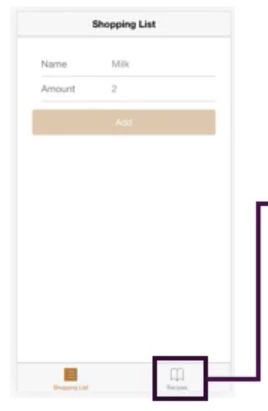
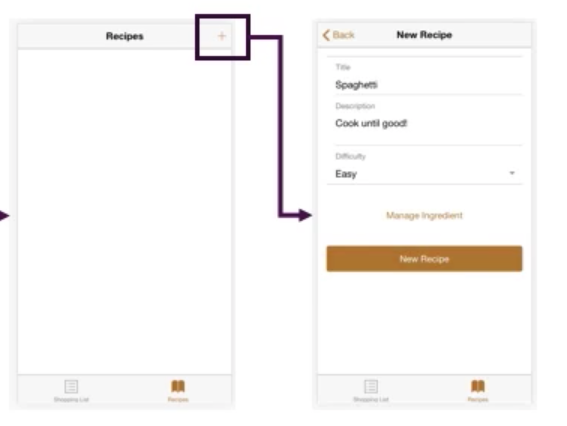
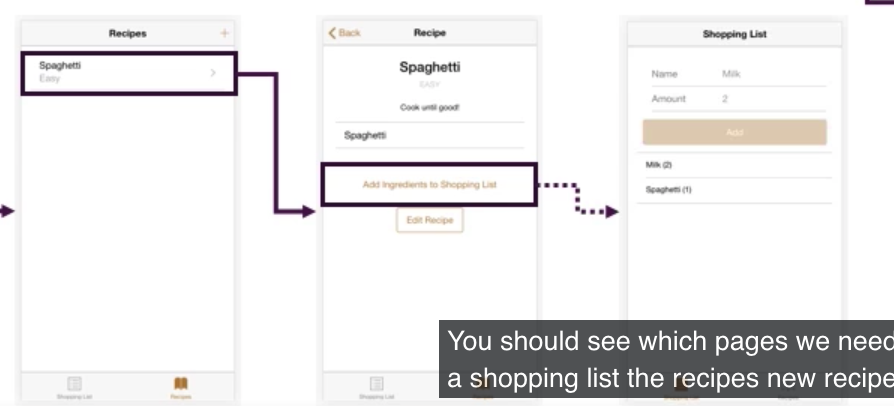
La pantalla principal contiene un input para agregar a la lista de compras:



Luego esta la pagina de recetas que se muestra en blanco inicialmente y luego se pueden agregar recetas al presionar el +, que despliega la plagina de añadir nueva receta con sus inputs.







Crear el nuevo proyecto y sus paginas con tabs:

Ionig g page …

Shopping List

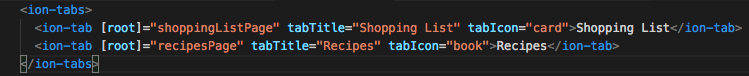
Recipes

Recipe

Tabs

Edit-recipes

Se crea la pagina tabas que va a ser la “inicial” que tendrá los tabs en la parte de abajo para desplazar las paginas que serán desplegadas en su parte superior.

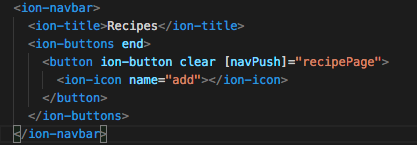


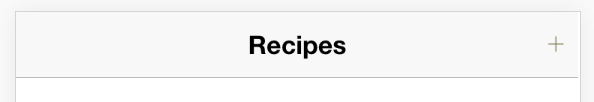


Se agregan los componentes en el app.module para poder utilizarlos.

Añadiendo el icono de añadir receta:

En la pagina de recetas añadir otro elemento al navBar

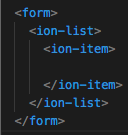


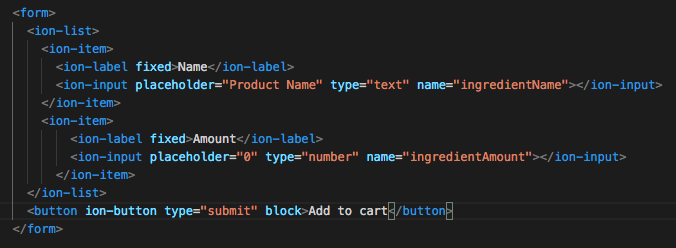


Lista la navegación, ahora se crea el contenido de las paginas.

Creando el formulario:

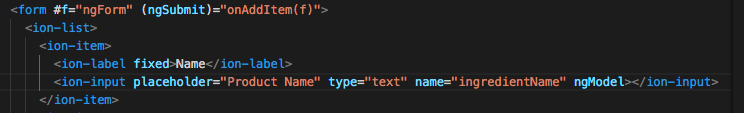
Se pueden crear los formularios ya sea dentro de un ion-list -> ion-item, o tenerlo dentro de un grid con cols.





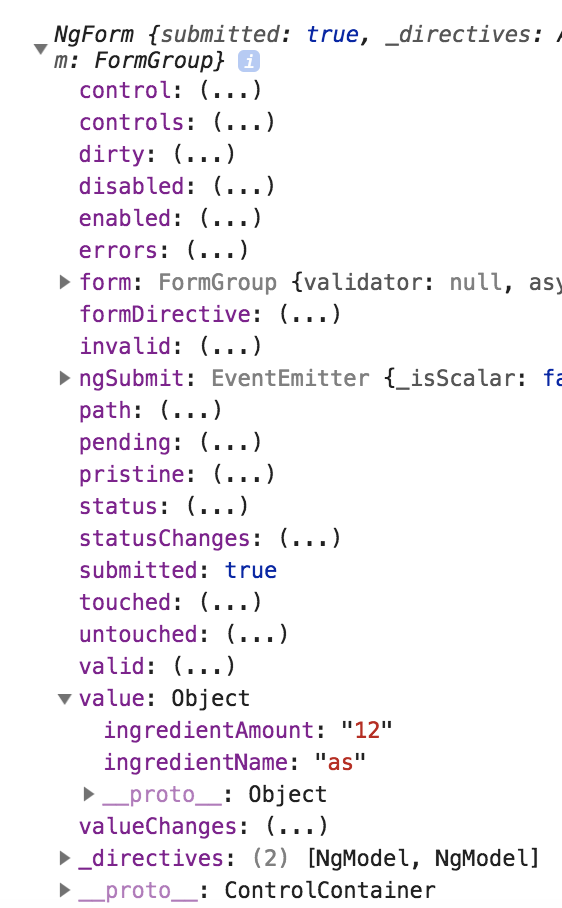
Para asignar controles a los inputs se agrega la directiva ngModules y asegurarse que cada campo tenga su nombre propio.

Se puede vincular un formulario añadiendo un #f y asignándole el valor de ngForm, para luego cuando se detecte el submit, llamar a la función necesaria pasándole los datos del form con el valor asociado (#f).



Ahora en el compontent.ts se crea la función que recibe como parámetro un formulario

Y accediendo a la propiedad value se observan los datos.

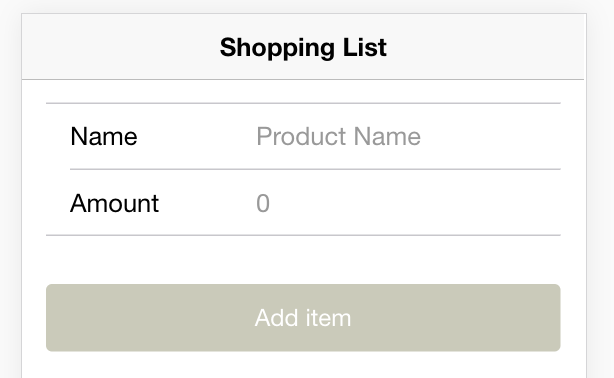
 

Validando los datos para que los campos sean necesarios. Y de uno no ser valido no habilitar el boton de add.

Al poner ejemplo requerido el input, se puede validar desde el boton y desabilitarlo si: la variable que se asigna al formulario es invalida.





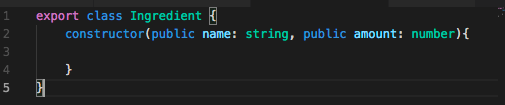


Ahora manejando los datos:

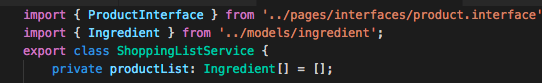
Se necesita un servicio para almacenar.

Y un modulo para tener el modelo de como se ve un ingrediente.

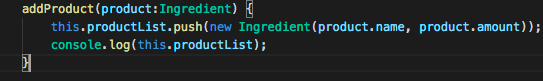




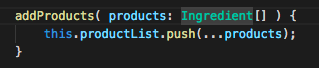
En el servicio se tiene el arreglo donde se guardaran los datos



Para añadir un elemento o varios se crean 2 metodos y con el constructor antes de añadir el elemento al arreglo se crea un nuevo ingrediente con el modelo creado, pasándole los datos necesarios que necesita su constructor.



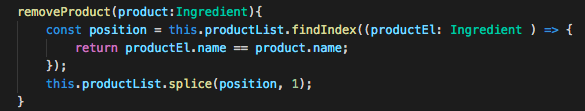
Como existe la posibilidad de recibir varios ingredientes (una receta), se crea el método aparte donde se recibe una lista de ingredientes para agregar a la lista, individualmente, osea cada uno con posición en el arreglo. Para esto es necesario dividir el arreglo que se recibe en elementos individuales. Se utiliza el operador spread (…),



Para obtener el arreglo (una copia para asi no tener una referencia directa ).



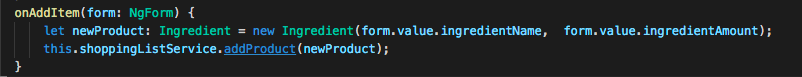
y para remover un producto recibido donde hace una búsqueda que recorre el arreglo y compara con la propiedad name del producto recibido y retorna su posición. Para luego borrarlo con el método splice.



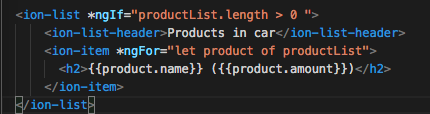
Añadiendo la funcionalidad al botón submit

En la función llamada por el submit

Se declara una variable del timpo ingrediente que tendrá un nuevo ingrediente creado por el constructor de esa clase, pasándole los parámetros recibidos por el form.value y luego llamando a la función del servicio pasándole la variable.

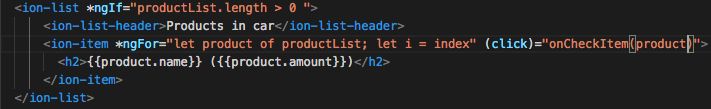


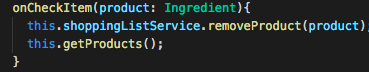
Mostrando la lista de productos ya agregados en el html.



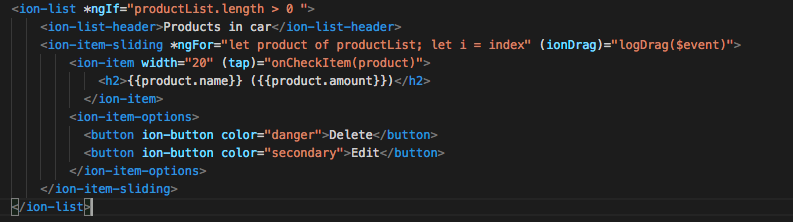


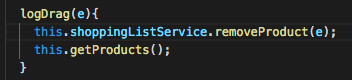
Borrando un producto al hacer click en el y mostrando la opción.





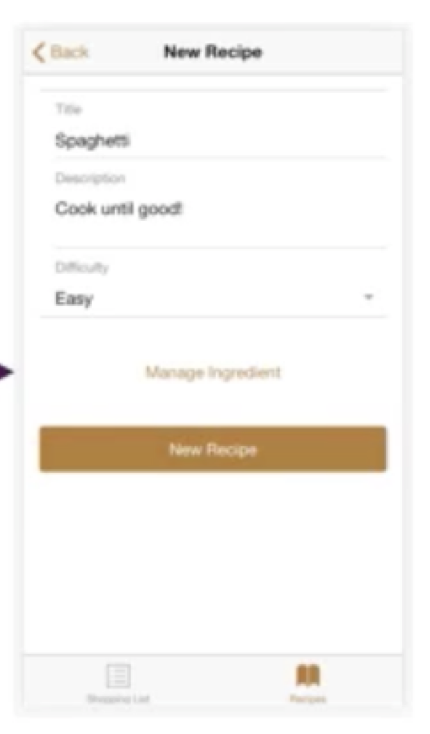
Añadiendo la funcionalidad de slide, borrar al hacer slide y mostrar botones:







Haciendo el formulario para agregar nueva receta (recipe)

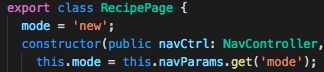


Recuperando el parámetro pasado de la pagina anterior.

Pag anterior:



recibiendo:



En el html:



y luego crear un dropdown para la dificultad:

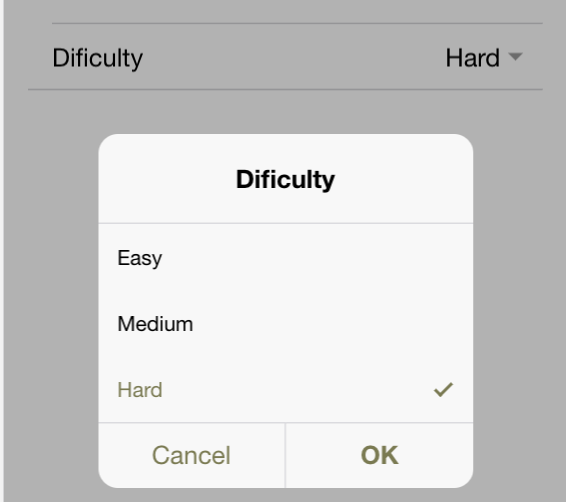
html, en donde es importante el [value]



ts



se carga el ion-select, y este se va a iterar según el contenido del arreglo que se crea.



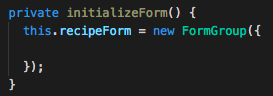
Creating the reactive form.

Queremos atar el formulario creado en el html con el que se creara programáticamente en el ts.

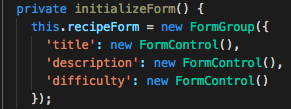
Se declara una variable recipeForm de tipo FormGroup



Para luego inicializarlo en el método que se crea. Este se hace de forma que el constructor de esta clase reciba un objeto.

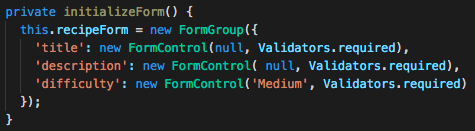


luego en el objeto se inicial los datos como se declaro el formulario declarándolas de tipo FormGroup



Ahora dentro de el constructor de las variables se setean los valores requeridos.

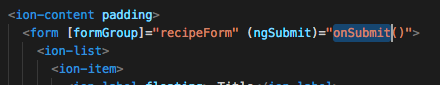
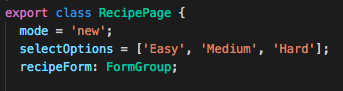
El primer valor es el por defecto, de querer dejarlo vacio se pone null, lo contrario se asigna un valor y esta función llamarla dentro del constructor.



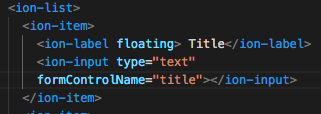
Una vez programado hay que conectarlo con el código html.

Para esto en el html al inicio del formulario se pone el tag formGroup ya declarado en el ts.

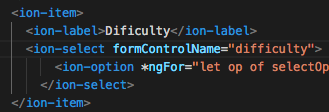
html

 ts

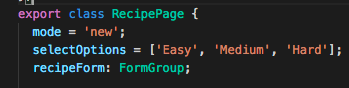
Ahora dentro de cada tag de input se debe agregar el controlador con su nombre para unirlos.



Para el select se pone el tag dentro del select, no en option.



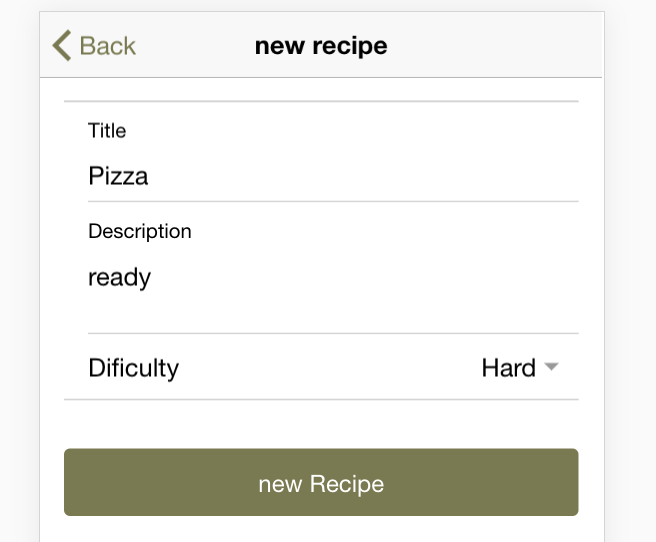
al declarar recipeForm del tipo FormGroup y vincularlo al form y sus validaciones:



Se puede validar el botón pueda presionarse o no con la propiedad valid de este form



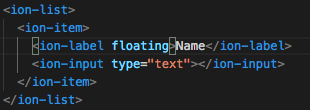
Añadir ingredientes a una receta:



se añade un botón donde al presionar se va agregar un nuevo ingrediente.



que se mostrara en una lista :



Creando un action sheet (Documentacion de ionic):

Implemetar el controlador del ActionSheet en el constructor y luego almacenarlo en una variable.



luego dentro esto recibe como parámetros un objeto con una estructura de botones y sus funciones dentro de cada handler

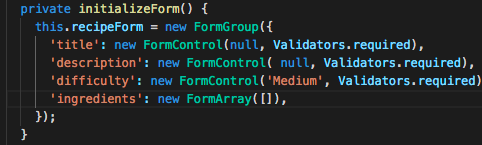


Mostrando un alert con input al añadir ingrediente:

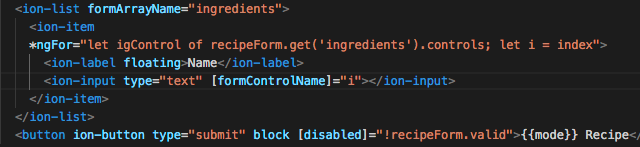
Se crea una función aparte que muestre el modal.



y se agrega un nuevo form control

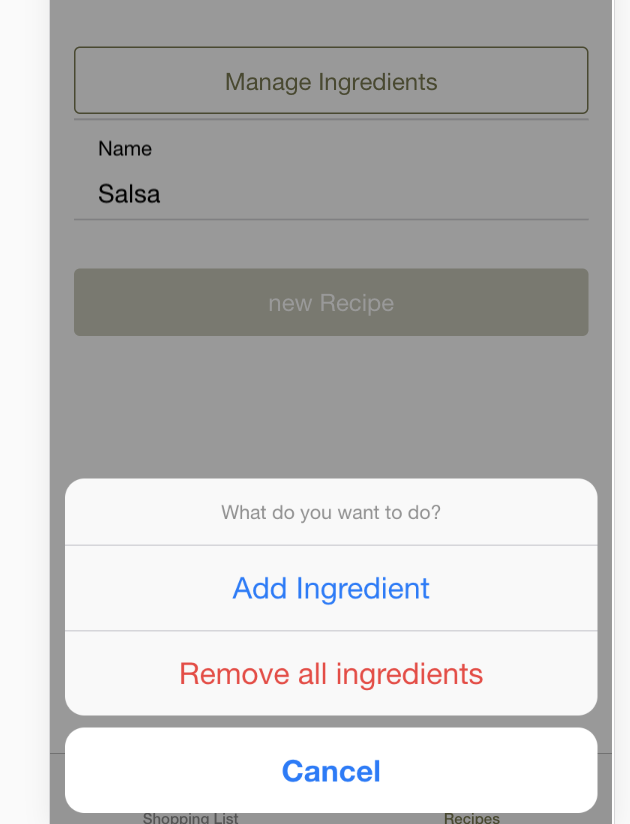


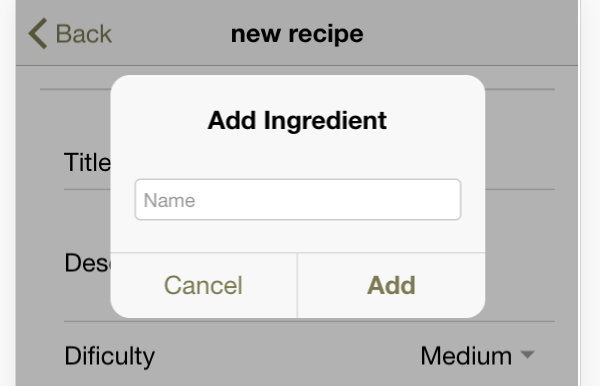
En el formulario nuevo se conecta de la siguiente forma:



Añadiendo una nueva validación a las ya creadas, como se creo una lamada ingredients de tipo arreglo se puede acceder a ella de la siguiente forma:

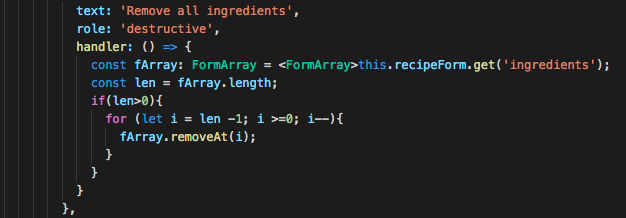




Removing Ingredient Control

Se accede al arreglo de ingredientes



Mostrando un toast para informar al usuario cuando sucede o no algo.

Se puede modificar su duración y la posicion



Añadiendo un récipe model y un servicio.

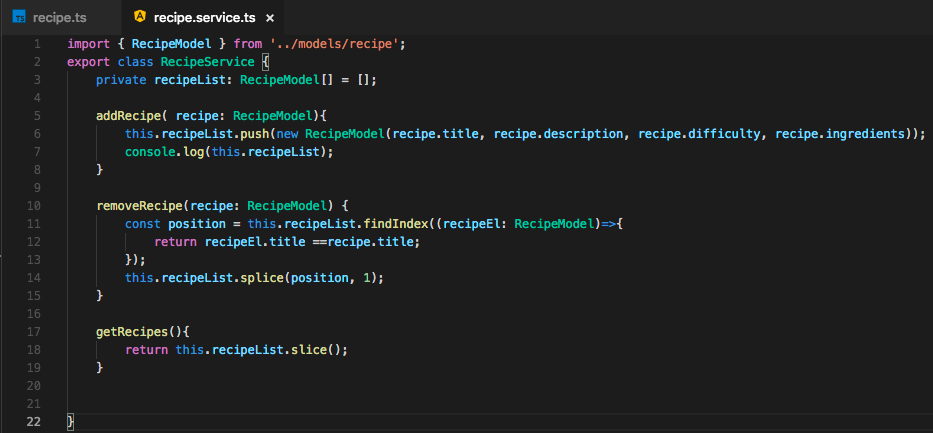
Las recetas tienen el comportamiento de los ingredientes, solo que estas tienen lista de ellos también,

El servicio tendrá un arreglo de este tipo con los métodos para agregar, eliminar, y traer las recetas.

Paso 1 creando el modelo:



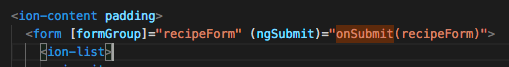
El servicio donde se crea un arreglo del tipo del modelo, y se crean los métodos en donde sea grega, utilizando el constructor del RecipeModel.



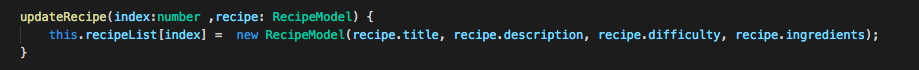
Que es llamado desde el component.ts, utilizando los datos del objeto value retornado desde el html.



html



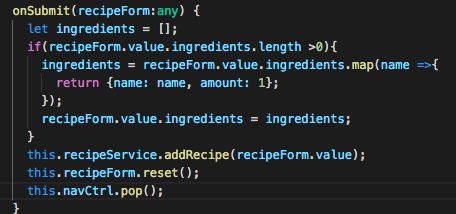
En el servicio también esta el método de editar una receta: que recibe el objeto y el índice para buscarlo en el arreglo de recetas.



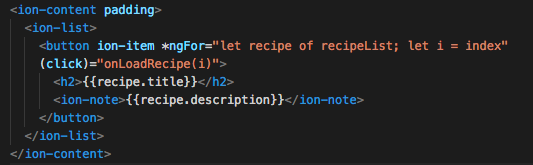
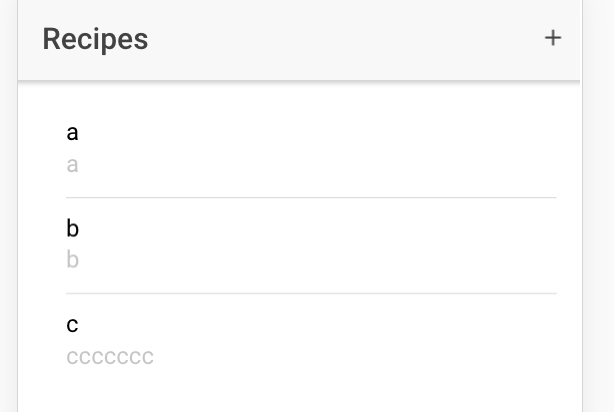
Mapeando al agregar ingredientes

Validando la exitencia de almenos un ingrediente en la receta, al tener el modelo de ingrediente los campos nombre y amount, hay que asignarle, asi que se valida si existe y se asigna valor al arreglo local de ingredientes como un objeto y luego se le asigna al la propiedad ingredients del objeto a mandar.

y luego se desea regresar a la pagina root de las recetas:

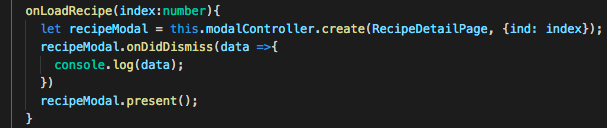


En la pagina de recetas integramos el método a llamar pasándole el índice del ítem a seleccionar.

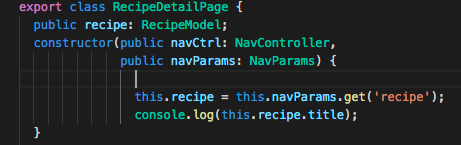


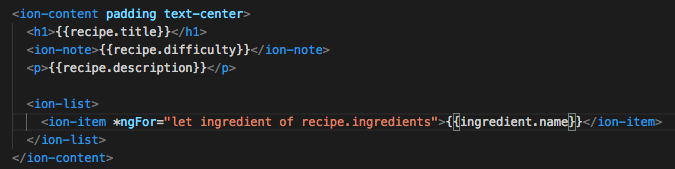
Moviendose a la pagina de detalles de la receta.

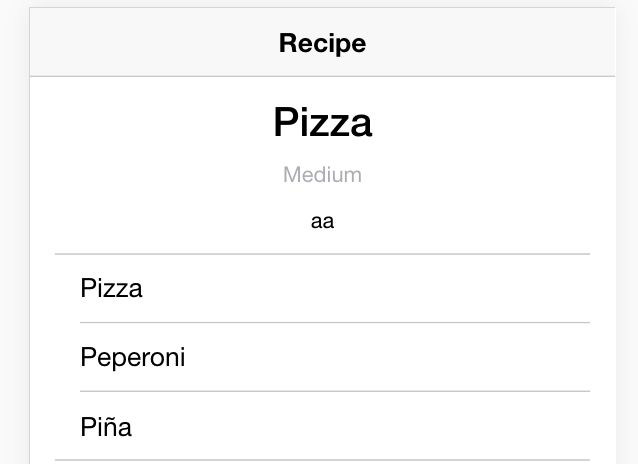
Se mostara como modal , para esto desde recipes:



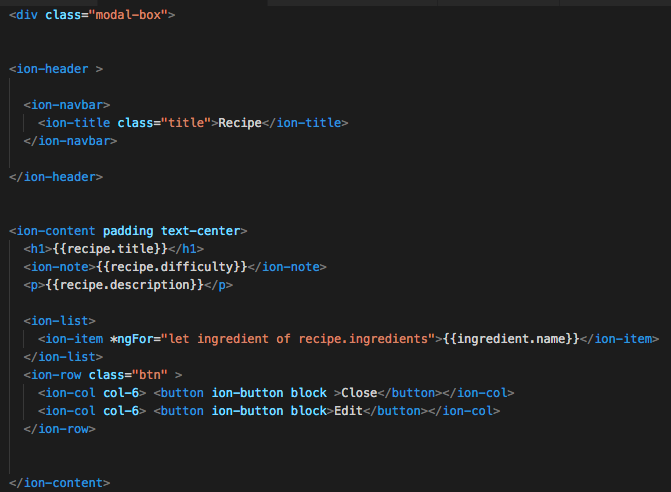
y desde la pagina de detalles recibiendo parámetros

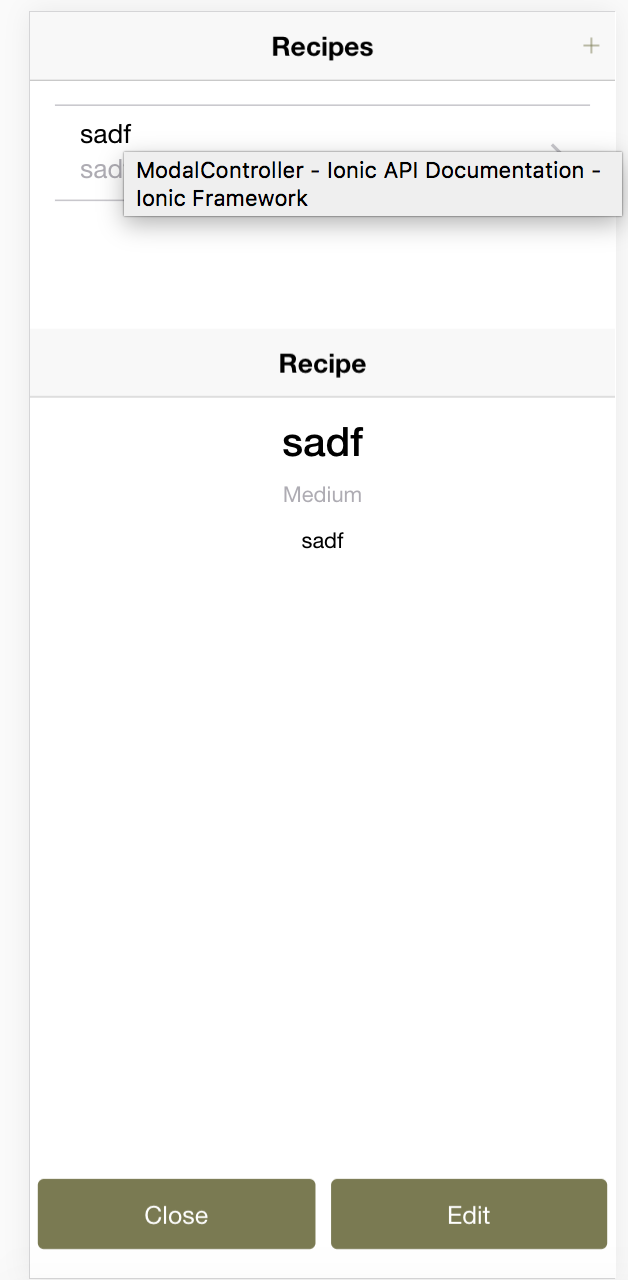




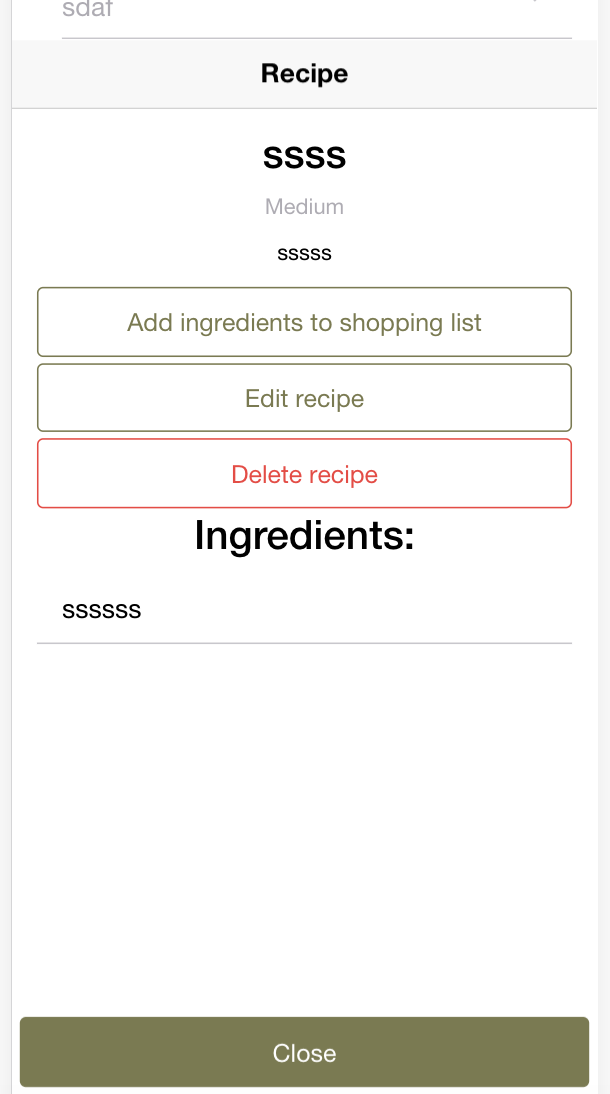


Haciendo una pequeña modificación para que el modal no se presente en toda la pantalla y tenga 2 botones abajo:

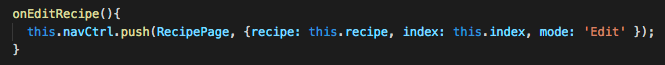




Propuesta final de la vista de receta



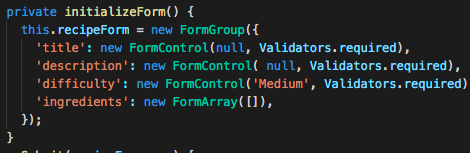
Al presionar editar hay que pasar a la pagina recipe los datos del recipe, index, y el modo que es edición



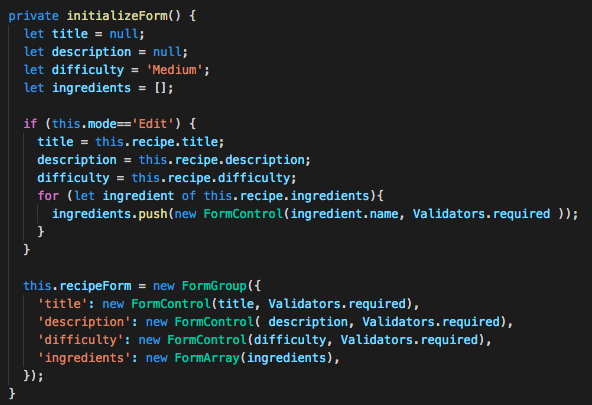
Luego en la pagina de recipe, que se va a mostrar ya sea para agregar una nueva receta o para editar, va a depender del modo recibido para inicializar el formulario.

En esta función recordando que el primer valor en los parámetros se seteo null, o un valor por defecto, este es el parámetro que queremos cambiar.

Pasando de:

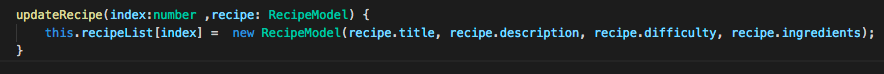


A:

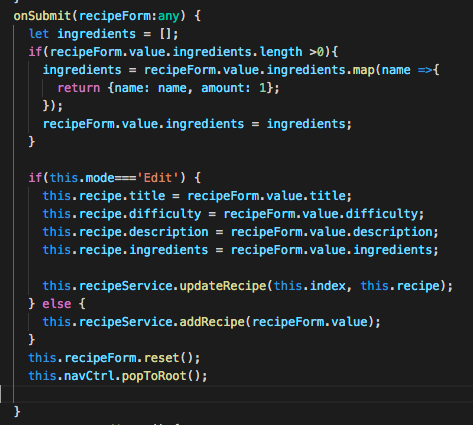


Tomando en cuenta que para los ingredientes, se espera un form control, asi que el arreglo local del método se llena con un form control de cada ingrediente.

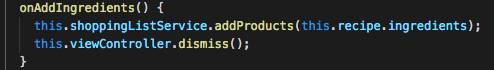
Para editarlo se hace desde el método del servicio.



y en el submit se modifica para que el recipe reciba los nuevos datos que se entraron en los campos

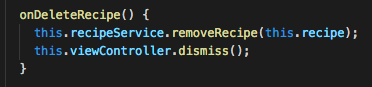


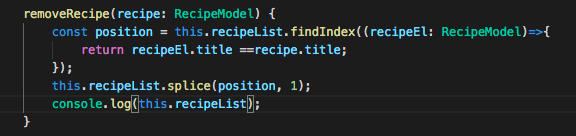
Luego para añadir los ingredientes que se encuentran en una receta:





Para borrar:





Y como el récipes-detail es un modal, al cerrarse se debe estar a la escucha para cargar nuevamente la lista de recetas:

