podemos acceder directamente a sus entornos de producción:

- Despliegue en Heroku
- Despliegue en OpenShift

O bien acceder al repositorio de la aplicación (se trata de licencia libre y código abierto) y seguir las instrucciones de instalación.

Una vez en la aplicación encontraremos una pantalla inicial de acceso, que nos permite acceder usando redes sociales, en este caso Google+ y Facebook. Esto es posible gracias al uso de su API (o interfaz de programación de aplicaciones) correspondiente, la cual es un conjunto de funciones o métodos que ofrece una librería para ser utilizado por otro software. En otras palabras, el equipo de desarrollo de Facebook crea esta herramienta para que otras aplicaciones, como Chefmanagement puedan hacer uso de ella. Esto tiene ventajas importantes como la seguridad, desde el punto de vista de desarrollo de software y la comodidad para los usuarios finales.

Figura 1.1: Acceso a la aplicación

Además existe la opción de validarnos en la aplicación, una vez que se haya completado el registro. Si el usuario de entrada o contraseña fueran incorrectos la aplicación lo notificaría. En caso de pérdida de contraseña podemos solicitar una nueva accediendo al enlace *Forgot your password?* y proporcionar nuestro nombre de usuario. Posteriormente puede modificarse la nueva contraseña en opciones ??.

Figura 1.2: Recuperar contraseña

Para registrarse accedemos a través del enlace *New account*. En caso de cualquier problema durante el proceso de registro, como que el correo o nombre de usuario estén en uso, la aplicación lo notificará al usuario.

Figura 1.3: Registro

Una vez dentro de la aplicación accederemos al *home* del usuario. Existe un barra superior donde podemos acceder a la configuración y salir de la aplicación

(Log out). También encontramos un menú lateral el cual facilita la interacción del usuario con la aplicación y ayudar a moverse y gestionar las distintas tareas. Por defecto, cuando accedemos a la aplicación, en *home* se nos despliega la lista de recetas creadas por ese usuario. Las opciones del menú son las siguientes:

- Recipe list: muestra el listado de recetas del usuario. Podemos acceder, editar y borrar recetas a través de esta lista.
- Recipe calculator: calcula el costo de producción de una receta y sus ingredientes para una determinada receta y raciones.
- New recipe: crea una nueva receta.
- Import: carga una copia de seguridad de las recetas del usuario.
- Export: almacena en el equipo local una copia de seguridad de las recetas del usuario.

## 1.2. Crear recetas

Para acceder a crear receta, nos situamos en el menú lateral y pinchamos en **New recipe**. Esto nos llevará a una nueva ventana, la cual presentará un formulario con una serie de campos que se detallan a continuación:

- Name: es un campo obligatorio y clave primaria en la bbdd. Nombre de la receta.
- Rations: Numero de raciones para crear esa receta. Este campo es muy importante ya que se tomará como referencia para el cálculo de recetas dado un determinado número de platos.
- Cost: es un campo de lectura, se calcula automáticamente con cada ingrediente.
- Ration cost: relación entre el coste de la receta y el número de raciones.
- Order: campo que sirve como etiqueta. Indica si se trata de un primer o segundo plato, un postre, etc.
- Type: segundo campo de etiqueta. Nos da información sobre el tipo de comida: snack, comida casera y otros.
- Nivel: nivel de dificultad de producción de la receta.
- Time: horas y minutos que se estima que lleva preparar esa receta.
- Vegan: indica si la receta es apta para vegetarianos.
- Allergens: campo de texto para introducir alimentos que puedan producir alergias.
- Origin: información sobre el país originario de la receta.

Una vez completado podemos guardar la receta con el botón **Save recipe** o cancelar el proceso y volver a *home*. Si guardamos la receta el sistema notificará de su registro en la bbdd y abrirá nuevos campos:

- Un área de texto desplegable que se utiliza para escribir las instrucciones de la receta. He utilizado la herramienta ckeditor para integrarla en la aplicación-
- Ingredientes, que puede ser:
  - Un **nuevo ingrediente** creador por el usuario. Campos:
    - Nombre del ingrediente
    - Cantidad: puede ser: peso, volumen o una determinada unidad.
    - Precio en relación con su cantidad
    - El porcentaje de merma o pérdida del ingrediente durante su preparación.
  - **Una receta** ya creada por el usuario. De forma que las recetas puedan servir de base para crear otras, por ejemplo, la salsa de tomate es una receta que se puede usar en otras. Si una receta usa otra, la primera no puede ser usada para crear otra y no aparecerá como opción en este apartado.

A medida que vayamos incluyendo ingredientes a la receta, en el margen lateral derecho se desplegará una lista informativa de la receta con los ingredientes agregados hasta el momento.

### 1.3. Editar y eliminar recetas

Desde el listado de recetas podemos editar o eliminar según nuestras necesidades. Para ello pinchamos en una determinada receta y accedemos a toda la información de la receta. Esta opción es de sólo lectura. En la parte superior derecha encontramos los iconos para eliminar y editar la receta.

Para eliminar la receta pulsamos el botón de eliminar. El programa pedirá la confirmación para eliminarla. Una vez eliminada no se puede recuperar. Las recetas que sirven de base a otras no pueden ser eliminadas hasta que no se rompa este vínculo, para ello debemos entrar en el modo edición.

Si pulsamos en edición, nos llevará a una nueva pantalla donde podemos editar todos los campos de la receta (excepto el nombre y los costos que se calculan automáticamente). Podemos añadir nuevos ingredientes, editar o borrar los actuales o bien borrar *recetas base* (en futuras aplicaciones se recomienda poder añadir *recetas bases*). Así mismo podemos editar las instrucciones de la receta.

## 1.4. Calculadora

Esta opción nos permite elegir entre el listado de recetas del usuario. Una vez hecho esto seleccionamos el número de raciones que queremos producir y pulsamos en calcular. Se mostrará una tabla resultante con el listado de ingredientes, sus cantidades para las raciones seleccionadas y su precio. De la misma forma se mostrarán las *recetas bases* si contiene alguna y por último el precio de producción final.

A modo de información adicional se muestra las instrucciones de la **receta original**.

## 1.5. Importar y exportar

Estas opciones nos permiten crear y cargar una copia de seguridad de las recetas y sus ingredientes del usuario. Para crear una copia, seleccionamos la opción **Export** del menú lateral. Se mostrará un campo de texto para elegir el nombre del archivo, (por defecto *recipe*) el cual podemos editar y pulsamos en crear. Se comprueba si la cadena introducida contiene caracteres inválidos, en caso contrario se notifica al usuario la operación exitosa y se descarga el archivo al equipo local en formato **json**.

Por otro lado, está la opción de cargar la copia de seguridad. Para ello pulsamos en la opción **Import**, nos permitirá seleccionar un archivo solo válido en formato json. Una vez seleccionado pulsamos en cargar, en caso de error, la aplicación lo notificará con un mensaje, y si el archivo se ha cargado correctamente nos redirige a *home* con el listado de recetas. Hay que tener en cuenta que si cargamos una copia de seguridad borrará el contenido actual de recetas para cargar ésta.

## 1.6. Opciones

Las opciones se encuentran en la parte derecha de la barra superior de la aplicación. Si pulsamos nos lleva a una nueva pantalla que nos muestra el perfil del usuario. Éste se podrá editar en caso de que el usuario haya accedido mediante registro en la propia aplicación, si accedió vía redes sociales debe acceder a las mismas para editar su perfil.

También permite la opción de darse de baja de la aplicación. Ello conlleva el borrado de todas sus recetas de la bbdd.

## 1.7. Diseño de la base de datos

El modelo propuesto para esta primera versión atendiendo a los requisitos de la aplicación, es el siguiente:

- **Usuarios (User):** Se trata del modelo base y del que depende la gestión del resto. Sus campos son: *username*, *email*, *password* y *network*. Siendo *username* y *email* sus claves primarias. Además debe realizar todas las acciones del CRUD, acrónimo de **C**rear, **O**btener, **A**ctualizar y **B**orrar (Create, Read, Update and Delete), lo que implica un cambio con las clases de los modelos con las que está relacionada. Estas relaciones son: un usuario puede tener varias recetas (clase *Recipe*) o varias recetas básicas (clase *Recipe2*).

Figura 1.4: Modelo usuario

- **Recetas (Recipe):** Este modelo contiene información sobre la receta creada por un determinado usuario. Establece la relación con la clase *Usuario*, del que depende de uno a muchos (1:M) y con las clases *ingredientes* y *recetas base*, cuya relación es en ambos casos de 1:M. Tiene muchos campos, de los que cabe destacar: *name*, nombre de la receta y clave primaria; *nrption*, campo requerido y que indica el número de platos, y *username* que es el autor de la receta.

Figura 1.5: Modelo receta

- **Ingredientes (Ingredient):** Se trata de los ingredientes para cada receta. Por lo tanto depende de la clase receta con una relación de 1:M. Sus campos más importantes son su *id* (clave primaria), *name* que es el nombre del ingrediente y *cost* que corresponde a su coste, siendo estos dos último campos requeridos.

Figura 1.6: Modelo ingrediente

- **Recetas base (Recipe2):** Esta clase es una extensión de recipe, que sirve para identificar las recetas usadas como ingredientes en otras, esto es una receta base. Su relación con la clase receta se presenta de 1:M.



Figura 1.7: Recetas base

Puede ver con detalle el código fuente del modelo de la aplicación en `model.rb`.