|  |  |
| --- | --- |
|  | **UNIVERSIDAD**  **REY JUAN CARLOS** |

**ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA**

**INGENIERÍA INFORMÁTICA**

**Curso Académico 2014 / 2015**

**Proyecto de Fin de Carrera**

**Desarrollo de una aplicación web basada en conocimientos de la web semántica sobre recetas**

**Autor:** Karen Wiznia Fresco

**Tutores:** Alberto Fernández

**Resumen**

**Redactar el resumen**

**sacar los pantallazos de la bbdd**

**Explicacion de la BBdd**

**Mejorar conclusiones**

**añadri apendice al final**

**AGRADECIMIENTOS**

El presente proyecto supone el final de un camino y el comienzo de una nueva etapa en mi vida. Por eso, antes de dar un paso más quería agradecer y recordar a todas aquellas personas que me acompañaron durante estos últimos cuatro años de mi vida.

En primer lugar quería agradecer a mi tutor, Alberto Fernández, por toda la ayuda prestada y por estar siempre ahí, incluso en momentos donde sentía que no avanzaba y que me paralizaba. No lo hubiese podido hacer sin esos empujones, gracias.

Por otro lado quería agradecer a todos mis profesores por inspirarme cada día. Durante las clases, o esas charlas después de clase de “cómo puedo mejorar para que haya menos suspensos” hacen que yo cada día me pregunte como puedo mejorar. Ante todo quería agradecer a Antonio Montemayor, Isaac Martin, Soto Montalvo y Roberto Muñoz. Gracias por acompañarme todos estos años y formarme tanto académicamente como profesionalmente.

También quiero agradecer a mis compañeros de clase, sobre todo a Miguel Cortecero, por acompañarme cada día en clase, ayudarme en momentos delicados y enseñarme que siempre hay que intentarlo hasta el final, que el mayor fracaso es no intentarlo.

Finalmente quería agradecer a mis padres y a mis hermanos. Gracias por apoyarme y aguantarme todos estos cuatro años. Gracias a mis hermanos por ayudarme a pensar mejoras para el proyecto y por guiarme sobre todo en la búsqueda de tutor y tecnologías a usar para poder empezar. Gracias por confiar en mí.

Muchas gracias a todos, y ante todo, me gustaría recordar que las cosas tarde o temprano salen. Lo importante es intentarlo y no abandonar hasta que salga. El peor fracaso es no intentarlo.

# Índice

[Índice 4](#_Toc416096692)

[Capítulo 1: Introducción 6](#_Toc416096693)

[1.1 Presentación del Problema 7](#_Toc416096694)

[1.2 Objetivos 9](#_Toc416096695)

[1.3 Método de Trabajo 10](#_Toc416096696)

[1.4 Medios Hardware y Software 12](#_Toc416096697)

[1.5 Estructura de la Memoria 13](#_Toc416096698)

[Capítulo 2: Estudios Previos 15](#_Toc416096699)

[1.6 Estudios previos 16](#_Toc416096700)

[Capítulo 3: Desarrollo del Caso de Estudio 19](#_Toc416096701)

[1.7 Especificación de Requisitos 20](#_Toc416096702)

[1.8 Modelo de casos de uso 21](#_Toc416096703)

[1.9 Flujos de eventos 23](#_Toc416096704)

[Capítulo 4: Diseño de la descripción informática 29](#_Toc416096705)

[1.10 Diseño 30](#_Toc416096706)

[4.1.1 Arquitectura 30](#_Toc416096707)

[4.1.2 Diseño de la ontología 31](#_Toc416096708)

[4.1.3 Diseño de la interfaz 32](#_Toc416096709)

[4.1.5 Diseño del módulo de control 47](#_Toc416096710)

[Capítulo 5: Conclusiones y trabajos futuros 52](#_Toc416096711)

[1.11 Conclusiones: 53](#_Toc416096712)

[1.12 Estimación del esfuerzo 54](#_Toc416096713)

[1.13 Mejoras futuras 56](#_Toc416096714)

[Capítulo 6: Bibliografía y lugares de internet 57](#_Toc416096715)

[1.14 Páginas cuya funcionalidad es parecida 57](#_Toc416096716)

[1.15 Páginas sobre SPARQL, RDF y Jena 57](#_Toc416096717)

[1.16 Páginas sobre NetBeans 58](#_Toc416096718)

[1.17 Páginas sobre Protégé 58](#_Toc416096719)

[1.18 Páginas sobre HTML y CSS 58](#_Toc416096720)

[1.19 Otras páginas de interés 59](#_Toc416096721)

# Capítulo 1: Introducción

En este capítulo se presenta el planteamiento y la justificación de este trabajo, una breve explicación de los objetivos planteados, su ejecución en fases, los resultados esperados, los medios hardware y software necesarios y el método de trabajo aplicado. Por último, se describe la organización de esta memoria de Proyecto de Fin de Carrera.

## Presentación del Problema

En este Proyecto Fin de Carrera (PFC) se va a desarrollar una aplicación web para la búsqueda de recetas.

Para la realización del algoritmo se utiliza RDF[1], que es un estándar en W3C[2] para codificar el conocimiento. Estos estándares facilitan la construcción de aplicaciones en sistemas descentralizados, es decir en bases de datos distribuidas [3]. Dicho esto, cabe destacar que hay una serie de aspectos a considerar al respecto, entre ellos se encuentran:

1. Los archivos necesitan poder expresar la información de manera flexible. Esto, en vez de hacerlo con tablas y relaciones entre sí, se hace con un grafo ya que da más flexibilidad. Como es lógico no es posible dibujar un grafo en la web, con lo cual esto se representa con una notación tabular para grafos.
2. Los archivos de la web semántica [4] necesitan una manera de relacionarse.

La cantidad de información que se maneja hoy en internet es abrumadora, tanto es así que hay veces que la información nos abruma y no somos capaces de encontrar con exactitud qué es lo que queremos conseguir, ahogándonos en páginas y páginas que ni si quiera se acercan a la solución del problema. Es posible que nos enfrentemos con este problema día tras día, y que no seamos capaces de detectarlo.

Con este proyecto se intenta facilitar el trabajo a los usuarios mediante un algoritmo que busque exactamente lo que ellos quieran. En este caso, concretamente el usuario dispone de una cantidad de ingredientes y quiere saber qué posibles recetas puede realizar con los ingredientes que tiene. La idea del algoritmo a desarrollar es que el usuario sea capaz, mediante una interfaz web amigable y fácil, de introducir tantos ingredientes como gusten, y la aplicación simplemente le devolverá todas las recetas que puede realizar con los ingredientes introducidos.

Además no se pretende que el usuario vaya a introducir siempre una gran cantidad de ingredientes, ya que este puede introducir solamente uno, lo cual la búsqueda se vería demasiado acotada. La solución a esto, es que, una vez haya sacado las recetas que coinciden justamente con los ingredientes introducidos por el usuario, también saque aquellas recetas que contengan esos ingredientes y uno menos, y dos menos, etc. E ir reduciendo la cantidad de ingredientes progresivamente.

Además también podría ser interesante que le del usuario la posibilidad de seleccionar otras recetas, a pesar de no tener todos los ingredientes para realizarlo, es por ello que una vez que saque todas las recetas que coincidan con los requisitos anteriores, también sacará las recetas que contengan esos ingredientes y uno más, y dos más, etc. E irá aumentando progresivamente la cantidad de ingredientes.

## Objetivos

El objetivo principal de este proyecto de fin de carrera es permitir a los usuarios la búsqueda de recetas a partir de unos ingredientes dados.

Con el fin de alcanzar este objetivo, se han definido los siguientes objetivos parciales:

* Estudiar a fondo las páginas ya existentes.
* Decidir qué tecnologías se van a utilizar para la realización de la aplicación web.
* Identificar mejoras (tanto a nivel estructural como de la interfaz).
* Implantar las mejoras en las funcionalidades que se muestran a continuación:
* **Recetas con ingredientes exactos.** El usuario podrá introducir los ingredientes que considere necesarios y se mostrará las recetas que contengan únicamente esos ingredientes.
* **Recetas con esos ingredientes y algunos menos.** El usuario podrá introducir los ingredientes que considere necesarios y se mostrará las recetas que contengan esos ingredientes y uno menos, y dos menos, etc. Hasta que solo tenga un único ingrediente.
* **Recetas con esos ingredientes y algunos más.** El usuario podrá introducir los ingredientes que considere necesarios y se mostrará las recetas que contengan esos ingredientes y uno más, y dos más, etc. Hasta que no existan más recetas que contengan esos ingredientes y más.
* **Recetas cuya dificultad es baja, media o alta.** El usuario podrá seleccionar el grado de dificultad de las recetas que quiere que le aparezcan. Puede seleccionar las recetas con dificultad baja, media, alta o también combinaciones como obtener recetas con dificultad baja y media, evitando las de dificultad alta.
* **Recetas según las calorías de la misma.** El usuario puede introducir la cantidad de calorías máximas que desea que tenga las recetas seleccionadas.
* **Recetas según el tiempo de preparación.** El usuario puede introducir un tiempo máximo, con el fin de que devuelva las recetas cuyo tiempo máximo es el indicado.
* **Sugerencias.** El usuario podrá enviar su sugerencia a través de un formulario en la página web que permita incluir mejoras y nuevas funcionalidades.

## Método de Trabajo

El método de trabajo seguido en la realización de este PFC está constituido por diferentes fases como se puede ver en las figuras 1 y 2. Paralelamente a las mismas se ha llevado a cabo la elaboración de la memoria de este trabajo. A continuación, se van a describir las tareas a realizar en cada una de las fases del desarrollo del presente proyecto fin de carrera.

**FASE 1. ESTUDIOS PREVIOS:** En primer lugar, se realizará el estudio de las diferentes páginas que hacen cosas parecidas. Durante esta fase también se analizara los fallos y carencias de cada una de ellas con el objetivo de esbozar nuevas ideas para mejorarlo y añadir posibles funcionalidades que completen el sistema.

**FASE 2. ESTUDIO DE LAS TECNOLOGIAS A UTILIZAR:** Una vez realizada la fase uno se procederá al estudio de las tecnologías a utilizar. Para la realización de este proyecto he de familiarizarme con tecnologías como Protégé [5], SPARQL [6], Jena [7] y Java [8]. En cuanto a la interfaz del portal web, se va a desarrollar mediante la utilización de lenguajes como HTML y CSS.

**FASE 3.IMPLEMENTACION DEL ALGORITMO DE BUSQUEDA:** Durante esta fase se realizará la implementación del algoritmo de búsqueda que permitirá la búsqueda de las diferentes recetas.

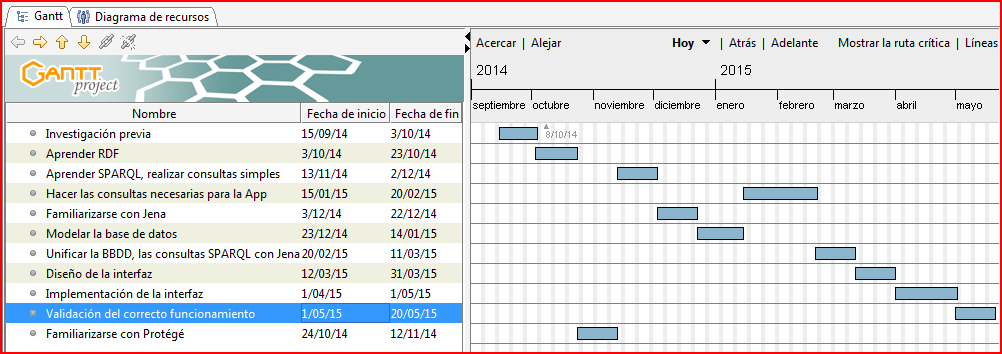
**FASE 4: IMPLEMENTACION DE LA INTERFAZ DE USUARIO:** Durante esta fase se diseñará y realizará la interfaz de usuario.

**FASE 5.VALIDACION:** Den esta última etapa del proyecto se pretende realizar pruebas que comprueben múltiples escenarios en los que el portal web devolverá desenvolverse sin problemas, con el fin de encontrar y solucionar posibles fallos. Además, en la página web, habrá un apartado donde el usuario puede leer como utilizar la aplicación para obtener mejores resultados.

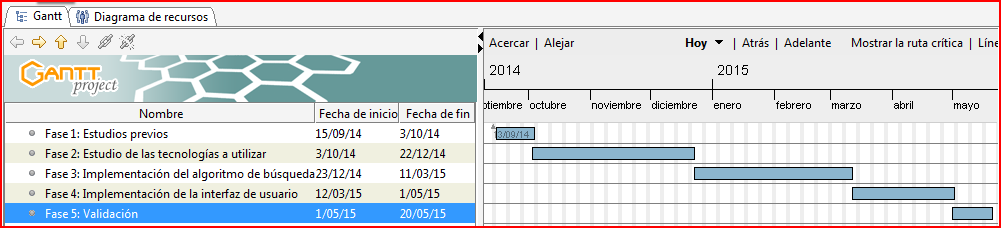
## Medios Hardware y Software

Este proyecto se ha realizado dentro del laboratorio del grupo de investigación de Inteligencia Artificial, haciendo uso de los siguientes medios hardware y software:

* **Medios Hardware:**
  + Ordenador Portatil Acer I7 4500U con 8GB de RAM.
* **Medios o herramientas software:** 
  + Microsoft Windows 8: sistema operativo empleado como plataforma de trabajo.
  + Microsoft Word: procesador de textos utilizado para la realización de toda la documentación de este proyecto.
  + Protégé: Es un editor de código abierto que se utiliza para la construcción de oncologías y la representación del conocimiento.
  + Netbeans 8.0.1: Entorno de desarrollo integrado, que permite entre otras cosas, la creación de aplicaciones web con Java.
  + GanttProject 2.7: software libre utilizado para la administración de proyectos usando diagramas de Gantt.
  + Balsamiq Mockup: Aplicación o servicio que permite al desarrollador la creación de bocetos y esquemas de la interfaz web.
  + GitHub: Repositorio de código donde se fue dejando el código.
  + ArgoUML: Programa para realizar los diferentes diagramas y modelos.



1. Ilustración



1. Ilustración

## Estructura de la Memoria

Este **primer capítulo** es una introducción al PFC que incluye una presentación del problema que se va a desarrollar, los objetivos que se pretenden alcanzar, el método de trabajo seguido y la descripción de los medios software y hardware que se han usado.

En el **segundo capítulo** se hace una descripción de los estudios previos necesarios para la realización de este trabajo. Es decir, que es lo que hay hecho ya en internet. Luego de realizar la comparativa de lo que hay en internet y lo que será mi proyecto se expone el modelo de casos de uso y después cada caso de uso puntual, es decir que hace en cada momento la aplicación.

El **tercer capítulo** se centra en los requisitos candidatos para la aplicación.

En el **cuarto capítulo** se exponen los casos de uso de la aplicación. Es decir mostrar a través de pantallazos de la aplicación que puede hacer el usuario, y que resultados obtiene en cada movimiento que realiza el usuario.

En el **quinto capítulo** se exponen las conclusiones y trabajos futuros.

También se incluye la bibliografía utilizada y los lugares de Internet visitados.

Finalmente, se incluye al final de esta documentación un apéndice que muestra las siglas y abreviaciones utilizadas en todo el documento.

# Capítulo 2: Estudios Previos

Este capítulo se puede estudiar desde dos puntos de vista. Por un lado se hace una breve explicación sobre aplicaciones cuya funcionalidad es parecida a la que se expone aquí. Por otro lado, a su vez, pretende proporcionar una visión general del marco en el que se desarrolla este proyecto y sus aportaciones al mismo.

## Estudios previos

En este apartado se va a describir cuales son las aplicaciones desarrolladas cuya funcionalidad intentan que sea la explicada en este proyecto. Una de las páginas más conocidas cuya funcionalidad es la de sacar las recetas que contienen únicamente esos ingredientes es conocida como *que hay en la nevera*. Esta pretende que el usuario introduzca cuantos ingredientes desea que tenga las recetas, y en relación a los ingredientes introducidos, la aplicación debe sacar las recetas que contienen únicamente esos ingredientes. El problema con esta solución es que es indiferente qué o cuántos ingredientes introduzca el usuario, está siempre devolverá las recetas que contengan al menos esos ingredientes sin distinción.

Otra página cuya funcionalidad es parecida es la llamada *mis-recetas*. En esta te permite introducir únicamente tres ingredientes que quieras que estén contenidos en las recetas y tres ingredientes que no quieras que tenga. A continuación te devuelve todas las recetas que contienen esos ingredientes y más. Tiene numerosos filtros por los que restringir como estilo de cocina, vegetariano, país, entre otros.

Otra página cuyas circunstancias son similares a las otras dos antes nombradas es *todareceta* en la cual te permite introducir únicamente un ingrediente y te busca todas las recetas que contienen ese ingrediente y más.

Además hay otras aplicaciones como *Rebanando*o *quierucomer* las cuales intentan realizar esta funcionalidad pero tanto en una como en otra, es indiferente los ingredientes que introduzca el usuario, esta devolverá recetas no acordes con la búsqueda seleccionada.

Ante esto a continuación se presenta una tabla comparativa entre la funcionalidad existente entre algunas de las aplicaciones antes nombradas y la que se lleva a cabo en este proyecto.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Permite introducir más de un ingrediente | Permite introducir más de tres ingredientes | Búsqueda por calorías | Búsqueda por tiempo de preparación | Búsqueda por dificultad | Devolución de recetas que contienen únicamente esos ingredientes | Devolución de recetas que contienen esos ingredientes y menos | Devolución de recetas que contienen esos ingredientes y más |
| quehayenlanevera |  |  |  |  |  |  |  |  |
| mis-recetas |  |  |  |  |  |  |  |  |
| todareceta |  |  |  |  |  |  |  |  |
| Rebanando |  |  |  |  |  |  |  |  |
| Quierucomer |  |  |  |  |  |  |  |  |
| Mi aplicación |  |  |  |  |  |  |  |  |

1. Ilustración

# Capítulo 3: Desarrollo del Caso de Estudio

## Especificación de Requisitos

A través de este apartado, se va a definir la especificación de requisitos correspondiente al soporte técnico para la búsqueda de recetas.

**Propósito**

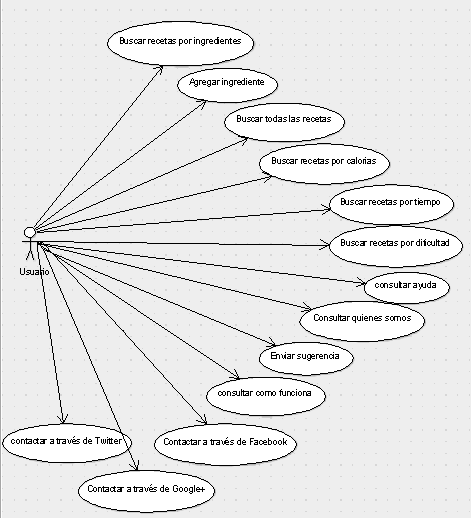
En el presente documento se aclararan y definirán los diferentes requisitos y restricciones que un usuario podrá realizar dentro de la aplicación web. Este estándar tiene como objetivo principal, exponer las funcionalidades que serán implementadas para la correcta realización del mismo.

**Requisitos**

* El sistema se ejecuta sobre un servidor HTTP.
* El usuario puede añadir cuantos ingredientes crea conveniente
* El usuario puede buscar filtrando por ciertos ingredientes introducidos anteriormente.
* El usuario puede buscar las recetas según su nivel de dificultad, mostrando solo las recetas fáciles.
* El usuario puede buscar las recetas según su nivel de dificultad, mostrando solo las recetas de dificultad media.
* El usuario puede buscar las recetas según su nivel de dificultad, mostrando solo las recetas difíciles.
* El usuario puede buscar las recetas según su nivel de dificultad, mostrando solo las recetas de dificultad fácil y media.
* El usuario puede buscar recetas según unas calorías máximas.
* El usuario puede buscar las recetas dado un tiempo máximo de realización.
* El usuario puede buscar todas las recetas que hay en la base de datos.
* El usuario puede enviar una sugerencia introduciendo su nombre, un correo electrónico y un comentario.
* El usuario puede acceder a la pestaña “quiénes somos”.
* El usuario puede acceder a la pestaña “sugerencias”.
* El usuario puede acceder a la pestaña “como funciona”.
* El usuario puede acceder a la pestaña “ayuda”.
* El usuario puede contactar con los administradores de la aplicación a través de redes sociales tales como Google+, Facebook o Twitter.

## Modelo de casos de uso

En este apartado se va a describir la aplicación mediante el uso de diagramas de casos de uso.

Caso de uso referido al usuario:

1. Ilustración

El diagrama de casos de uso representa todo lo que un usuario puede realizar en la aplicación. El usuario podrá agregar cuantos ingredientes desee para a continuación buscar recetas por ingredientes. Por otra parte puede acceder a todas las recetas que existen en la base de datos indistintamente. Por otra parte, el usuario es capaz de buscar recetas utilizando otros filtros. Entre estos el usuario puede buscar las recetas por tiempo de realización, es decir el usuario puede poner un máximo de tiempo para realizar la receta. También puede poner un límite en las calorías, poniendo un número máximo de calorías que desea que tenga las distintas recetas. Finalmente también puede consultar las recetas por el nivel de dificultad (medio bajo o alto). Además dentro de la página el usuario puede acceder a las diferentes pestañas mostradas en esta. El usuario puede consultar la ayuda en caso de tener alguna duda o consultar el apartado de cómo funciona para obtener una ayuda más detallada de cómo funciona la aplicación y que se puede hacer en ella. Además el usuario puede enviar una sugerencia al grupo de desarrolladores en caso de que desee sugerir una nueva funcionalidad o advertir un mal funcionamiento en la misma. Finalmente si estos medios no fueron suficientes para él, también puede contactar con los administradores de la página accediendo a través de redes sociales como Google+, Twitter o Facebook.

## Flujos de eventos

Para mayor claridad se describe a continuación los flujos de eventos de los casos de uso, en todos ellos se incluye los pasos necesarios para que el sistema valide al usuario:

1. Consultar recetas por ingredientes:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona buscar por ingredientes |  |
|  | 3.El sistema muestra la pantalla para buscar por ingredientes |
| 4.El usuario introduce un ingrediente |  |
| 5. El usuario presiona el botón “agregar otro ingrediente” | 6.El sistema devuelve una lista con los ingredientes que lleva introducidos |
| 7.El usuario presiona el botón de “buscar recetas” | 8. El sistema procesa la opción seleccionada y muestra todas las recetas que se corresponden con dichos ingredientes. |
| 9. El actor selecciona la receta que desea  visualizar | 10. El sistema procesa la receta seleccionada y muestra la receta con todas sus propiedades. |

Nota: El paso 4 y 5 se repiten tantas veces como ingredientes quiera introducir el usuario.

1. Consultar recetas por dificultad:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona buscar por dificultad |  |
|  | 3.El sistema muestra la pantalla para buscar por dificultad |
| 4.El usuario selecciona el nivel de dificultad que desea |  |
| 5.El usuario presiona el botón de “buscar recetas” |  |
|  | 6. El sistema procesa la opción seleccionada y muestra todas las recetas que se corresponden con dicho nivel de dificultad. |
| 9. El actor selecciona la receta que desea  visualizar | 10. El sistema procesa la receta seleccionada y muestra la receta con todas sus propiedades. |

1. Consultar recetas por calorías:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona buscar por dificultad |  |
|  | 3.El sistema muestra la pantalla para buscar por calorías |
| 4.El usuario escribe la cantidad de calorías máximas que desea que tengan sus recetas |  |
| 5.El usuario presiona el botón de “buscar recetas” |  |
|  | 6. El sistema procesa la opción seleccionada y muestra todas las recetas que se corresponden. |
| 9. El actor selecciona la receta que desea  visualizar | 10. El sistema procesa la receta seleccionada y muestra la receta con todas sus propiedades. |

1. Consultar recetas por tiempo de preparación:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona buscar por tiempo |  |
|  | 3.El sistema muestra la pantalla para buscar por tiempo |
| 4. El usuario escribe el tiempo que desea que tarden en realizarse las recetas. |  |
| 5.El usuario presiona el botón de “buscar recetas” |  |
|  | 6. El sistema procesa la opción seleccionada y muestra todas las recetas correspondientes. |
| 9. El actor selecciona la receta que desea  visualizar | 10. El sistema procesa la receta seleccionada y muestra la receta con todas sus propiedades. |

1. Buscar todas las recetas

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2. El usuario presiona el botón de buscar todas las recetas. |  |
|  | 3. El sistema muestra la pantalla todas las recetas con sus propiedades. |

1. Enviar sugerencia:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona enviar sugerencia |  |
|  | 3.El sistema muestra la pantalla para enviar la sugerencia |
| 4. El usuario escribe su nombre, email y sugerencia |  |
| 5.El usuario presiona el botón de “enviar” |  |
|  | 6. El sistema procesa los datos introducidos. |
|  | 7.El sistema guarda la sugerencia  enviada y muestra un aviso de que la  sugerencia ya ha sido procesada |

1. Consultar ayuda:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona la pestaña *consultar ayuda* |  |
|  | 3.El sistema muestra la pantalla de ayuda |

1. Consultar quienes somos:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona la pestaña *quienes somos* |  |
|  | 3.El sistema muestra la pantalla de quienes somos |

1. Consultar como funciona:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario selecciona la pestaña *como funciona* |  |
|  | 3.El sistema muestra la pantalla de cómo funciona |

1. Contactar a través de Facebook:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2.El usuario presiona sobre el icono de Facebook |  |
|  | 3. El sistema redirige al usuario a la página de Facebook de la aplicación. |

1. Contactar a través de Google+:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2. El usuario presiona sobre el icono de Google+. |  |
|  | 3. El sistema redirige al usuario a la página de Google+ de la aplicación. |

1. Contactar a través de Twitter:

|  |  |
| --- | --- |
| Acción del usuario | Respuesta del sistema |
| 1.El usuario accede a la aplicación |  |
| 2. El usuario presiona sobre el icono de Twitter. |  |
|  | 3. El sistema redirige al usuario a la página de Twitter de la aplicación. |

# Capítulo 4: Diseño de la descripción informática

## 4.1 Diseño

En este capítulo se describirán aspectos del diseño de la aplicación. Principalmente consta de los siguientes apartados:

-Una descripción de la arquitectura de la aplicación, donde se especifican los módulos necesarios para la misma así como también una descripción de las funciones que desempeñan cada uno de ellos.

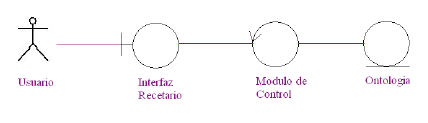
-Una descripción detallada del diseño de la ontología. En ella se incluye una descripción detallada de cada uno de los módulos que componen la misma así como los criterios utilizados para representar la misma.

-Una descripción del diseño de la interfaz en donde se explica como el usuario puede interactuar con la misma.

-Una descripción del módulo de control en el cual se explica cuáles son las funciones y operaciones utilizadas y qué es lo que puede hacer el usuario en cada una de ellas.

### **4.1.1 Arquitectura**

### En función de la arquitectura podemos apreciar tres grandes módulos que se muestran en el siguiente diagrama:



1. Ilustración

La interfaz es el modulo que se encuentra en contacto con el usuario. Esta se presenta como una aplicación web en la cual le muestra al usuario todas las posibles operaciones que este puede realizar sobre la misma. Esta interfaz, mostrará las posibles operaciones de búsqueda que se pueden realizar, las propiedades de la misma una vez realizada la búsqueda y las diferentes operaciones que se pueden realizar en la interfaz (como consultar la ayuda o enviar una sugerencia entre otros).

El segundo módulo, es el de control. Es el que se encarga de hacer de intermediario entre la interfaz de usuario y la ontología, es decir la base de datos. Este módulo está constituido por elementos que se encargan de procesar la información recibida por la interfaz y enviársela a la ontología (en un formato legible para esta). Una vez procesada la información la ontología se la devuelve al módulo de control y esta se la devuelve a la interfaz (en un formato legible para esta) para que finalmente le llegue al usuario.

Finalmente el tercer módulo es el de la ontología. Este es el módulo de almacenamiento donde se guardan todos los elementos y relaciones de la misma. En nuestro caso están guardadas todas las recetas con su correspondiente información.

### **4.1.2 Diseño de la ontología**

### Una ontología es una especificación de conceptos así como también la relación entre ellos. Para poder describir ontologías existen diferentes lenguajes y formalismos, entre los cuales los más utilizados son RDF (proporciona un lenguaje básico para describir ontologías) y OWL [9](extiende RDF).

En nuestro caso concreto, la ontología se utiliza para representar recetas y su información. Tras un gran análisis de la aplicación que se quería llevar a cabo, junto con la consulta y análisis de numerosas fuentes en internet y libros, se propuso la ontología que se ve en la figura 1.

En la oncología que vemos anteriormente se muestra una clase principal llamada receta que contiene recetas e ingredientes.

-Clase receta: Está compuesta por todas las recetas contenidas en su base de datos. En la figura 2 se muestra cómo está representada la clase recetas.

-Clase ingrediente: Está compuesta por todos los ingredientes contenidos en la base de datos. En la figura 3 se puede apreciar cómo están representados los ingredientes.

Para mayor claridad a continuación en la figura 4 se muestra como está representada una receta.

Como podemos observar en la imagen una receta está formada por un nombre de la receta, y una serie de nodos que representan cada uno de los ingredientes que componen dicha receta. Además, hay otros atributos que forman parte de la receta que son el modo de preparación, que representa los pasos que hay que seguir para llevar a cabo la receta. También hay un atributo que marca la dificultad de la receta (1, 2,3 correspondientes a bajo media o alta dificultad respectivamente), las calorías totales de la misma y el tiempo medio de realización de la receta. Además hay otro atributo denominado lista de ingredientes en la cual está contenido todos los ingredientes con sus respectivas cantidades. Esto se ha hecho de este modo, porque (pensar bien cómo explicar esto que es importante)

### **4.1.3 Diseño de la interfaz**

La aplicación fue diseñada para que el usuario pueda buscar las recetas que más se ajustaban a sus necesidades en cada momento. También está diseñada para que este pueda buscar por dificultad, calorías o tiempo de preparación. Es por ello, que la interfaz se intentó hacer lo más fácil y sencilla posible para que cualquiera, fuera cual fuera sus conocimientos, pudiese obtener resultados satisfactorios.

Diagrama de navegación

Buscar por ingredientes

Facebook

Buscar por tiempo de preparación

Inicio

Twitter

Buscar por tiempo máximo de preparación

Google+

Buscar por tiempo mínimo de preparación

Página principal

Buscar por grado de dificultad

Recetas

Buscar entre dos tiempos de preparación

Ayuda

Buscar recetas de dificultad media y baja

Sugerencias

Quienes somos

Buscar recetas de dificultad media y alta

Cómo funciona

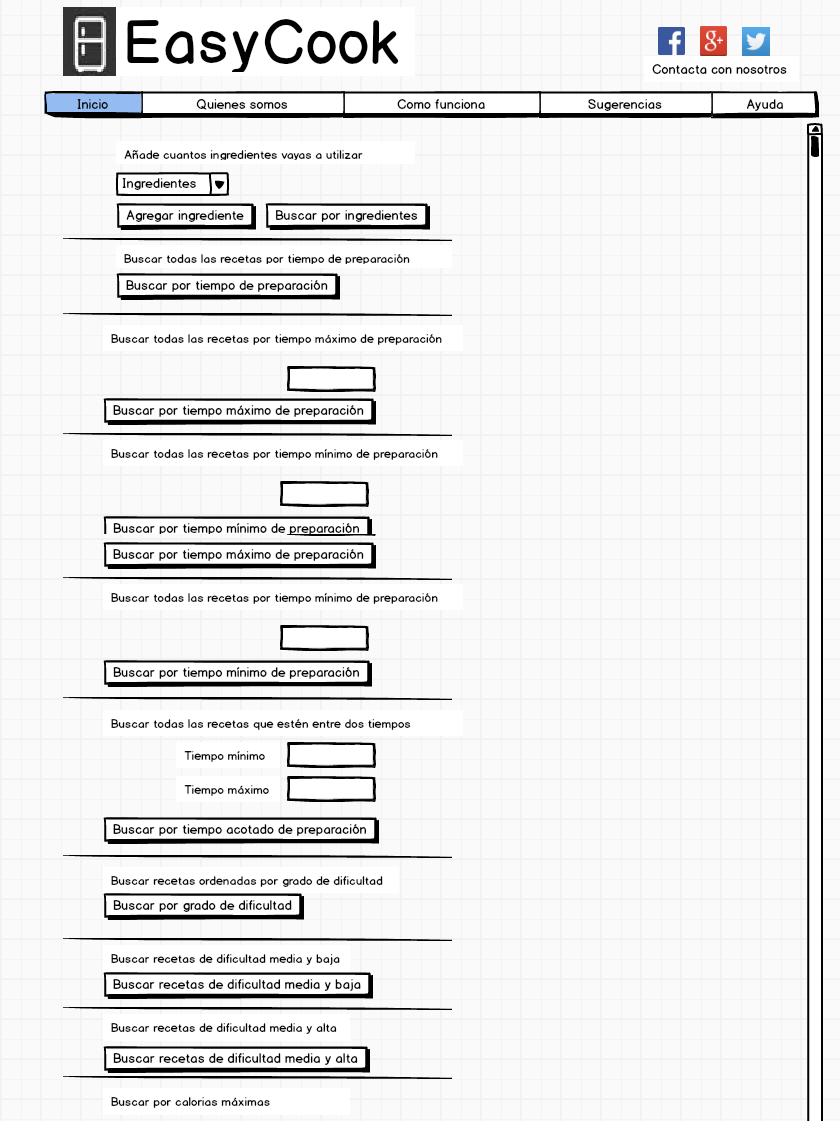
Buscar todas

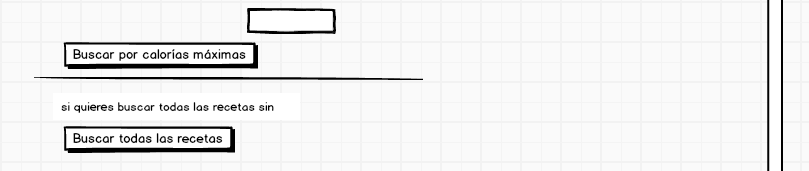
Buscar por calorías

#### Diseño previo de la interfaz de usuario

### Es importante realizar un diseño previo de la interfaz de usuario con el fin de plasmar el esquema y la funcionalidad de la misma. Es por ello que para este proyecto se ha utilizado una herramienta que permite esbozar cada una de las pantallas contenedoras de nuestra aplicación. A continuación se muestra el prototipo de cada una de las pantallas.

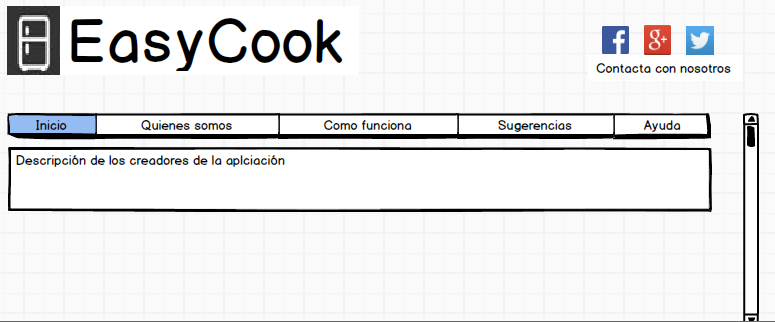
* Pantalla de inicio



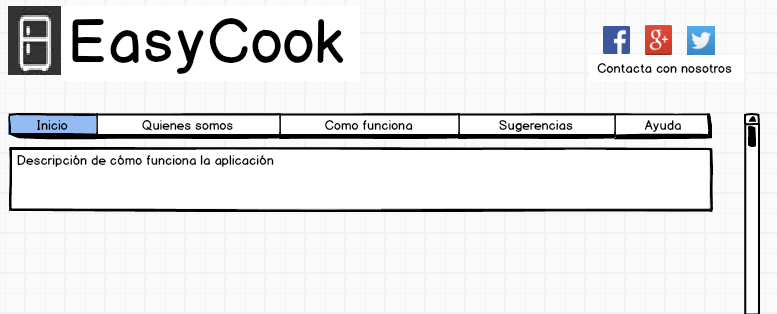


1. Ilustración

* Pantalla Quienes somos

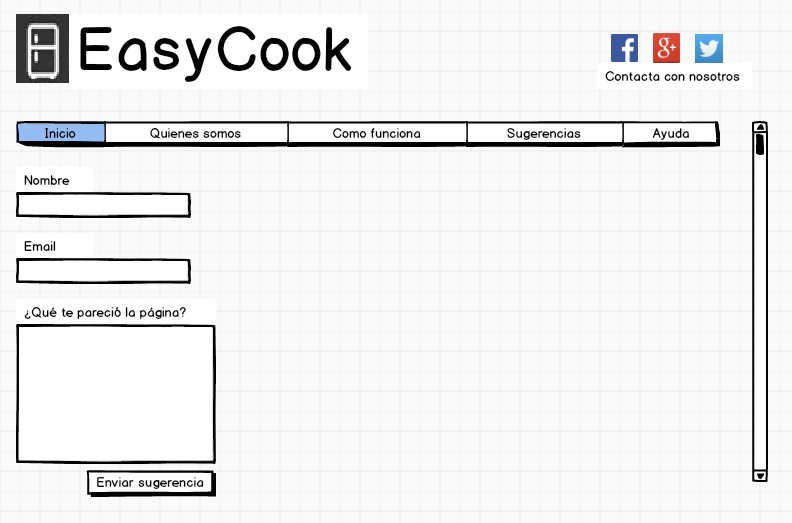


1. Ilustración

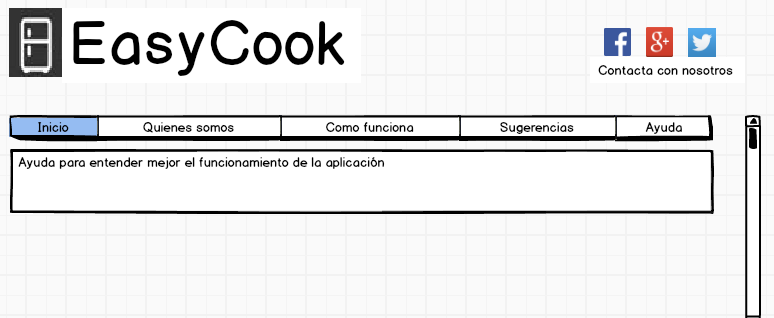
* Pantalla como funciona

1. Ilustración

* Pantalla sugerencias

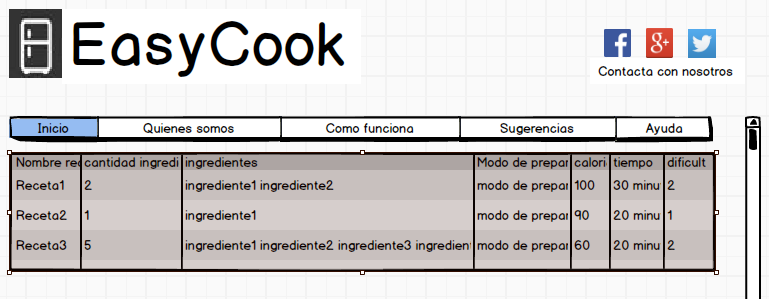


1. Ilustración

* Pantalla ayuda

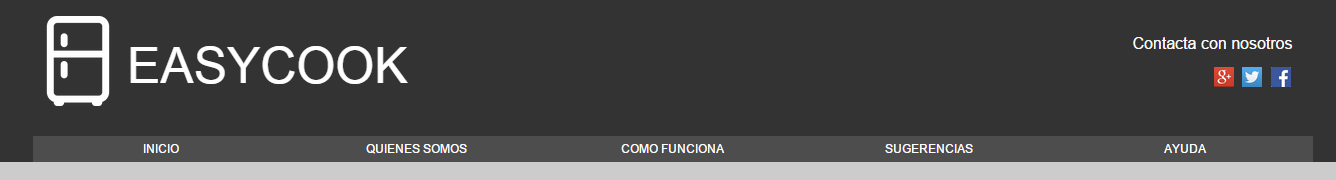
1. Ilustración

* Pantalla que muestra las recetas

Ilustración

1. Ilustración

La interfaz fue diseñada para que cualquier tipo de usuario, pueda acceder a realizar cualquier tipo de gestión en la aplicación. Por esto, la ventana principal está dividida en secciones: inicio, quienes somos, como funciona, sugerencias y ayuda. Además, en la parte superior de la aplicación podemos observar que hay tres símbolos que presionando sobre cada uno de ellos, el usuario puede contactar con los administradores de la aplicación a través de redes sociales.



A continuación se realizará una explicación sobre la funcionalidad y el funcionamiento detallado de cada una de las pestañas.

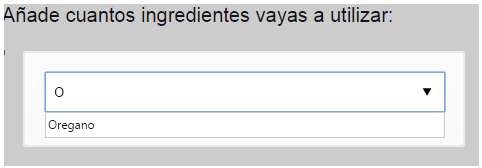
1. Ilustración

Presionando sobre la pestaña de inicio es donde se puede realizar las diferentes búsquedas. Estas se detallan a continuación:

* La imagen XXX muestra como el usuario es capaz de buscar las recetas que se ajustan de la mejor manera a los ingredientes que él desea. De este modo, debe escribir un ingrediente y presionar sobre el botón de *agregar ingrediente*s si desea agregar más ingredientes o presionar sobre el botón de *buscar por ingredientes* en el caso de que quiera agregar más de un ingrediente. En el caso de que desea agregar más de un ingrediente, debe escribir el ingrediente deseado y apretar sobre el botón correspondiente tantas veces como ingredientes desea agregar.

1. Ilustración
2. Ilustración

Para evitar posibles errores por parte del usuario al escribir los ingredientes, a medida que este va escribiendo el ingrediente deseado, le va apareciendo todos los ingredientes que coinciden con las letras introducidas hasta el momento. Esto se muestra en la imagen XXX

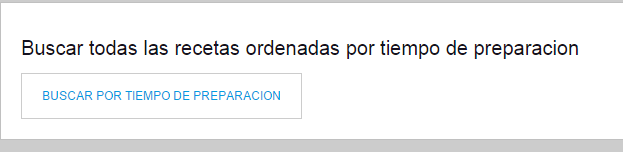
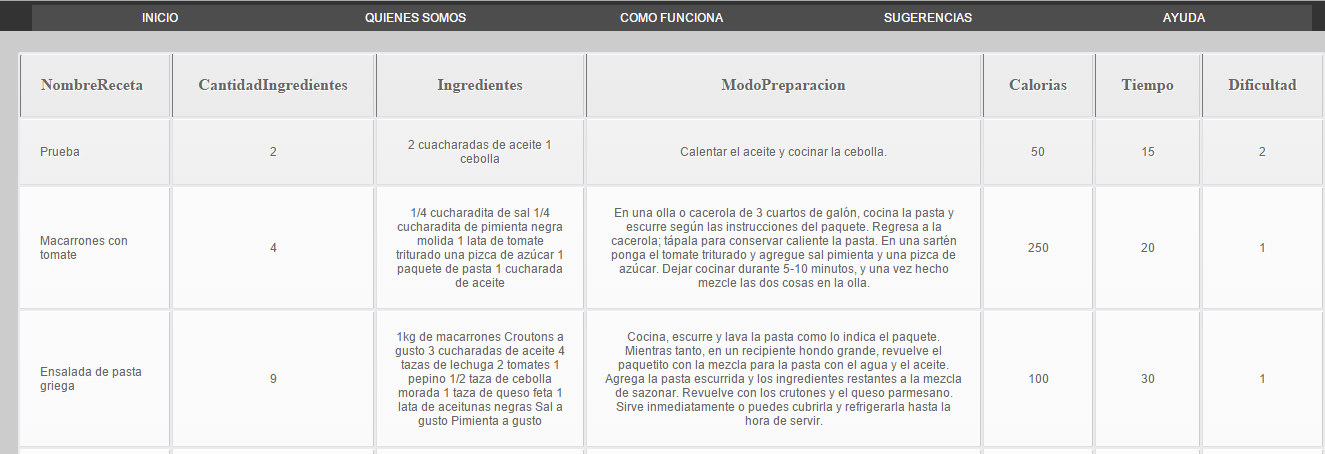


1. Ilustración

Una vez el usuario selecciona todos los ingredientes deseados, simplemente debe presionar el botón de *buscar por ingredientes* y la pagina mostrará las recetas que contengan únicamente esos ingredientes en primer lugar, a continuación las recetas que contengan esos ingredientes y uno menos, y dos menos, etc. Y finalmente las recetas que contengan esos ingredientes y uno más, y dos más etc. Junto con ellas, se muestra además, los ingredientes junto con sus cantidades, el modo de preparación de cada una de ellas, la cantidad de ingredientes que contiene, la dificultad y las calorías totales. En la imagen XXX puede apreciarse los resultados si el usuario introduce como ingredientes cebolla, sal y aceite.

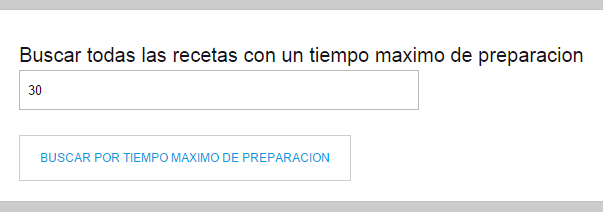
1. Ilustración

* La imagen XXX muestra lo que tiene que hacer el usuario si quiere buscar todas las recetas ordenadas por tiempo de preparación y en la imagen XXX parte de los resultados obtenidos.



1. Ilustración
2. Ilustración

* La imagen XX muestra lo que tiene que hacer el usuario si quiere buscar todas las recetas con un tiempo de preparación máximo.



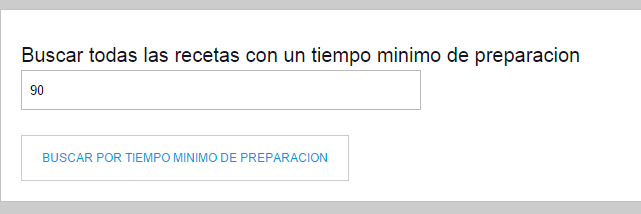
1. Ilustración

En este ejemplo se deseó poner un tiempo máximo de 30 minutos, y con esto, los resultados obtenidos se muestran en la imagen XXX.



1. Ilustración

* La imagen XXX muestra lo que tiene que hacer el usuario si desea buscar las recetas por un tiempo mínimo de preparación.

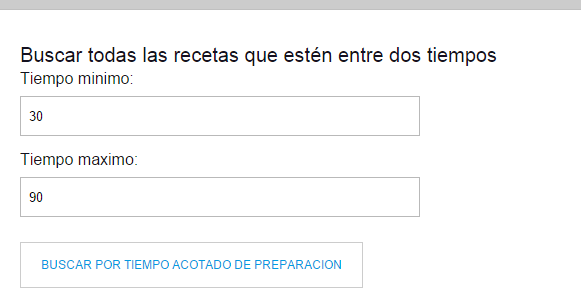


1. Ilustración

En este En este ejemplo se deseó poner un tiempo mínimo de 90 minutos, y con esto, los resultados obtenidos se muestran en la imagen XXX.

* La imagen XXX muestra lo que debe hacer el usuario si desea buscar recetas acotadas en un mínimo y máximo tiempo de preparación.

1. Ilustración

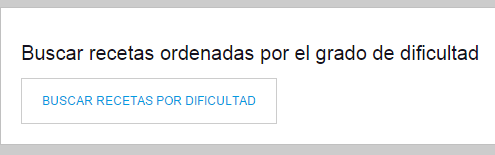
En este ejemplo se puso un tiempo máximo de 90 minutos y mínimo de 30. Con esto los resultados se aprecian en la imagen XXX.

1. Ilustración



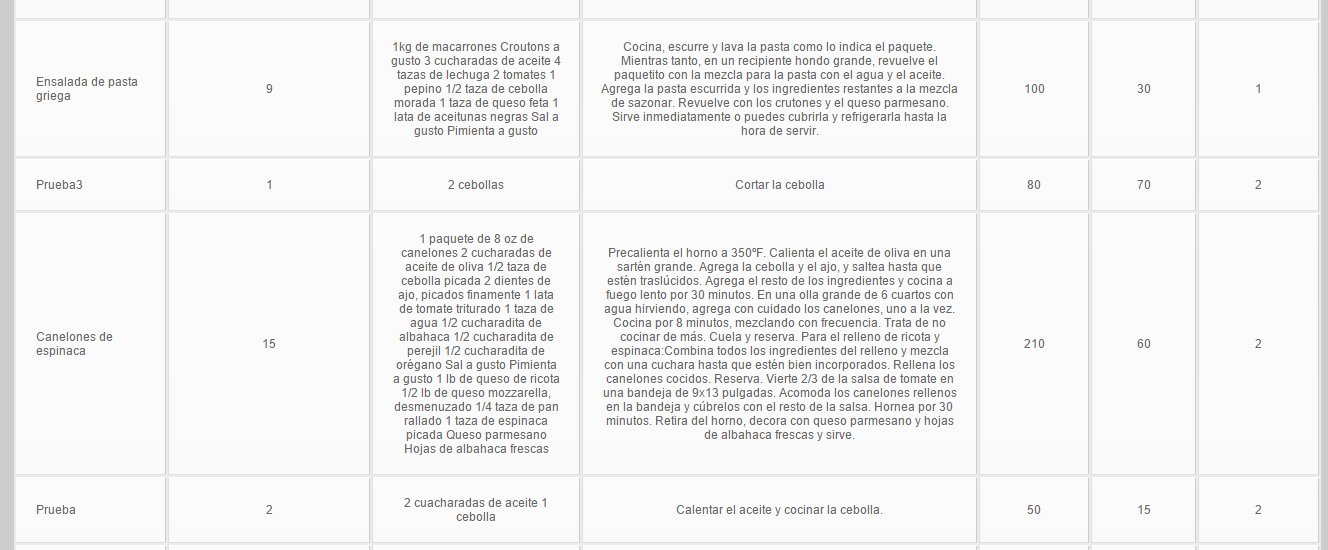
1. Ilustración

* La imagen XXX muestra lo que debe hacer el usuario si desea buscar todas las recetas ordenadas por grado de dificultad.

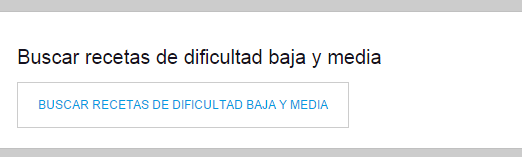


1. Ilustración

La imagen XXX muestra algunos de los resultados obtenidos.

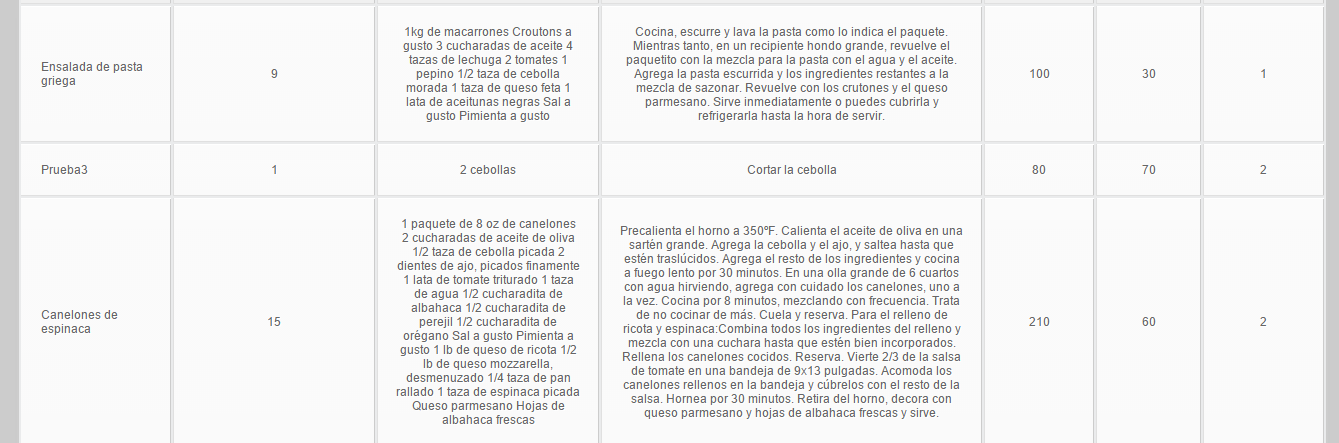


1. Ilustración

* La imagen XXX muestra lo que debe hacer el usuario si quiere buscar únicamente las recetas de dificultad baja y media.

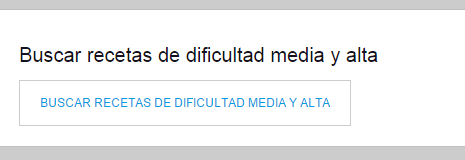
1. Ilustración

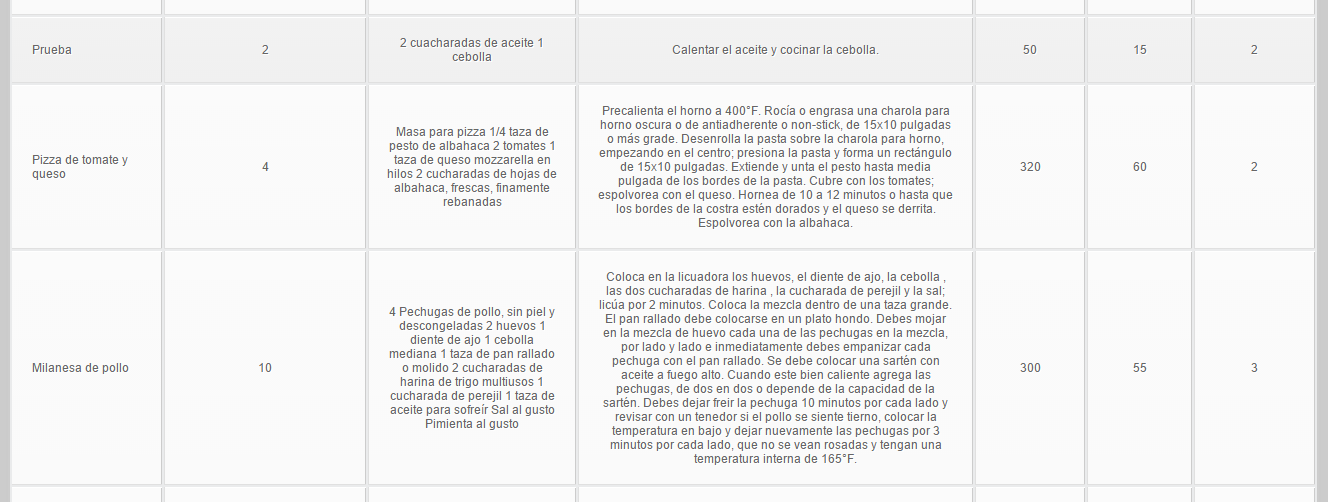
La imagen XXX muestra parte de los resultados obtenidos.



1. Ilustración

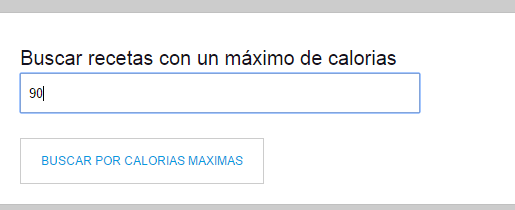
* La imagen XXX muestra lo que debe hacer el usuario si quiere buscar únicamente las recetas de dificultad media y alta.



La imagen XXX muestra parte de los resultados obtenidos.

1. Ilustración
2. Ilustración

* La imagen XXX muestra lo que debe hacer el usuario si desea buscar recetas con un número máximo de calorías.



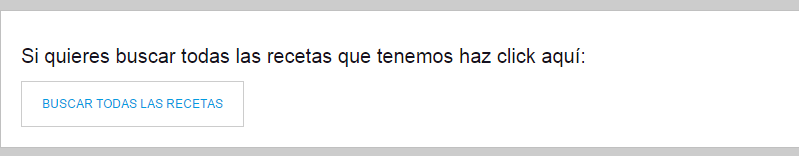
1. Ilustración

En este ejemplo se deseó poner un máximo de 90 calorías, y con esto, los resultados obtenidos se muestran en la imagen XXX.



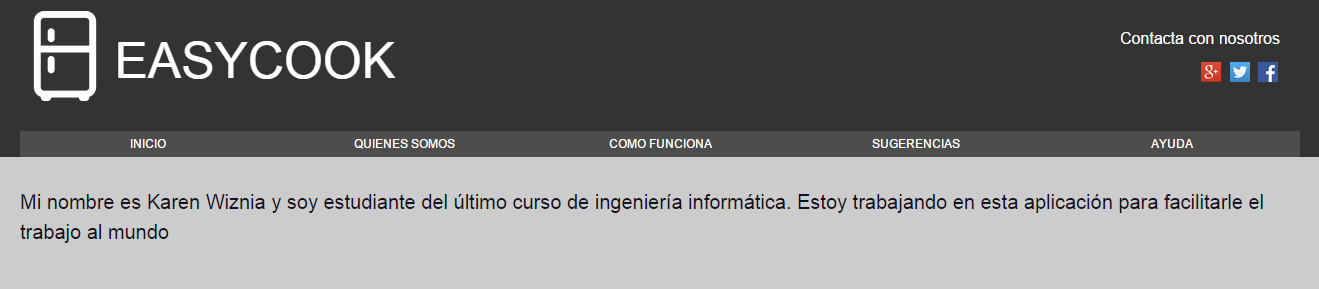
1. Ilustración

* En la imagen XXX se muestra lo que debe hacer el usuario si desea buscar todas las recetas sin aplicar ningún filtro. Parte de los resultados obtenidos se pueden observar en la imagen XXX.



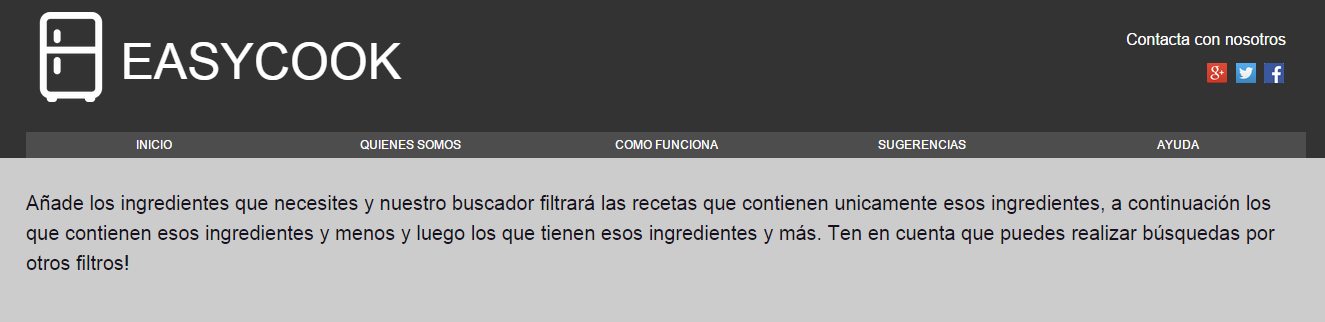
1. Ilustración
2. Ilustración

Presionando sobre la pestaña de quienes somos, hay una pequeña explicación sobre quienes fueron los que realizaron la aplicación. Esto se aprecia en la imagen XXX



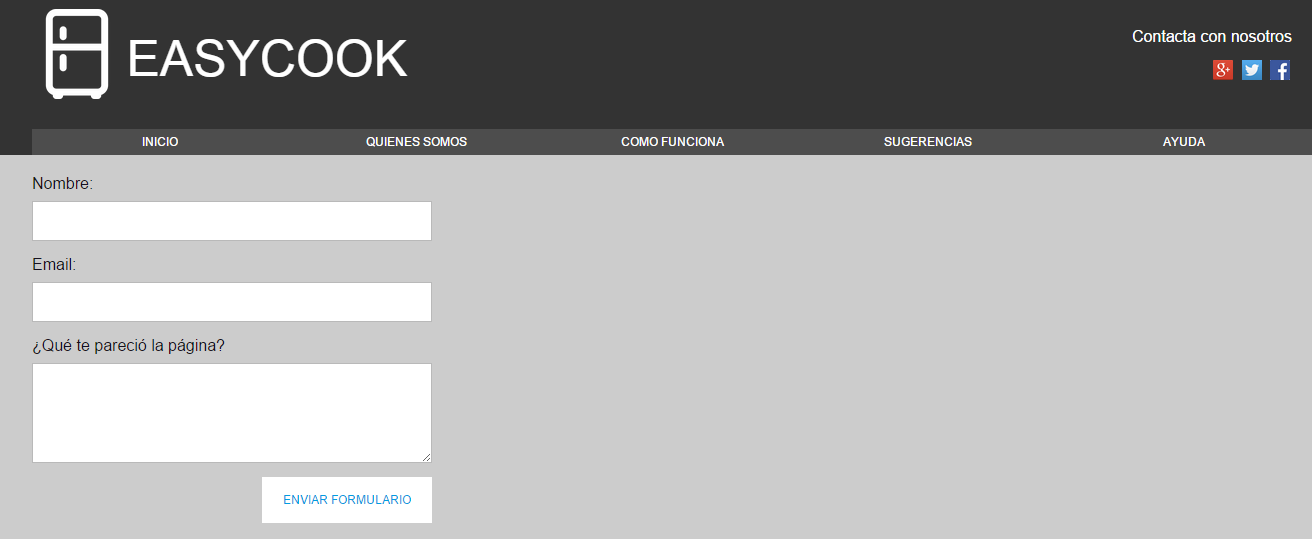
1. Ilustración

Presionando sobre la pestaña de cómo funciona el usuario puede leer una breve descripción de cómo funciona la aplicación y las distintas funcionalidades que tiene. Esto puede apreciarse en la imagen XXX



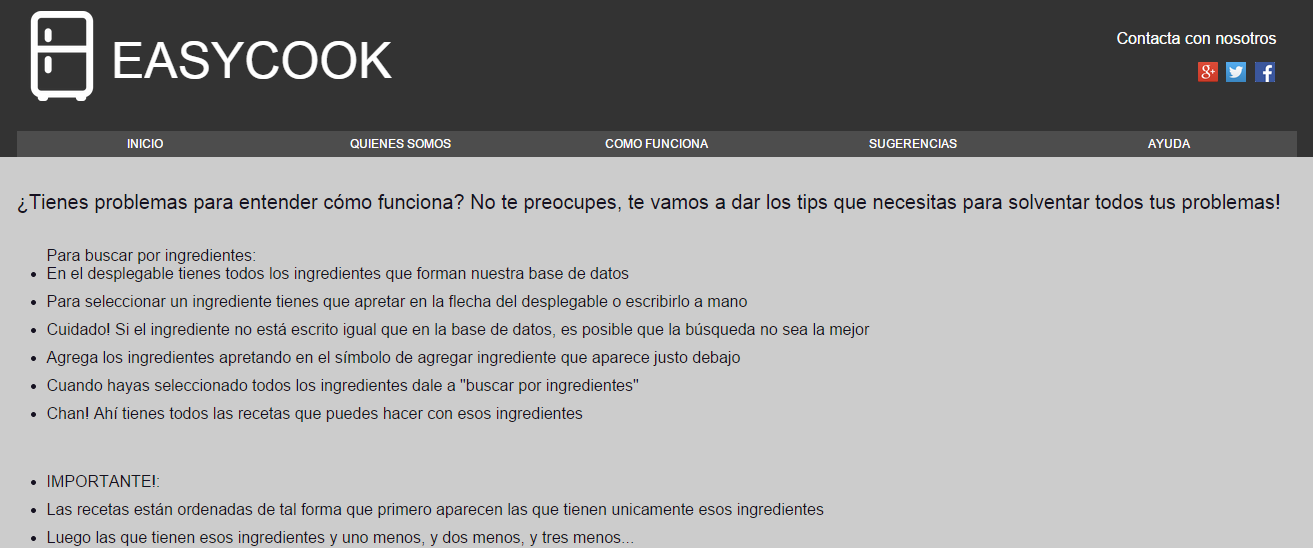
1. Ilustración

Presionando sobre la pestaña de sugerencias es posible enviar un comentario, una sugerencia de mejora, una nueva funcionalidad, etc. Que el usuario considere oportuna, siempre y cuando rellene los campos de nombre, email y una breve descripción, tal y como puede apreciarse en la imagen XXXX.



1. Ilustración

Presionando sobre la pestaña de ayuda, el usuario podrá ver una explicación detallada sobre cómo se utiliza la aplicación, así como también algunos consejos para un uso más eficiente de la misma. Esto puede apreciarse en la imagen XXX.

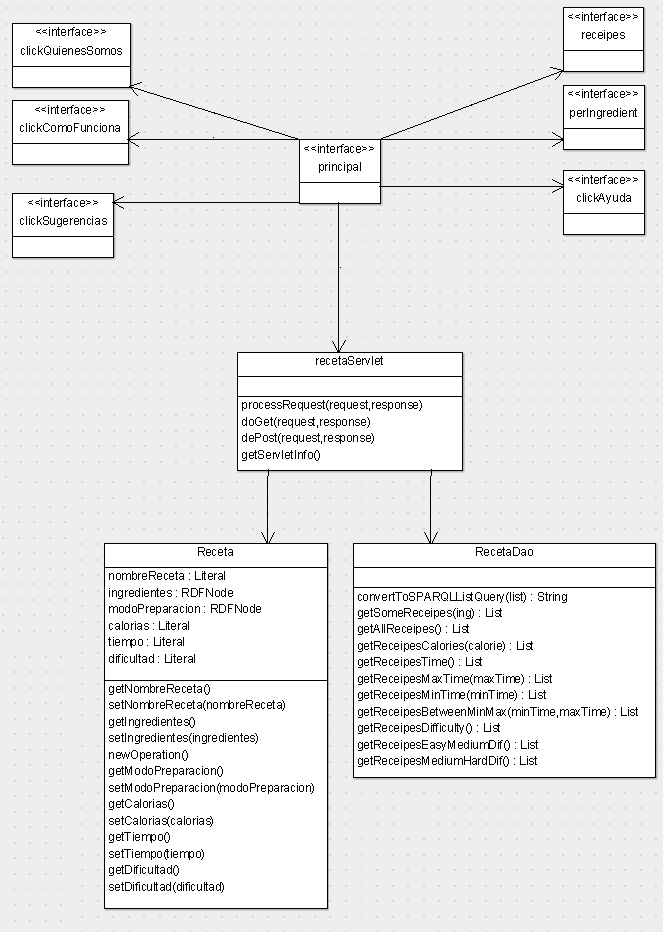


1. Ilustración

### **4.1.4 Diseño del módulo de control**

El módulo de control, como bien se ha especificado antes, es el encargado de la funcionalidad general de la aplicación. Además de esta función, también es el encargado de conectar la interfaz (es decir lo que ve el usuario) con la ontología (es decir con la información existente en la base de datos).

Como se puede ver en la imagen 1, la aplicación se ha implementado con un modelo-vista-controlador.



1. Ilustración

Modelo

Vista

Controlador

A continuación se detalla cada uno de los bloques mostrados en la imagen:

La vista

Aquí se define la parte visual del sistema, es decir lo que está en contacto con el usuario. Se establecen todas las interfaces necesarias para poder acceder a las diferentes partes de la aplicación.

* Principal: Es la ventana de inicio que ve el usuario en cuanto entra a la página.
* clickQuienesSomos: Es la ventana que ve el usuario cuando presiona la pestaña de *quienes somos.*
* clickComoFunciona: Es la ventana que ve el usuario cuando presiona la pestaña de *cómo funciona.*
* clickSugerencia: Es la ventana que ve el usuario cuando presiona la pestaña de *sugerencia.*
* clickAyuda:Es la ventana que ve el usuario cuando presiona la pestaña de *ayuda.*
* Receipes: Es la ventana que ve el usuario cuando selecciona la opción de buscar todas las recetas.
* perIngredient: Es la ventana que ve el usuario cuando selecciona algún criterio de búsqueda de las recetas, cualquiera que este fuese.

El controlador

La clase recetaServlet es la encargada de definir la lógica de administración del sistema, establecer la conexión entre la vista y el modelo. Es decir, es quien actúa de intermediario entre la interfaz y el modelo. Este contiene un método llamado processRequest el cual recibe información de la interfaz, y se la pasa al modelo para que este pueda realizar las operaciones necesarias.

El modelo

Aquí se define la lógica de negocio. Es decir, la funcionalidad del sistema. En este bloque podemos distinguir dos grandes clases:

Receta: Aquí representamos los diferentes atributos de la base de datos. Cada receta, contenida en nuestra base de datos contiene nombre de la receta, ingredientes, cantidad, calorías, dificultad y modo de preparación. Por ende, nuestra clase recetas tendrá estos mismos atributos y de esta manera podemos transportar un objeto receta con todos estos valores por medio de los métodos get y set implementados en la misma.

RecetaDao:En esta clase, tenemos la lógica de negocio propiamente dicha. Es en esta, donde se realizan las consultas necesarias a la base de datos, y obtiene la información correspondiente por parte de la misma. Dentro de esta tenemos una serie de métodos:

* convertToSparqlListQuery: Metodo auxiliar para utilizar dentro de las diferentes consultas SPARQL en java.
* getSomeReceipes: Este método recibe una lista de ingredientes y devuelve una lista de Receta. Se encarga de devolver las recetas que contienen únicamente los ingredientes contenidos en la lista que recibe como parámetro, además lo concatena con las recetas que contienen dichos ingredientes y uno más, dos más, etc. junto con las recetas que contienen esos ingredientes y uno menos, dos menos, etc.
* getAllReceipes: Este método devuelve una lista con todas las recetas almacenadas en la oncología.
* getReceipesCalories: Este método recibe un valor, correspondiente a la cantidad máxima de calorías que desea que contenga la receta, y devuelve una lista de recetas ordenadas por calorías de menos a mayor.
* getReceipesTime: Este método devuelve una lista con todas las recetas, por el tiempo de preparación de manera ascendente.
* getReceipesTimeMax: Este método recibe un valor, correspondiente al tiempo máximo de preparación, y devuelve una lista de recetas ordenadas de manera ascendente.
* getReceipesTimeMin: Este método recibe un valor, correspondiente al tiempo mínimo de preparación, y devuelve una lista de recetas ordenadas de manera ascendente.
* getReceipesTimeBetweenMinMax: Este método recibe un valor, correspondiente al tiempo mínimo y máximo de preparación, y devuelve una lista de recetas, cuyo tiempo de preparación está correspondido entre esos dos valores, ordenadas de manera ascendente.
* getReceipesDifficulty: Este método devuelve una lista con todas las recetas, ordenadas según el nivel de dificultad de las mismas, de manera ascendente (es decir de más fácil a más difícil).
* getReceipesEasyMediumDif: Este método devuelve una lista con todas las recetas cuya dificultad es baja o media, mostrando primero las de dificultad baja y a continuación las de dificultad media.
* getReceipesMediumHardDif: Este método devuelve una lista con todas las recetas cuya dificultad es media o alta, mostrando primero las de dificultad media y a continuación las de dificultad alta. Añadir aquí una imagen sacada de netbeans con las clases y tal

# Capítulo 5: Conclusiones y trabajos futuros

## Conclusiones

La realización del proyecto me ha llevado a sacar diferentes conclusiones tanto a nivel académico y profesional como personal. Por un lado a nivel académico y profesional, gracias a la realización del mismo fui capaz de ampliar mis conocimientos sobre la web semántica y lo que se puede realizar con ella, destacando la facilidad que proporciona a la hora de realizar búsquedas eficientes utilizando SPARQL.

En este aspecto, también cabe destacar la gran variedad de clases, propiedades, relaciones, entre otros que tanto RDF como OWL nos permiten definir. Esto nos permite una mayor claridad y facilidad a la hora de ser tratados y utilizados.

Por motivos como estos, entre otros, es importante poder desarrollar tanto aplicaciones como lenguajes de utilización referentes a la Web Semántica ya que abre una visión nueva de la web y abre una ventana con infinidad de posibilidades nuevas en las que el usuario podrá disponer de una web más eficiente y fácil de utilizar.

A nivel personal, la realización de este proyecto me hizo desarrollarme como profesional. Me hizo darme cuenta que a pesar de no saber nada sobre una tecnología o lenguaje, eso no me impide la amplia capacidad de aprender que adquirí durante estos cuatro años de carrera. Me ayudo a darme cuenta, que a pesar de no estar familiarizada con ciertas tecnologías o lenguajes de programación, que tengo la capacidad de aprender y realizar cualquier cosa que me aparezca por el camino. También aprendí, que las buenas ideas no son las más complejas de entender ni de explicar, sino que son las que están bien hechas y que llegan a cubrir las necesidades de las personas a las que va destinadas.

## Estimación del esfuerzo

Para realizar un correcto desarrollo del proyecto, es necesario hacer una buena planificación y división de las tareas a desarrollar, estableciendo para cada una de ellas una duración estimada y un orden cronológico.

A pesar de que esta estimación no sea exacta, hacerla inicialmente tiene ciertas ventajas como puede observarse a continuación:

* Asegurar el proceso de desarrollo, estableciendo los recursos necesarios y el tiempo necesario para la finalización.
* Establecer plazos fiables para cada una de las fases del proyecto.
* Mantener un control del estado del proyecto pudiendo identificar problemas y retrasos en el desarrollo del mismo.
* Ayudar a tomar decisiones correctas en cada momento.

## En la sección 1.4 pueden apreciarse dos diagramas detallados sobre la división de las etapas a lo largo del desarrollo del mismo. Además hay que tener en cuenta la carga lectiva que este conlleva, ya que un proyecto de fin de grado son 15 créditos, que en términos de tiempo equivale a unas 600 horas de trabajo, es decir 40 horas por crédito. Para el cómputo de horas totales dedicadas al proyecto se considero una media de 4 horas al día, teniendo en cuenta que los meses con más carga lectiva (es decir con más exámenes o prácticas), como pueden ser los meses de diciembre y enero, esta media se redujo considerablemente. Dicho esto, se estableció como fecha de comienzo el 15 de septiembre y como fecha final el 20 de mayo. Finalmente esta estimación fue positiva, ya que la fecha de finalización real fue el 1 de mayo.5.3 Análisis de la consecución de objetivos

Al comenzar este proyecto de fin de grado se fijó como objetivo crear un algoritmo que dado una seria de ingredientes te devolviese las recetas que contienen esos ingredientes únicamente. A medida que la idea se iba generando, se fueron sacando una serie de conclusiones tales como que el usuario también podría querer ver las recetas que contuviesen los ingredientes introducidos y alguno menos, al igual que las recetas que contienen algún ingrediente adicional al introducido.

Una vez realizada esta funcionalidad se decidió añadir otras funcionalidades que pueden ser interesantes para el usuario tales como buscar por tiempo de preparación o introduciendo una cantidad máxima de ingredientes, entre otras funcionalidades extras.

Durante la realización del proyecto, se marcaron una serie de objetivos en las diferentes fases que se han ido cumpliendo.

Durante la fase 1, se realizó un estudio exhaustivo de las páginas y aplicaciones cuya funcionalidad era parecida. Para cada una de ellas se identificaron los puntos positivos, los puntos negativos, si cumplía la funcionalidad para la cual fue implementada, entre otras cosas.

Durante la fase 2 se decidieron las tecnologías a utilizar, estudiando los pros y los contras de utilizar unas u otras. Además, también fue necesario aprender y familiarizarse con la mayoría de ellas.

Durante la fase 3 se desarrolló el algoritmo de búsqueda. Se realizaron las consultas necesarias para que el algoritmo funcione satisfactoriamente.

Durante la fase 4 se procedió al diseño y la implementación de la interfaz de usuario.

Durante la fase 5, última fase del desarrollo del mismo, se realizaron múltiples pruebas para asegurarse el correcto funcionamiento de la aplicación, y con ello la modificación de diferentes partes del algoritmo para que sea exitoso en múltiples escenarios.

## Mejoras futuras

Como todo proyecto informático, podríamos considerar que nunca está finalizado, siempre hay una serie de nuevas funcionalidades que se pueden añadir, o incluso mejorara las existentes, así como la interfaz de usuario. En este caso las mejoras a añadir podrían ser:

* Añadir un filtro para vegetarianos con el fin de que busque las recetas que no contengan ningún tipo de animal.
* Añadir un filtro para celíacos, con el fin de que busque recetas que no contengan productos no aptos para celiacos.
* Añadir un filtro para que te saque las recetas adecuadas según la temporada del año que el usuario escoja.
* Permitir al usuario valorar la receta del uno al cinco.
* Poder buscar las recetas más valoradas.

Una de las razones por las cuales se decidió hacer esta aplicación como trabajo de fin de grado, fue ya que se llevará a cabo y se implementará como nueva funcionalidad a la página <http://www.cocinario.es/>. Esta es una red social de recetas cuya funcionalidad no está implementada, con lo cual el algoritmo será llevado a cabo en esta página web para estar a disposición de cualquier usuario que desea realizar dicha búsqueda.

# Capítulo 6: Bibliografía y lugares de internet

## Páginas cuya funcionalidad es parecida

* <http://www.quehayenlanevera.com/>
* <http://www.hogarutil.com/>
* <http://www.quierucomer.com/>
* <http://www.mis-recetas.org/>
* <http://www.rebanando.com/>
* <http://sevilla.abc.es/mobility/las_mejores_app/android/las-mejores-app-android/las-10-mejores-aplicaciones-android-de-recetas-de-cocina/>
* <http://www.maquecitos.com/2011/09/13/recetas-y-compras-que-ingredientes-tengo-y-que-cocino/>
* <http://www.todareceta.es/>

## Páginas sobre SPARQL, RDF y Jena

* <http://en.wikipedia.org/wiki/SPARQL>
* <http://www.w3.org/TR/rdf-sparql-query/>
* <http://www.cambridgesemantics.com/semantic-university/sparql-by-example>
* <http://www.learningsparql.com/>
* <http://stackoverflow.com/questions/12123270/how-to-display-the-output-of-sparql-queries-in-a-play-view>
* <http://thesai.org/Downloads/Volume4No6/Paper_20-Format_SPARQL_Query_Results_into.pdf>
* <http://stackoverflow.com/questions/12123270/how-to-display-the-output-of-sparql-queries-in-a-play-view>
* <http://www.w3.org/RDF/>
* <http://en.wikipedia.org/wiki/Resource_Description_Framework>
* <http://www.w3schools.com/webservices/ws_rdf_intro.asp>
* [https://github.com/JoshData/rdfabout/blob/gh-pages/intro-to-rdf.md#](https://github.com/JoshData/rdfabout/blob/gh-pages/intro-to-rdf.md)
* <https://jena.apache.org/documentation/rdf/index.html>
* <http://jena.apache.org/documentation/javadoc/arq/com/hp/hpl/jena/query/ResultSetFormatter.html>
* <http://en.wikipedia.org/wiki/Jena_%28framework%29>
* <https://github.com/castagna/jena-examples>
* <https://www.youtube.com/watch?v=nUdHneViLp4>

## Páginas sobre NetBeans

* <https://netbeans.org/>
* <https://netbeans.org/kb/docs/web/quickstart-webapps.html>
* <https://netbeans.org/kb/docs/web/mysql-webapp.html>
* <https://www.youtube.com/watch?v=4dyxKBaTdTE>
* <http://transvar.org/6112/WebStartAppInstruction.pdf>
* <http://www.objectdb.com/tutorial/jpa/netbeans/ee>
* <https://docs.oracle.com/cd/E19776-01/820-4867/ggqxv/>
* <http://javawebaction.blogspot.com.es/2013/12/custom-web-application-development-using-primefaces-jsf-java-ee-in-netbeans-id-and-glass-fish-web-application-server.html>

## Páginas sobre Protégé

* <http://protege.stanford.edu/>
* <http://webprotege.stanford.edu/>
* <http://webprotege.stanford.edu/#Edit:projectId=bb11a09e-29f5-47c5-b2f9-a81c3a88aa9d>
* <http://webprotege.stanford.edu/#Edit:projectId=16b48d29-dca5-4ed2-bdbc-e35653a71ba0>
* <http://en.wikipedia.org/wiki/Prot%C3%A9g%C3%A9_%28software%29>

## Páginas sobre HTML y CSS

* <http://www.w3schools.com/html/>
* <http://www.tutorialspoint.com/html/>
* <http://www.codecademy.com/en/tracks/web>
* <http://www.w3schools.com/tags/tag_button.asp>
* <https://css-tricks.com/use-button-element/>
* <http://www.rapidtables.com/web/html/html-link.htm>
* <http://www.ironspider.ca/format_text/fontsize.htm>
* <http://html-color-codes.info/>
* <http://www.csstablegenerator.com/>
* <https://css-tricks.com/complete-guide-table-element/>

## Otras páginas de interés

* <https://balsamiq.com/products/mockups/>
* <http://www.ganttproject.biz/>
* <http://www.stackoverflow.com>
* <https://github.com/>
* <http://sourceforge.net/projects/argouml.mirror/>
* <http://codejavu.blogspot.com.es/2013/06/ejemplo-modelo-vista-controlador.html>

# Definiciones y siglas

-[1] RDF

-[2] W3C

-[3] Bases de datos distribuidas

-[4] Web semántica

-[5] Protégé

-[6] SPARQL

-[7] Jena

-[8] Java

-[9] OWL