**Recetas - application documentation**

[**Requirements**](http://en.wikipedia.org/wiki/Requirement) - Identify attributes, capabilities, characteristics and qualities of the system.

**Architecture/Design** - Overview of the systems, including relations to an environment and construction principles.

**Technical** - Documentation of code, algorithms, interfaces, and [APIs](http://en.wikipedia.org/wiki/API_documentation).

[**End user**](http://en.wikipedia.org/wiki/End_user) - Manuals for the end-user, system administrators and support staff.

**Marketing** - How to market the product and analysis of the market demand.

**Requirements**

**Use Cases:**

1. Home page.

Main menu.

Includes a list of most recent recipes, with links.

1. Login page.

Automatic for access to every other page except Home. (TO DO)

Include 'remember-me'.

1. View recipe page.

View the recipe information. (TO DO: wikipedia)

List of ingredients with links.

1. View ingredient page.

Includes nutrition facts.

Takes information from FatSecret (tm)

List of recipes that use it, with links.

1. Recipes management main (menu) page.

List of recipes.

Options: CRUD.

1. Delete recipe confirmation page.
2. Recipe edit page.

For modification and new recipes.

Include ingredient search. Different to the Edit page? (TO DO)

1. Search recipes page.

Different criteria: name (like), max calories, use of ingredients...

List of results with links.

1. Search ingredient page.

Different criteria ? (TO DO)

List of recipes that use the ingredient, with links.

**Architecture/Design**

**Constraints.**

* Use FatSecret database.
* Programming: java 7 + Spring 4

**Entities**.

1. User.

Registered users.

1. Recipe.

Belongs to one User.

1. Ingredient.

Belongs to one Recipe.

Links to one Food object (from FatSecret database).