



BITPARTY

Proyecto μ Search

Versión 1.0

06/2014

Alberto Berbel Aznar

Javier Briz Alastrué

Héctor Francia Molinero

Daniel García Páez

Alejandro Gracia Mateo

Simón Ortego Parra



**Departamento de
Informática e
Ingeniería de Sistemas
Universidad Zaragoza**

Índice

1. Introducción	5
1.1. Identificación del proyecto	5
1.2. Objetivo y alcance	5
1.3. Identificación del equipo	5
1.4. Descripción del contenido	6
2. Requisitos del sistema	7
3. Descripción técnica	9
3.1. Aspectos arquitecturales y tecnológicos	9
3.2. Modelos de datos	10
3.3. Diseño de los componentes	11
4. Verificación y validación	12
4.1. Metodología de pruebas	12
4.2. Pruebas. Defectos. Correcciones	12
5. Manuales	16
5.1. Manual de usuario	16
5.2. Manual de instalación	16
6. Gestión del proyecto	17
6.1. Fases y actividades	17
6.1.1. Riesgos	17
6.1.2. Estimaciones iniciales	17
6.1.3. Cronogramas	18
6.1.4. Tareas y estimación de esfuerzos	18
6.1.5. Ficheros de esfuerzos individuales	20
6.1.6. Esfuerzos reales de las tareas	20
6.1.7. Esfuerzos reales de las personas	21
6.2. Procesos de seguimiento y control	23
6.2.1. Calendario de reuniones	23
6.2.2. Actas de reuniones	24
6.3. Coste	24
7. Gestión de configuraciones	25
7.1. Políticas de nombrado	25
7.2. Control de versiones	25
7.3. Copias de seguridad	25
7.4. Elementos de configuración y línea base	25

8. Aseguramiento de la calidad	26
8.1. Estándares utilizados	26
8.2. Planificación de las auditorías	26
8.3. Auditorías	26
8.4. Auditoría externa	26
8.5. No conformidades	27
9. Postmortem	28
9.1. Lecciones aprendidas	28
9.2. Problemas encontrados	29
9.3. Catálogo de riesgos	29
9.4. Diferencias entre los esfuerzos y tamaños reales y los estimados	29
9.5. Plan real vs planificación inicial	29
9.6. Diferencias entre el coste real y el presupuesto	29
10. Conclusiones	30
10.1. Conclusiones del proyecto	30
10.2. Ideas de mejora del proceso	30
10.3. Ideas de mejora del desarrollo del proyecto dentro de la asignatura	30
10.4. Valoraciones subjetivas	30
A. Currículums Vitae	31
A.1. CV: Daniel García Páez	31
A.2. CV: Javier Briz Alastrué	32
A.3. CV: Simón Ortego Parra	33
A.4. CV: Alejandro Gracia Mateo	34
A.5. CV: Alberto Berbel Aznar	35
A.6. CV: Héctor Francia Molinero	36
B. Hojas de esfuerzos	37
B.1. Hoja de esfuerzos: Daniel García Páez	37
B.2. Hoja de esfuerzos: Javier Briz Alastrué	37
B.3. Hoja de esfuerzos: Simón Ortego Parra	37
B.4. Hoja de esfuerzos: Alejandro Gracia Mateo	37
B.5. Hoja de esfuerzos: Alberto Berbel Aznar	37
B.6. Hoja de esfuerzos: Héctor Francia Molinero	37
C. Cronogramas	38
C.1. Cronograma inicial	38
C.2. Cronograma final	39
D. Presupuesto	40
E. Informes de pruebas	41

1. INTRODUCCIÓN

1.1. Identificación del proyecto

Identificamos el proyecto y su funcionalidad con el siguiente título:

Página web para la venta de microcontroladores a través de un catálogo electrónico.

Además del título, se ofrece la posibilidad de identificar y referenciar el proyecto a través del siguiente código:

ps-14-e20-usearch

Donde:

- “ps” hace referencia a “Proyecto Software”
- “14” hace referencia al año 2014
- “e20” hace a referencia al equipo del proyecto, “Equipo 20”
- “usearch” hace referencia al nombre del proyecto, “*μSearch*”

1.2. Objetivo y alcance

La presente aplicación tiene como objetivo dar solución al problema planteado por la empresa cliente. El cliente nos ha transmitido la necesidad de diseñar un catálogo electrónico de los microcontroladores que distribuye, que permita la gestión de los productos disponibles, y que permita a los clientes realizar búsquedas y efectuar pedidos.

1.3. Identificación del equipo

El proyecto será llevado a cabo por un equipo de 6 personas, compuesto por:

- Alberto Berbel Aznar
- Javier Briz Alastrué
- Héctor Francia Molinero
- Daniel García Páez
- Alejandro Gracia Mateo
- Simón Ortego Parra

El equipo cuenta con experiencia en:

- El mantenimiento de administración de sistemas.
- El desarrollo y mantenimiento de sistemas web utilizando diferentes lenguajes como HTML, PHP, CSS, JavaScript.
- La administración de gestores de contenidos como Drupal, Wordpress...
- El desarrollo y mantenimiento de bases de datos.
- Montar web con funcionalidades similares con gestión de contenido como películas o series.

- El ámbito de la web.
- Entornos cliente-servidor.
- Creación de interfaces web.

Se puede encontrar un mayor detalle sobre la formación de cada miembro del equipo en el anexo **A** de Curriculum Vitae.

1.4. Descripción del contenido

En la presente memoria se detallan todos los aspectos referentes a la aplicación y al proceso de su desarrollo.

Se detallan los requisitos que el cliente plateó inicialmente para el sistema, la descripción técnica de la solución adoptada, las pruebas que se han realizado para verificar que los requisitos se cumplen, los manuales generados, tanto el manual técnico como el de usuario.

También se detallan en esta memoria todas las actividades llevadas a cabo en el proyecto, incluidas las reuniones y sus actas y otra documentación generada en ellas, así como las configuraciones que se han utilizado en el proyecto.

Finalmente se explican las prácticas llevadas a cabo con el fin de asegurar la calidad de la aplicación y se concluye este documento comentando las lecciones aprendidas en el transcurso del proyecto y las conclusiones extraídas del mismo.

Se anexan, con la intención de facilitar la lectura de la memoria, los curriculum vitae de los miembros del equipo, las hojas de esfuerzo de cada uno de ellos, y otros documentos que aportan información adicional a secciones anteriormente mencionadas.

2. REQUISITOS DEL SISTEMA

Se presentan en este apartado los requisitos funcionales con los que el sistema debe cumplir:

RF-1 Un microcontrolador (elemento) estará compuesto de los siguientes campos:

1. Referencia (será única para cada elemento).
2. Arquitectura
3. Frecuencia (MHz)
4. Flash (KB)
5. RAM (KB)
6. Precio (Euros)

RF-2 Insertar un elemento en el carro de compra.

RF-3 Eliminar un elemento del carro de compra.

RF-4 Modificar un elemento del carro de compra. Por modificar se entiende alterar el número de unidades de los elementos.

RF-5 Se podrá generar en cualquier momento un listado de todos los elementos del catálogo.

RF-6 Se podrá actualizar varios elementos del carro de manera simultánea. Por actualizar se entiende a recalcular los precios de cada artículo en el caso de que éstos hayan sido modificados.

RF-7 Se podrá en cualquier momento realizar búsquedas de productos en base a un único campo de búsqueda (una y sólo una de las características de un elemento).

RF-8 Los resultados de la búsqueda se presentarán como un listado (sin paginación) que mostrará, de cada elemento, todos sus campos en columnas.

RF-9 Los listados de elementos del catálogo estarán ordenados en base al campo arquitectura del elemento.

RF-10 Se permitirá realizar pedidos. Cada vez que se realice un pedido se le pedirá al cliente la introducción de sus datos personales. Es decir, no existirá persistencia de los datos del cliente tras realizar pedidos.

RF-11 Los pedidos contendrán la suficiente información para identificar a los clientes. Además, no permitirán la reserva de los productos solicitados, únicamente generarán un presupuesto con el coste de los productos elegidos.

RF-12 Los datos solicitados del cliente para los pedidos serán los siguientes:

1. Nombre
2. Apellidos
3. Dirección
4. Ciudad
5. Provincia
6. País
7. Código postal
8. Teléfono
9. Correo electrónico
10. CIF y Empresa aparecerán como campos opcionales que servirán de distinción entre particulares y entidades.

RF-13 Se contará con una vista diferente para la administración del catálogo. Pudiendo un administrador de la empresa realizar las siguientes acciones:

- Insertar un nuevo elemento en el catálogo.
- Eliminar un elemento del catálogo.
- Modificar un elemento del catálogo (cambiar cualquiera de sus características).
- Generar en cualquier momento un listado completo de todos los elementos del catálogo.
- Realizar en cualquier momento búsquedas de productos en base a un único campo de búsqueda (una y sólo una de las características de un elemento).

3. DESCRIPCIÓN TÉCNICA

La solución técnica que se ha dado con el desarrollo de este proyecto está basada en tecnologías web, capaces de resolver tanto los requisitos de interacción de la aplicación con el usuario como los problemas relacionados con tratamiento y persistencia interna de la información. Concretamente, se utilizará una interfaz web compatible con las últimas versiones de los navegadores más utilizados (más adelante se detallará esto), y se utilizará el framework CodeIgniter, en lenguaje PHP, como base del proyecto. Para el almacenamiento de la información se utilizará una base de datos MySQL. Para evitar problemas de latencias con la base de datos, y dado el reducido tamaño del sistema, se optará por alojar la base de datos y todo el resto del sistema (servidor web e intérprete PHP) en un mismo servidor.

Las tecnologías, lenguajes y aplicaciones utilizadas en el desarrollo del proyecto han sido:

- HTML 5
- CSS 3
- PHP 5
- CodeIgniter 2.1.4
- MySQL 5.5

Se ha asegurado que la web renderice de forma correcta en los siguientes navegadores:

- Google Chrome >=30
- Internet Explorer >=10
- Mozilla Firefox >=27
- Opera >=12

La documentación y manuales de usuario se entregaran al cliente en formato PDF.

3.1. Aspectos arquitecturales y tecnológicos

El patrón que utilizamos para el diseño arquitectural de nuestro catálogo electrónico es el de Modelo-Vista-Controlador, en concreto, la variante Modelo-Vista-Presentador del mismo (véase la figura 1).

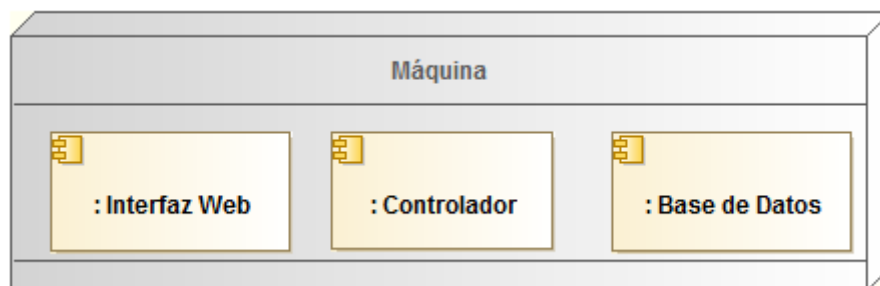


Figura 1: Diagrama de despliegue del sistema

Los componentes de este diseño arquitectural son:

Interfaz web. Es el componente que representa la vista de nuestra aplicación. Es el componente con el que los usuarios interactúan directamente para navegar por la aplicación y visualizar los resultados que producen las interacciones que realicen con la vista. Las acciones del usuario que impliquen el acceso a los datos del modelo o la modificación/eliminación de los mismos son delegadas al componente Controlador.

Controlador. Es nuestro componente Presentador, tiene toda la lógica de la vista y es responsable de sincronizar el modelo y la vista. Es decir, cuando la vista notifica al Presentador que el usuario ha realizado alguna acción (por ejemplo, hacer clic en un botón) que afecta de alguna manera al modelo de datos del sistema, entonces el presentador se encarga de actualizar los datos pertinentes y sincronizar los cambios entre el modelo y la vista.

Base de datos. Es nuestro componente modelo, se encarga de encapsular los datos y ofrecer operaciones para su acceso y procesamiento, es decir, ofrece persistencia de datos para la aplicación. Sólo el componente Controlador interactúa con el modelo de datos..

Las tecnologías utilizadas son las que se detallaron en el apartado anterior.

3.2. Modelos de datos

Nuestro modelo de datos es muy sencillo puesto que solo necesitamos almacenar en la base de datos la información relacionada con los microcontroladores. El resto de la información que manejamos no se almacena en nuestra base de datos; los microcontroladores que introduce un usuario en el carrito de la compra, los guardamos temporalmente utilizando las funciones de PHP; y la información asociada a un cliente que realiza una compra, no se almacena en ningún sitio puesto que solo queda reflejada en la factura que se genera cuando se solicita un pedido.

Por lo tanto, nuestro modelo de datos es el siguiente:

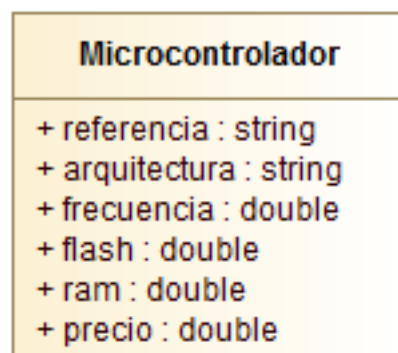


Figura 2: Modelo de datos

3.3. Diseño de los componentes

Los componentes relevantes del sistema son los que ya han sido explicados en el apartado de Aspectos Arquitecturales y Tecnológicos, propios del patrón de diseño de Modelo-Vista-Presentador. Hemos elegido este patrón de diseño porque consideramos que es un patrón que se ajusta perfectamente al problema que tenemos que resolver y a su vez ofrece una solución lo más sencilla posible.

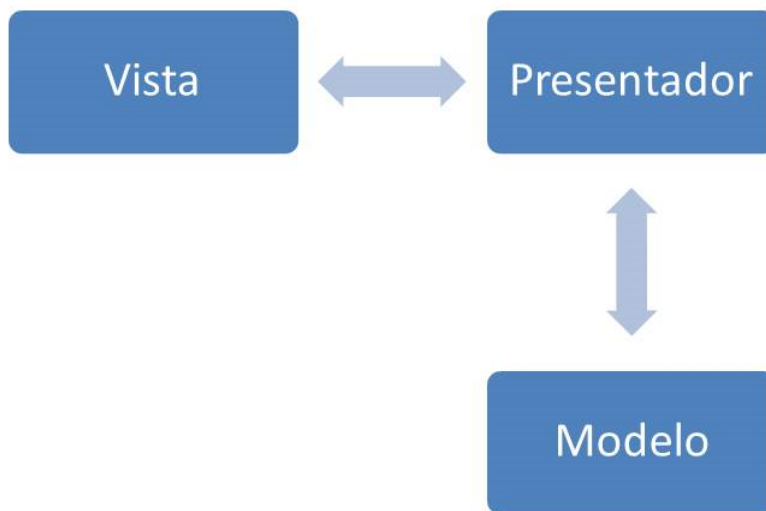


Figura 3: Componentes relevantes del sistema

4. VERIFICACIÓN Y VALIDACIÓN

Se han utilizado las técnicas de verificación y validación conocidas por Alberto Berbel, nuestro experto de equipo en este apartado. Se han reutilizado además herramientas y procedimientos utilizados en la verificación y validación de proyectos anteriores, pues se consideraron interesantes por las características de nuestro nuevo proyecto al ser de índoles similares.

4.1. Metodología de pruebas

Puesto que para la realización de las pruebas hemos utilizado nuestros conocimientos previos adquiridos en la aplicación de Validación y Verificación en otros proyectos, para no desviarnos de lo conocido, nuestra metodología de pruebas se aproxima a la metodología de pruebas utilizada en dichos proyectos anteriores: TMAP Next (Test Management Approach).

Los fundamentos de TMAP se basan en cuatro elementos esenciales:

- **Proceso dirigido por el negocio.** La economía marca el esfuerzo a realizar, y cuáles son los riesgos prioritarios.
- **Un proceso de pruebas estructurado.** Nos guía a la hora de responder a las cuestiones típicas de *qué/cuándo, cómo, con qué y quién* (ciclo de vida de pruebas).
- **Un kit de herramientas.** Se ofrece información práctica para establecer la infraestructura (*con qué*), las técnicas (*cómo*), y la organización (*quién*).
- **Método completo y adaptable.** Flexibilidad para adaptar la metodología a distintas situaciones de desarrollo: nuevos desarrollos, mantenimiento, desarrollo propio o basado en software comercial, ...

Pasos a dar en la gestión de pruebas dirigida por el negocio:

1. Identificar los objetivos de las pruebas
2. Determinar los riesgos
3. Determinar si una característica/parte se debe probar de forma detallada o ligera
4. Estimar y planificar
5. Elegir las técnicas de prueba y ejecutarlas
6. Informar sobre el progreso, calidad

4.2. Pruebas. Defectos. Correcciones

Durante el proceso de desarrollo de nuestro proyecto, los desarrolladores han ido probando que cada método que iban implementando cumplía sus funciones y satisfacía las precondiciones y postcondiciones para las que han sido creados. Sin embargo, hemos considerado que no era necesario documentar este tipo de pruebas puesto que las personas que nos han encargado el proyecto no las necesitaban, tan solo las pruebas de sistema. Además, preparar y documentar debidamente las pruebas unitarias supone un elevadísimo coste de recursos (humanos y temporales). Por lo tanto, se puede decir que se han realizado pruebas unitarias para asegurar el correcto funcionamiento del código implementado pero no hay constancia física de ellas.

Para la realización de las pruebas de sistema, nos hemos centrado en comprobar que nuestro sistema cumple las características que nos comprometimos a desarrollar antes de comenzar este proyecto, cuando conocimos las necesidades que tenía nuestro cliente y quedaron reflejadas en la propuesta del proyecto en forma de requisitos funcionales y no funcionales del sistema.

Las pruebas de sistema que hemos realizado han sido las siguientes:

■ **Primera iteración, funcionalidades de administración.**

- **Comprobación de la inserción de un nuevo elemento al catálogo.** Queremos comprobar que se introduce correctamente un nuevo elemento al catálogo y al volver a cargar el listado de elementos disponibles aparece el nuevo elemento introducido. Desde que se realizó esta prueba por primera vez, los resultados han sido los esperados.
- **Comprobación de la inserción de un nuevo elemento al catálogo sin rellenar alguno de los campos.** Queremos comprobar que el sistema no nos permite introducir un elemento incompleto al catálogo, pues todos sus campos deberían ser obligatorios. La primera vez que se realizó esta prueba, el sistema sí introducía el elemento incompleto y por lo tanto fue necesario cambiar el código implicado en esta funcionalidad. Ahora, si se intenta introducir un elemento incompleto en el catálogo, aparece un mensaje informando de que todos los campos son obligatorios y el nuevo elemento no es insertado en el catálogo.
- **Comprobación de la modificación de un elemento existente en el catálogo.** Queremos comprobar que el sistema permite modificar las características de un elemento existente en el catálogo. Queremos comprobar que el sistema permite modificar las características de un elemento existente en el catálogo. Para ello se ejecutan pruebas modificando sus diferentes características (Arquitectura, Frecuencia, Flash, RAM y Precio) de manera combinada: de dos en dos, un sólo campo, todos los campos, etc. Desde la primera vez que se ejecutó esta prueba, los resultados han sido los esperados.
- **Comprobación de la eliminación de un elemento existente en el catálogo.** Queremos comprobar que el sistema permite eliminar un elemento existente en el catálogo y que por lo tanto al volver a cargar el listado del catálogo, el elemento eliminado ya no aparece en el listado. Desde la primera vez que se ejecutó esta prueba, los resultados han sido los esperados.
- **Comprobación del listado de los elementos del catálogo.** Queremos comprobar que el sistema muestra todos los elementos existentes en el catálogo, desde el primero hasta el último. Desde la primera vez que se ejecutó esta prueba, los resultados han sido los esperados.

■ **Primera iteración, funcionalidades propias del usuario.**

- **Comprobación del listado de los elementos del catálogo.** Queremos comprobar que un usuario tiene acceso al listado de todos los elementos existentes en el catálogo, desde el primero hasta el último. Desde la primera vez que se ejecutó esta prueba, los resultados han sido los esperados.

- **Comprobación de los datos que son solicitados al cliente para generar la factura con los elementos del carro de compra.** Queremos comprobar que los datos que son solicitados al cliente son los mismos que los que acordamos con el cliente en los requisitos funcionales del sistema. Desde la primera vez que se ejecutó esta prueba, los resultados han sido los esperados.
 - **Comprobación de la generación de la factura si el cliente no ha introducido cualquiera de los campos obligatorios.** Queremos comprobar que todos los datos requeridos al cliente deben ser introducidos para la generación de la factura. Que no permita generar la factura si hay algún campo vacío. La primera vez que se ejecutó esta prueba, el sistema sí generaba la factura aunque quedara algún campo sin completar. Por lo tanto, fue necesario modificar el código implicado en esta funcionalidad, y ahora el resultado que se produce es que el sistema muestra un mensaje indicando que todos los campos son obligatorios.
- **Segunda iteración, funcionalidades de administración.**
- **Comprobación de la inserción de un nuevo elemento al catálogo.** Queremos comprobar que se introduce correctamente un nuevo elemento al catálogo y al volver a cargar el listado de elementos disponibles aparece el nuevo elemento introducido. Igual que en la primera iteración, los resultados han sido los esperados.
 - **Comprobación de la inserción de un nuevo elemento al catálogo sin rellenar alguno de los campos.** Queremos comprobar que el sistema no nos permite introducir un elemento incompleto al catálogo, pues todos sus campos deberían ser obligatorios. Como ya se corrigió durante la primera iteración, los resultados obtenidos son los esperados.
 - **Comprobación de la modificación de un elemento existente en el catálogo.** Queremos comprobar que el sistema permite modificar las características de un elemento existente en el catálogo. Para ello se ejecutan pruebas modificando sus diferentes características (Arquitectura, Frecuencia, Flash, RAM y Precio) de manera combinada: de dos en dos, un sólo campo, todos los campos, etc. Igual que en la primera iteración, los resultados han sido los esperados.
 - **Comprobación de la eliminación de un elemento existente en el catálogo.** Queremos comprobar que el sistema permite eliminar un elemento existente en el catálogo y que por lo tanto al volver a cargar el listado del catálogo, el elemento eliminado ya no aparece en el listado. Igual que en la primera iteración, los resultados han sido los esperados.
 - **Comprobación del listado de los elementos del catálogo.** Queremos comprobar que el sistema muestra todos los elementos existentes en el catálogo, desde el primero hasta el último. Igual que en la primera iteración, los resultados han sido los esperados.

- **Comprobación de la búsqueda exitosa de elementos del catálogo por cualquiera de los campos de búsqueda.** Queremos comprobar que el sistema es capaz de realizar búsquedas en el catálogo con cada criterio de búsqueda (Arquitectura, Frecuencia, Flash, RAM). Es decir, que devuelve como resultado el listado de todos los elementos del catálogo que cumplen el criterio de búsqueda. Los resultados obtenidos en esta prueba son los esperados.
 - **Comprobación de la búsqueda fallida de elementos del catálogo por cualquiera de los campos de búsqueda.** Queremos comprobar que el sistema es capaz de gestionar los casos en los que las búsquedas con cada criterio de búsqueda (Arquitectura, Frecuencia, Flash, RAM) no sean satisfactorias e informe al usuario de que no hay elementos en el catálogo que coincidan con su criterio de búsqueda. Los resultados obtenidos en esta prueba son los esperados.
- **Segunda iteración, funcionalidades propias del usuario.**
- **Comprobación del listado de los elementos del catálogo.** Queremos comprobar que un usuario tiene acceso al listado de todos los elementos existentes en el catálogo, desde el primero hasta el último. Igual que en la primera iteración, los resultados han sido los esperados.
 - **Comprobación de la búsqueda exitosa de elementos del catálogo por cualquiera de los campos de búsqueda.** Queremos comprobar que el sistema es capaz de realizar búsquedas en el catálogo con cada criterio de búsqueda (Arquitectura, Frecuencia, Flash, RAM). Es decir, que devuelve como resultado el listado de todos los elementos del catálogo que cumplen el criterio de búsqueda. Los resultados obtenidos en esta prueba son los esperados.
 - **Comprobación de la búsqueda fallida de elementos del catálogo por cualquiera de los campos de búsqueda.** Queremos comprobar que el sistema es capaz de gestionar los casos en los que las búsquedas con cada criterio de búsqueda (Arquitectura, Frecuencia, Flash, RAM) no sean satisfactorias e informe al usuario de que no hay elementos en el catálogo que coincidan con su criterio de búsqueda. Los resultados obtenidos en esta prueba son los esperados.
 - **Comprobación de los datos que son solicitados al cliente para generar la factura con los elementos del carro de compra.** Queremos comprobar que los datos que son solicitados al cliente son los mismos que los que acordamos con el cliente en los requisitos funcionales del sistema. Desde la primera vez que se ejecutó esta prueba, los resultados han sido los esperados.
 - **Comprobación de la generación de la factura si el cliente no ha introducido cualquiera de los campos obligatorios.** Queremos comprobar que todos los datos requeridos al cliente deben ser introducidos para la generación de la factura. Que no permita generar la factura si hay algún campo vacío. Como ya se corrigió durante la primera iteración, los resultados obtenidos son los esperados.

Se puede acceder en el anexo E al informe completo y detallado de las pruebas.

5. MANUALES

En este apartado se presentan los manuales de usuario necesarios para que tanto el usuario cliente de la aplicación como el administrador de la misma aprendan a utilizarla y navegar por ella de la forma más rápida y sencilla posible.

Además se añade un manual de instalación para que, en caso de que la empresa cliente a la que va dirigida este producto decida mantener ella el catálogo electrónico, pueda instalar y preparar el entorno necesario para ello.

5.1. Manual de usuario

El manual de usuario, incluye dos versiones conjuntas en un mismo documento:

- La del usuario cliente que utilizará la aplicación para navegar por la misma y realizar sus compras de microcontroladores
- La del usuario administrador de la empresa cliente que tendrá acceso interno al catálogo electrónico para añadir, modificar y eliminar microcontroladores del mismo.

Se puede acceder a él a través del [siguiente enlace](#).

5.2. Manual de instalación

Dado que nuestra empresa, **BitParty**, ofrece el servicio de encargarse de lanzar y poner en funcionamiento el catálogo electrónico y del mantenimiento del servidor y la base de datos que permitirán a la aplicación estar totalmente operativa *online* y disponible 24 horas al día, el cliente no deberá preocuparse por como instalar la aplicación, ni poner en marcha un servidor, etc...

Así pues en un principio el cliente sólo deberá preocuparse de abastecernos con la información completa de su catálogo de microcontroladores.

Aún así, si en un futuro el cliente se ve capacitado para realizar dichas tareas de lanzamiento y mantenimiento de la aplicación, gustosamente se le generará y entregará un manual de instalación para facilitarle dicha tarea.

6. GESTIÓN DEL PROYECTO

En esta parte se ha planificado, organizado, monitorizado y controlado el proyecto. El objetivo de todo esto es disminuir los costes, satisfacer las fechas de entrega, conseguir un buen producto y prever adecuadamente los posibles problemas que puedan surgir a lo largo de todo el proceso.

6.1. Fases y actividades

En esta sección se explican las diferentes fases en las que se ha dividido y por las que ha pasado el proyecto, cómo se han planificado, qué tareas se han realizado en cada una de ellas y que esfuerzos globales han supuesto llevarlas a cabo.

6.1.1. Riesgos

Se han identificado, definido y clasificado los posibles riesgos que el equipo se podía encontrar durante el desarrollo del proyecto; estableciendo además las consiguientes estrategias de respuesta y mitigación para cada uno de ellos. En el [siguiente enlace](#) se puede acceder al catálogo donde se encuentran.

6.1.2. Estimaciones iniciales

Se muestra en la figura 4 los tiempos que se han asignado a las diferentes tareas. En esta figura se muestran las horas concretas que se estimaron al principio del proyecto. Más adelante se mostrara cuantas horas reales se invirtieron en cada apartado.

	Estimadas	Estimadas con corrección
TAREA 0: Lanzamiento del proyecto	90	144
TAREA 1: Implementación vista web básica	12	19
TAREA 2: Implementación control añadir/modificar/eliminar elemento	8	13
TAREA 3: Implementación control listado de elementos	8	13
TAREA 4: Implementación control carrito de compra	10	16
TAREA 5: Implementación control generación factura en texto plano	4	6
TAREA 6: Implementación control para búsquedas	10	16
TAREA 7: Implementación control generación factura en PDF	10	16
TAREA 8: Mejora implementación interfaz web	20	32
Cierre de proyecto	18	29
Población de la BD	3	5
Documentación Técnica (documentación de código y Wiki)	14	22
Documentación de Manual y Guía de Usuario	18	29
Pruebas	40	64
Gestión		63
Gestión de configuraciones		21
Aseguramiento de la calidad		30
TOTAL	265	538

Figura 4: Estimación esfuerzos totales

Se muestra en la figura 5 las horas asignadas a cada una de las tareas. Se puede ver que las partes más costosas son sobre todo el lanzamiento del proyecto, gestión y pruebas. Implementación se llevaría otra de las grandes partes del proyecto, pero al estar dividida en subtareas parece que ocupe un tiempo menor del que en realidad lleva.

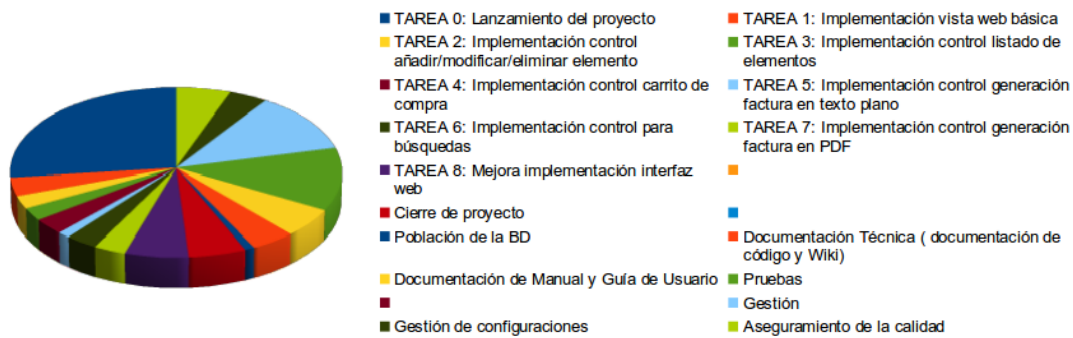


Figura 5: Gráfica esfuerzos totales

6.1.3. Cronogramas

Se pueden visualizar, tanto el cronograma que se planificó inicialmente y como el cronograma que resultó finalmente, en el anexo C de este documento.

En general las tareas se retrasaron un poco frente a la planificación prevista, tanto su comienzo como su finalización. Pero siempre se ha ido desarrollando el proyecto con los tiempos previstos y cumpliendo los objetivos adecuadamente. Alguna de las tareas se pudo alargar algo más en el tiempo que lo mostrado en el cronograma, pero debido siempre a pequeñas modificaciones de última hora o detalles olvidados que no eran relevantes.

6.1.4. Tareas y estimación de esfuerzos

Se muestran los esfuerzos previstos para la primera iteración del proyecto en la figura 6.

	Estimadas	Estimadas con corrección
TAREA 0: Lanzamiento del proyecto	90	144
TAREA 1: Implementación vista web básica	12	19
TAREA 2: Implementación control añadir/modificar/eliminar elemento	8	13
TAREA 3: Implementación control listado de elementos	8	13
TAREA 4: Implementación control carrito de compra	10	16
TAREA 5: Implementación control generación factura en texto plano	4	6
		0
Población de la BD	3	5
Documentación Técnica (documentación de código y Wiki)	6	10
Documentación de Manual y Guía de Usuario	6	10
Pruebas	20	32
Gestión		40
Gestión de configuraciones		13
Aseguramiento de la calidad		19
TOTAL	167	339,2

Figura 6: Esfuerzos primera iteración

Se puede apreciar como la parte más importante de la primera iteración fue el lanzamiento, en el cual se planifico todo el proyecto para concretar todo e intentar evitar posibles problemas. Entre las demás partes se planifico que pruebas, gestión y implementación serías las siguientes tareas más costosas. Esto se puede ver en la figura 7.

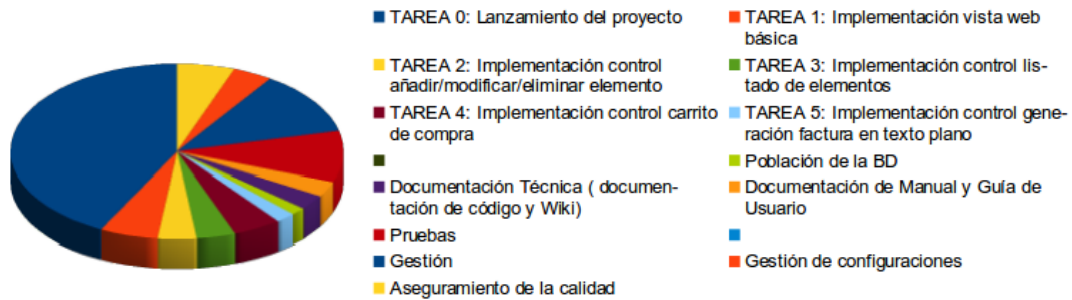


Figura 7: Gráfica esfuerzos primera iteración

Se muestran los esfuerzos previstos para la segunda iteración del proyecto en la figura 8. Podemos observar que la mayor parte de horas se la llevan la mejora de la implementación de la interfaz web (con la intención de que la misma tenga un aspecto llamativo y atractivo para el cliente) y las pruebas (con la intención de asegurar completamente que el producto que se le entrega al cliente funciona correctamente en su totalidad). Podemos ver dicho reparto de trabajo más claramente en la figura 9.

	Estimadas	Estimadas con corrección
TAREA 6: Implementación control para búsquedas	10	16
TAREA 7: Implementación control generación factura en PDF	10	16
TAREA 8: Mejora implementación interfaz web	20	32
Documentación Técnica (documentación de código y Wiki)	8	13
Documentación de Manual y Guía de Usuario	12	19
Pruebas	20	32
Cierre del proyecto	18	29
Gestión		23
Gestión de configuraciones		8
Aseguramiento de la calidad		11
TOTAL	98	198,8

Figura 8: Esfuerzos segunda iteración

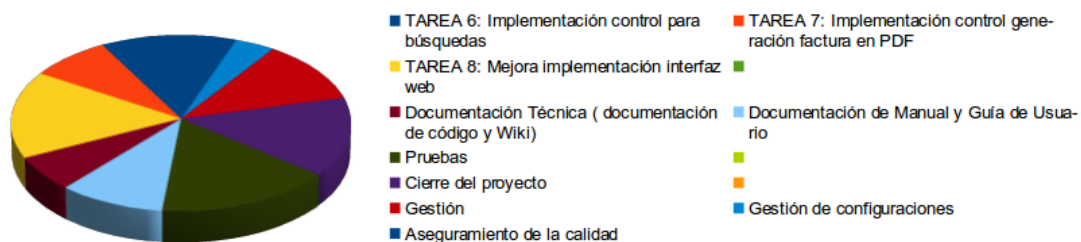


Figura 9: Gráfica esfuerzos segunda iteración

6.1.5. Ficheros de esfuerzos individuales

Las hojas de esfuerzos individuales para cada uno de los miembros del equipo se encuentran disponibles en el Anexo B de este documento.

6.1.6. Esfuerzos reales de las tareas

Para mostrar los esfuerzos reales de cada iteración se han realizado varias gráficas.

Como se aprecia en la figura 10 el mes de más trabajo fue Marzo. Además se muestran las horas invertidas en el proyecto y concretamente en cada una de las tareas. En total se realizaron 274 horas frente a las 339 previstas. Pero estas 339 son con la corrección aplicada al desconocimiento de las mismas al inicio del proyecto. Sin esa corrección se estimaron 212 por lo que se han mantenido las horas entre ambas.

	Febrero	Marzo	Abril	Mayo	Total	Estimadas	Estimadas con corrección	Porcentaje completado
Lanzamiento de proyecto	22	64	0	0	86	90	144	100,00%
					0		0	
Implementación vista	0	15	9		24	12	19	100,00%
Implementación modelo y controlador	0	22	6		28	30	48	100,00%
Población BD	0	3	5		8	3	4	100,00%
Documentación técnica	0	3	6		9	6	9	100,00%
Documentación guía de usuario	0	0	5		5	6	9	100,00%
Pruebas	0	2	13		15	20	32	100,00%
					0		0	
					0		0	
Reuniones	0	8	21		29		0	
Aprendizaje	19	6	2		27		0	
Otros	29	10	4		43		0	
TOTAL					274	212,09	339	

Figura 10: Esfuerzos reales primera iteración

En la figura figura 11 se muestra lo anterior más gráficamente. La ultima columna sería el global de la primera iteración y las demás las subtareas.

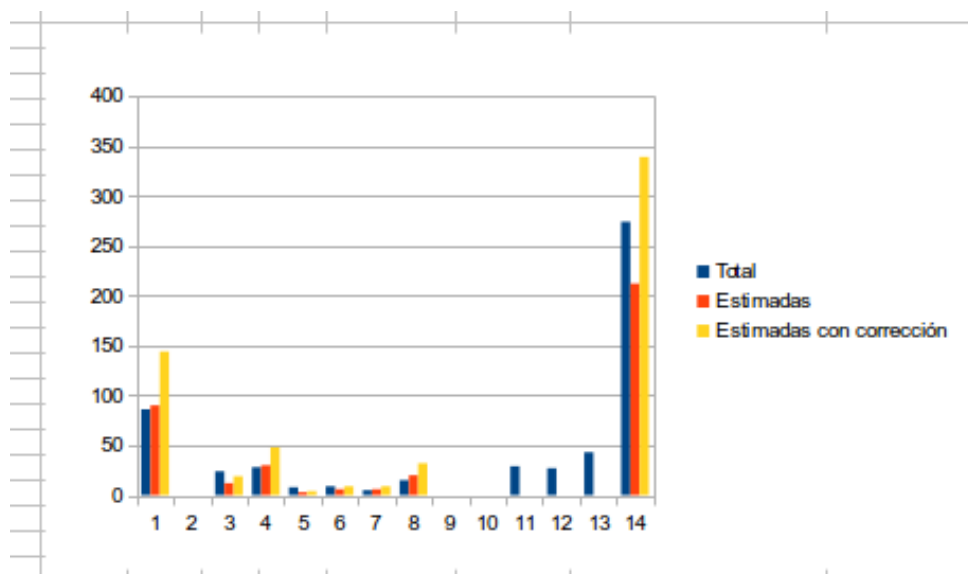


Figura 11: Esfuerzos reales primera iteración

Para la segunda iteración se muestran la misma gráficas utilizadas para el análisis de la primera iteración pero con las horas invertidas en la segunda iteración (figura 12 y figura 13).

	Mayo	Total	Estimadas	Estimadas con corrección	Porcentaje completado
Implementación vista	27	27	20	32	100,00%
Implementación modelo y controlado	20	20	20	32	100,00%
Documentación técnica	13	13	8	12,8	100,00%
Documentación guía de usuario	8	8	12	19,2	100,00%
Pruebas	20	20	20	32	100,00%
Cierre del proyecto	35	35	18	28,8	100,00%
		0		0	
Reuniones	14	14		23	
Aprendizaje		0		8	
Otros	22	22		11	
TOTAL		159	98	198,8	

Figura 12: Esfuerzos reales segunda iteración

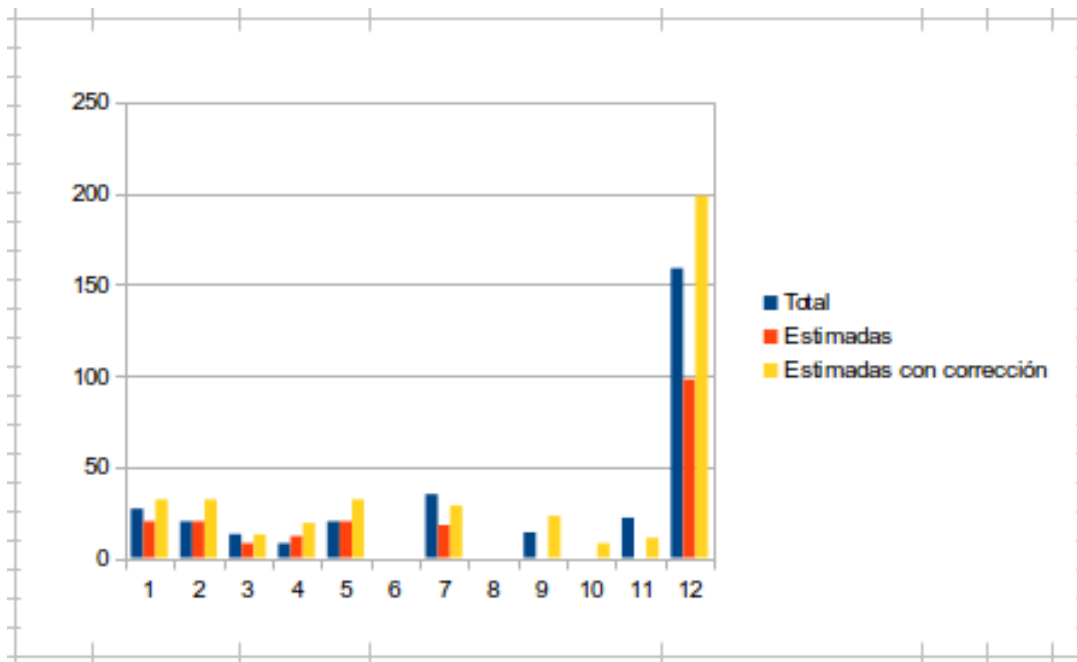


Figura 13: Esfuerzos reales segunda iteración

6.1.7. Esfuerzos reales de las personas

Se muestran los datos y las gráficas con las horas realizadas por las diferentes personas del equipo en las figuras figura 14 y figura 15.

Los roles de los miembros del grupo son los siguientes:

- Daniel ⇒ Director de proyecto
- Alberto ⇒ Verificación y validación
- Javier ⇒ Gestor de configuraciones
- Héctor ⇒ Gestor de calidad
- Simón ⇒ Gestor de desarrollo
- Alejandro ⇒ Gestor de planificación

En la gráfica figura 14 se ven las diferentes horas invertidas por los integrantes del grupo. Algún aspecto a destacar es que Daniel, como director del proyecto, invirtió el que más horas en el mismo. Además el mes de Marzo, como ya se ha recalcado anteriormente, fue en el que más horas se invirtieron, es decir, ese mes fue el grueso del proyecto. También podemos observar como Javier, al ser el principal encargado de la implementación de la aplicación, tenía una carga de trabajo inicial menor pero después mientras el resto del grupo bajaba el nivel de trabajo él lo subía.

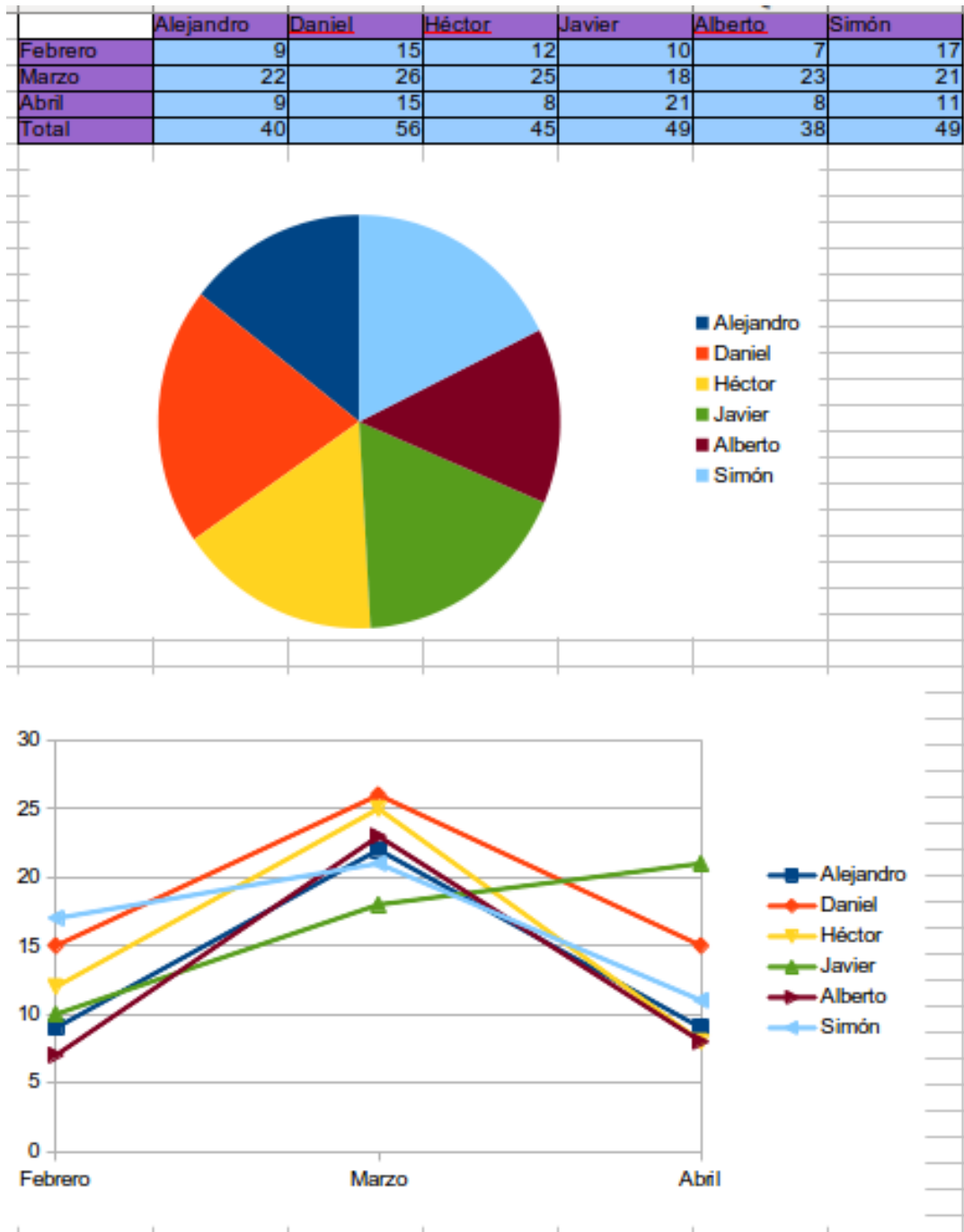


Figura 14: Esfuerzos reales por persona primera iteración

Se presentan a continuación los datos para la segunda iteración. De nuevo el director, Daniel, invirtió el que más horas en el proyecto y los demás integrantes del grupo más o menos hicieron las mismas horas de trabajo. Datos que representan una buena organización y reparto de trabajo.

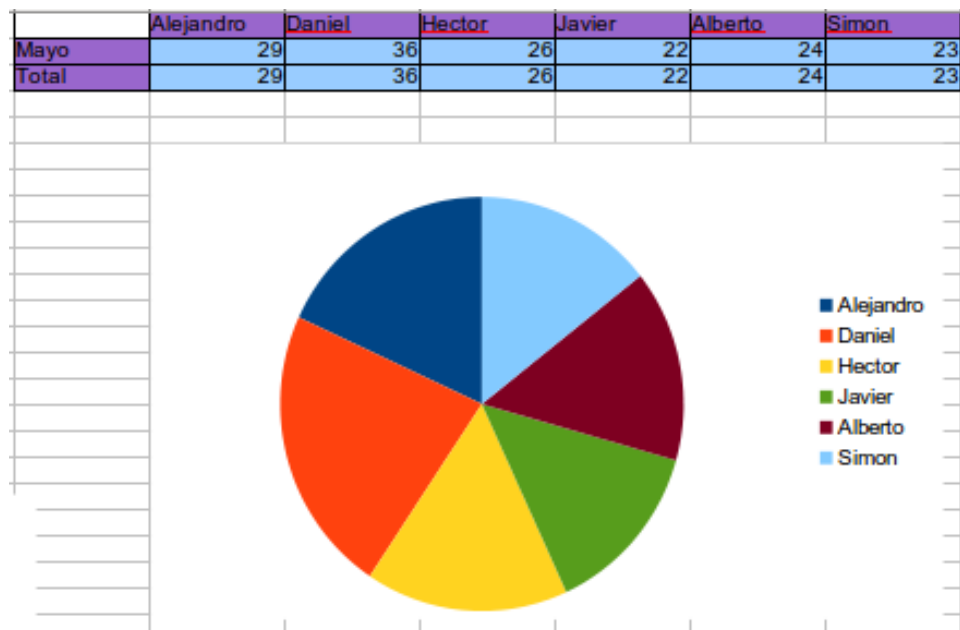


Figura 15: Esfuerzos reales por persona primera iteración

6.2. Procesos de seguimiento y control

Para el seguimiento y control del estado en el que se encontraba el desarrollo de nuestro proyecto se han ido realizando reuniones de todo los miembros del grupo cada semana o cada dos semanas. En ellas se analizaba el trabajo realizado hasta ese momento, se planificaban y repartían nuevas tareas y se resolvían posibles dudas o problemas que pudieran surgir.

6.2.1. Calendario de reuniones

Se muestra a continuación el calendario de las reuniones que el equipo ha realizado durante el desarrollo el proyecto. Cada reunión tiene identificador propio, nombre, fecha y hora:

- R-01 Lanzamiento de equipo: 20/02/2014 - 10:00
- R-02 Lanzamiento de proyecto: 20/02/2014 - 12:00
- R-03 Lanzamiento de proyecto: 26/02/2014 - 20:00
- R-04 Seguimiento de proyecto y 1ª iteración: 06/03/2014 - 10:00
- R-05 Práctica 3, Comienzo primera iteración: 06/03/2014 - 12:00
- R-06 Reunión de seguimiento semanal: 10/03/2014 - 20:00
- R-07 Reunión interna de seguimiento: 12/03/2014 - 20:00
- R-08 Reunión de seguimiento: 27/03/2014 10:00
- R-09 Reunión para Aseguramiento de la Calidad del proyecto: 27/03/2014 -12:00
- R-10 Reunión de seguimiento: 09/04/2014 -20:00
- R-11 Reunión final de 1ª iteración y comienzo de la 2ª: 09/04/2014 - 20:00

6.2.2. Actas de reuniones

Se muestran a continuación los enlaces directos a las actas de cada una de las reuniones

- [R-01 Lanzamiento de equipo](#)
- [R-02 Lanzamiento del proyecto](#)
- [R-03 Lanzamiento del proyecto II](#)
- [R-04 Seguimiento de la propuesta de proyecto](#)
- [R-05 Comienzo 1ª iteración](#)
- [R-06 Reunión de seguimiento del proyecto](#)
- [R-07 Reunión de seguimiento del proyecto](#)
- [R-08 Reunión de seguimiento del proyecto](#)
- [R-09 Aseguramiento de calidad del proyecto](#)
- [R-10 Reunión de seguimiento del proyecto](#)
- [R-11 Reunión final de 1ª iteración y comienzo de la 2ª](#)

6.3. Coste

Para analizar el coste real del proyecto, se han sumado las horas realizadas en cada una de las iteraciones: 274 horas + 159 horas = 433 horas en total. No se han comprado ni servidores ni han sido necesarios gastos extras.

Por lo que el presupuesto final son 8010,50€. que sumando el IVA son : 9692,2€.

Como se ha realizado el proyecto en menos horas que las estimadas y se aplicó un buen margen de beneficio, se ha conseguido un gran beneficio del proyecto para la empresa que asciende a 6931€.

7. GESTIÓN DE CONFIGURACIONES

7.1. Políticas de nombrado

El nombrado de ficheros debe realizarse en minúsculas, sin espacios, pudiendo utilizar el carácter '_' en su lugar.

El nombre de cada fichero debe finalizar con una extensión separada del nombre del fichero por un punto, siendo la extensión en minúsculas y apropiada al tipo de contenido del fichero (php, png, html, txt, pdf...).

7.2. Control de versiones

El control de versiones utilizado será subversion, utilizándose como servicio de almacenamiento el de Google: Google Code, dada su elevada disponibilidad, siendo de esta manera accesible el proyecto para todos los miembros del equipo.

7.3. Copias de seguridad

Durante el desarrollo del proyecto se ha dispuesto en todo momento de una copia del sistema, operativa, a fin de poder llevar a cabo pruebas con el sistema online. Esta copia ha jugado un doble papel: el de servir como copia de seguridad en caso de no poder acceder a Google Code y el de servir como copia viva del entorno.

7.4. Elementos de configuración y línea base

Durante el desarrollo del proyecto se generarán diversos documentos que conformarán la configuración del proyecto.

En primer lugar se establecerá un marco de trabajo, anteriormente descrito, que establecerá las políticas de nombrado, formatos utilizados, plataforma de control de versiones, y otros elementos que harán posible el trabajo en equipo y el intercambio de información.

El primer documento generado de cara al cliente será la propuesta de proyecto, junto con el presupuesto del mismo. Este documento recoge un plan de proyecto, la definición de requisitos, un primer análisis y una descripción técnica de la propuesta de solución. También se integra en esa primera propuesta la composición del equipo que llevará a cabo el proyecto.

Otros documentos que conforman la configuración del proyecto son los documentos de análisis y pruebas del sistema, el prototipo, los documentos de diseño e imagen del proyecto, el código fuente del programa, los documentos de calidad, los manuales, y otros documentos disponibles en el control de versiones del proyecto.

Se establece como línea base del programa, el resultado de la primera iteración del desarrollo, a partir de la cual se inicia la segunda iteración a cuya finalización se deberían haber completado los requisitos planteados por el cliente.

8. ASEGURAMIENTO DE LA CALIDAD

Para asegurar la calidad de nuestro proyecto nuestros objetivos han sido mejorar el software monitorizándolo apropiadamente y también el proceso de desarrollo que lo produce. De esta manera se busca asegurar la completa concordancia con los estándares y procedimientos establecidos para nuestro software y nuestro proceso de desarrollo. Con ello conseguimos que cualquier elemento que no sea adecuado en el producto, el proceso, o los estándares sea puesto en conocimiento de los responsables para que pueda ser resuelto.

8.1. Estándares utilizados

Como estándar utilizado a lo largo de todo el proceso se ha seguido una **guía de codificación** definida al principio del proyecto y de la cual no nos hemos apartado. Se compone de las siguientes pautas:

- Longitud máxima de línea recomendada, 80 caracteres.
- Métodos/funciones de un máximo recomendado de 30 líneas.
- Nombres de clase en mayúscula.
- Nombres de métodos/funciones en minúscula.
- Tabulado con dos espacios.
- Codificación UTF-8 en todos los ficheros de texto plano.
- Es importante respetar el indentado en el código fuente php/html crudo"pero no se tratará de que el código esté correctamente indentado tras pasar por el intérprete php.

8.2. Planificación de las auditorías

- La **auditoría interna** se planificó para el día 7 de Mayo de 2014 y se estimó una duración de 1 hora. Finalmente se llevó a cabo ese día pero la duración fue de 50 minutos.
- La **auditoría externa** se planificó para el día 9 de Mayo de 2014 y se estimó una duración de 1 hora. Finalmente se llevó a cabo ese día pero la duración fue de 80 minutos.

8.3. Auditorías

Para el seguimiento de la calidad del equipo de desarrollo, se ha comprobado el seguimiento de las bases establecidas al inicio del proyecto con una herramienta clave como son las auditorías.

Éste es el método principal para validar la calidad de un proceso o de un producto ya que realiza un examen de parte o todo de ese sistema y de su documentación para encontrar problemas potenciales.

En el [siguiente enlace](#) se puede encontrar la auditoría interna realizada a nuestro proyecto.

8.4. Auditoría externa

En el [siguiente enlace](#) se puede encontrar la auditoría externa realizada a nuestro proyecto.

8.5. No conformidades

9. POSTMORTEM

Análisis postmortem del desarrollo del proyecto.

9.1. Lecciones aprendidas

Se enumeran aquí alguna de las lecciones aprendidas por los miembros del equipo durante el desarrollo del proyecto:

- Se ha aprendido la importancia del reparto de roles. Si bien la dimensión del proyecto no era quizás lo suficientemente grande para sacarle el máximo beneficio a esto, sí que sin este tipo de organización hubiera sido mucho más difícil llevar todo a cabo. Teniendo cada miembro un papel bien definido dentro del proyecto es mucho más fácil el realizar todas las tareas y juntar el trabajo periódicamente.
- A pesar de tener varios canales de comunicación entre los miembros del equipo (*Whatsapp*, correo electrónico (*Google Groups*)...), es muy importante las reuniones periódicas para poner en común el trabajo. Es decir, en persona y cara a cara es mucho más efectivo trabajar, es muy importante para el desarrollo del proyecto.
- Se ha aprendido la importancia de buscar siempre varias alternativas a los problemas que surgen, poner ideas en común y llegar a un acuerdo como grupo. Por ejemplo, la decisión final de realizar el proyecto implementando la el catálogo con un *Framework* como *CodeIgniter* fue una decisión consensuada y finalmente vital para realizar a tiempo el proyecto. Así como la decisión de utilizar *LaTeX* para la documentación, decisión consensuada también y que ha agilizado mucho todo el desarrollo del proyecto.
- Se ha aprendido la importancia de que los miembros del equipo no sólo se ciñan a su papel, sino que también estén al día como mínimo de otros aspectos del proyecto. Pues si algún día faltaba un miembro o no estaba disponible durante algún tiempo, alguien tenía que suplirle temporalmente. De igual manera, es importante que el director esté al tanto de todo y se interese por el trabajo del resto de miembros del equipo.
- Se ha aprendido la importancia de planificar un proyecto antes de comenzarlo. Sí aún planificándolo ha habido momentos de cierta descordinación, sin la planificación pudiera haber sido un desastre que se ha evitado.
- Se ha aprendido la importancia de utilizar una herramienta de control de versiones como *SubVersion* para desarrollo del proyecto, pudiendo deshacer cambios y errores, teniendo al instante los cambios realizados por otro compañero, etc. Además de aprender a utilizar este tipo de herramientas para futuros proyectos.

En resumen, estás son algunas de las lecciones más importantes aprendidas, pero ha sido de gran importancia el aprender en general como trabajar en grupo con roles definidos y planificando cada aspecto de un proyecto.

9.2. Problemas encontrados

Se enumeran aquí algunos de los problemas que el equipo se ha ido encontrado durante el desarrollo del proyecto:

- Aprendizaje para manejar el control de versiones de *SubVersion*. Hubo problemas al inicio cuando había conflictos de modificación de ficheros.
- Aprendizaje de PHP por el resto de integrantes del equipo, pues sólo teníamos un experto en dicho lenguaje.
- Encontrar días y horas para realizar las reuniones y que todos los integrantes del equipo pudieran acudir.

En general se podría decir que el proyecto se ha ido desarrollando sin muchos problemas. La mayoría de poca importancia y rápida solución, lo cual se puede considerar un punto a favor de la organización del equipo.

9.3. Catálogo de riesgos

Como ya se ha indicado en el punto 6.1.1 de este documento, se han identificado, definido y clasificado los posibles riesgos que el equipo se podía encontrar durante el desarrollo del proyecto; estableciendo además las consiguientes estrategias de respuesta y mitigación para cada uno de ellos. En el [siguiente enlace](#) se puede acceder al catálogo donde se encuentran.

Analizándolos tras la realización del proyecto, realmente se podría decir que sólo hemos tenido que lidiar con la falta de conocimiento o experiencia con la metodología y algunos de los elementos de planificación impuestos para el desarrollo del proyecto; así como con que sólo algunos integrantes del equipo poseían los conocimientos suficientes para desarrollar ciertas partes del proyecto, especialmente la implementación. Pero esto se ha solucionado con el añadido de horas extras por el resto de miembros del equipo para con el máximo esfuerzo adquirir nuevos conocimientos y así mitigar estos riesgos

9.4. Diferencias entre los esfuerzos y tamaños reales y los estimados

9.5. Plan real vs planificación inicial

9.6. Diferencias entre el coste real y el presupuesto

10. CONCLUSIONES

10.1. Conclusiones del proyecto

10.2. Ideas de mejora del proceso

10.3. Ideas de mejora del desarrollo del proyecto dentro de la asignatura

10.4. Valoraciones subjetivas

A. CURRÍCULUMS VITAE

A.1. CV: Daniel García Páez

Curriculum Vitae

Lenguajes de programación utilizados

Conocimiento avanzado de Java

Conocimiento medio-avanzado de HTML,CSS,JSP,XML. Programación Web

Conocimiento deMySQL,SQL

Conocimiento medio de programación Android

Conocimiento medio de C

Conocimiento básico de PHP

Experiencia

Realización de una aplicación web (HTML, CSS, JSP, XML) de red social.

Realización de una aplicación web (HTML, CSS, JSP, XML) de operaciones con clientes y cuentas bancarias.

Creación y mantenimiento de varias bases de datos en MySQL.

Realización de una aplicación móvil de gestión de notas para Android. Incluido todo el proceso de análisis, requisitos, diseño e implementación del proyecto.

Formación

2008 a 2014 Estudiante de la EINA Grado en **Ingeniería informática** en la rama de **Software**

Inglés nivel avanzado(B2) de la escuela de Cambridge.

1 de junio de 2014

A.2. CV: Javier Briz Alastrué**Curriculum Vitae****Lenguajes de programación utilizados**

Java, C, C++, Wiring, Python, Haskell, Ada, Perl, Bash

HTML CSS, JSP, JavaScript

MySQL

Experiencia

- | | |
|-------------------|---|
| 2013 | Co-fundador de Prototyp3D y FaryNozzle. Experiencia en el ámbito de la impresión 3D. |
| 2007 - actualidad | Presidente en Púlsar, Certificado de servicios distinguidos de la Universidad. |
| 2008 - actualidad | Secretario en ISC. |
| 2008 - actualidad | Administrador de sistemas y clusters de computación en el grupo de Fluidodinámica Numérica de la Universidad de Zaragoza. |
| 2008 | Administrador de servidores y estaciones de trabajo en el Área de Mecánica de Fluidos de la Universidad de Zaragoza. |

Formación

- | | |
|-------------|--|
| 2007 - 2014 | Estudiante de la EINA: Grado en Ingeniería informática |
| 2011 | : Certificado en "Software as a Service"(a course of study initiative of The University of California at Berkeley) |
| 2011 | Curso de Administración de Sistemas Linux, Universidad de Barcelona |
| Inglés | Nivel avanzado (B2) en la Escuela Oficial de Idiomas. |

1 de junio de 2014

A.3. CV: Simón Ortego Parra

Curriculum Vitae

Lenguajes de programación utilizados

Java (+Android SDK), C

Ensamblador : ARM (+THUMB), SPARC, Intel, DLXV. OpenMP

Haskell, Erlang

Python, Ruby, sh

HTML, CSS, JSP, XML

SQL

Experiencia

Actualidad	Desarrollo de una aplicación Web para la conversión y visualización de preparaciones histológicas virtuales utilizando pirámides de imágenes (HTML, PHP y Javascript).
2012-actualidad	Creación y mantenimiento de varias bases de datos relacionales (Oracle y MySQL).
2013	Desarrollo de una aplicación Web de películas mediante el uso de HTML junto con JSPs y una base de datos MySQL.
2013	Realización de una aplicación móvil de gestión de notas para Android.
2012	Participación en un curso básico de Latex en la Asociación de Ingenieros en Informática de Aragón.

Formación

2010 a 2014	Estudiante de la EINA Grado en Ingeniería informática en la rama de Ingeniería de Computadores
Inglés	Certificate in Advanced English (equiv. C1), de la universidad de Cambridge.
Alemán	Goethe-Zertifikat C1 (Zentrale MittelStufenPrüfung)

1 de junio de 2014

A.4. CV: Alejandro Gracia Mateo

Curriculum Vitae

Lenguajes de programación utilizados

Java, C, C++,(ensamblador)

ARM, ARM thumb

CLIPS, Haskell, erlang

HTML CSS,JSP, XML

Matlab,SQL

Experiencia

- | | |
|------|---|
| 2013 | Realización de un compresor de ficheros de texto. |
| 2013 | Realización de una aplicación de subastas online. |
| 2013 | Realización de una página de recomendación de películas. |
| 2013 | Realización de un sistema de chat para múltiples usuarios con Erlang. |
| 2013 | Realización de un compilador para un lenguaje similar a Pascal. |

Formación

- | | |
|-------------|---|
| 2010 a 2014 | Estudiante de la EINA Grado en Ingeniería informática en la rama de computación |
|-------------|---|

1 de junio de 2014

A.5. CV: Alberto Berbel Aznar**Curriculum Vitae****Lenguajes de programación utilizados**

Java, C, C++,(ensamblador)

ARM, ARM thumb

CLIPS, Haskell, erlang

HTML, CSS, JSP, XML

Matlab, SQL

Experiencia

2013	Realización de un Documento de Especificación de Requisitos de un sistema real.
2013	Desarrollo de una nueva funcionalidad para un sistema de gran tamaño en lenguaje de programación Java.
2013	Herramientas de control de versiones como Bitbucket.
2013	Técnicas y herramientas de validación y verificación de software.
2013	Herramientas para la gestión y análisis de requisitos.
2013	Herramientas de seguimiento de errores como Bugzilla.
Actualidad	Desarrollo de una aplicación web consistente en un Smart Campus.

Formación

2010 a 2014	Estudiante de la EINA Grado en Ingeniería Informática en la especialidad de Ingeniería Software
-------------	---

1 de junio de 2014

A.6. CV: Héctor Francia Molinero

Curriculum Vitae

Lenguajes de programación utilizados

Ada, Java, C, C++, Android

Ensamblador, ARM, ARM thumb

CLIPS, Haskell, erlang

HTML, CSS, XML

Matlab, SQL

Experiencia

2011	Participación en la creación de una empresa de ocio.
2011	Creación de un programa para verificar documentos bien formados en XML con Flex y Bison
2012	Creación de un juego de dominó con programación concurrente.
2013	Realización de un compresor de ficheros de texto.
2013	Creación de una BD “policia” en MySQL.
2013	Realización de una aplicación móvil de gestión de notas para Android.
2013	Realización de una página de recomendación de películas.
2013	Realización de un sistema de chat para múltiples usuarios con Erlang y Java.
2013	Realización de un compilador para un lenguaje similar a MiniLang.
2013	Creación de una página Web usando el CMS Wordpress.

Formación

2008 a 2011	Estudiante de Ingeniería Superior informática
2011 a 2014	Estudiante de la EINA Grado en Ingeniería informática en la rama de computación

1 de junio de 2014

B. HOJAS DE ESFUERZOS

B.1. Hoja de esfuerzos: Daniel García Páez

En el [siguiente enlace](#) se puede acceder a la hoja de esfuerzos de Daniel García.

B.2. Hoja de esfuerzos: Javier Briz Alastrué

En el [siguiente enlace](#) se puede acceder a la hoja de esfuerzos de Javier Briz.

B.3. Hoja de esfuerzos: Simón Ortego Parra

En el [siguiente enlace](#) se puede acceder a la hoja de esfuerzos de Simón Ortego.

B.4. Hoja de esfuerzos: Alejandro Gracia Mateo

En el [siguiente enlace](#) se puede acceder a la hoja de esfuerzos de Alejandro Gracia.

B.5. Hoja de esfuerzos: Alberto Berbel Aznar

En el [siguiente enlace](#) se puede acceder a la hoja de esfuerzos de Alberto Berbel.

B.6. Hoja de esfuerzos: Héctor Francia Molinero

En el [siguiente enlace](#) se puede acceder a la hoja de esfuerzos de Héctor Francia.

C. CRONOGRAMAS

C.1. Cronograma inicial



E20: BITPARTY - Proyecto μ Search

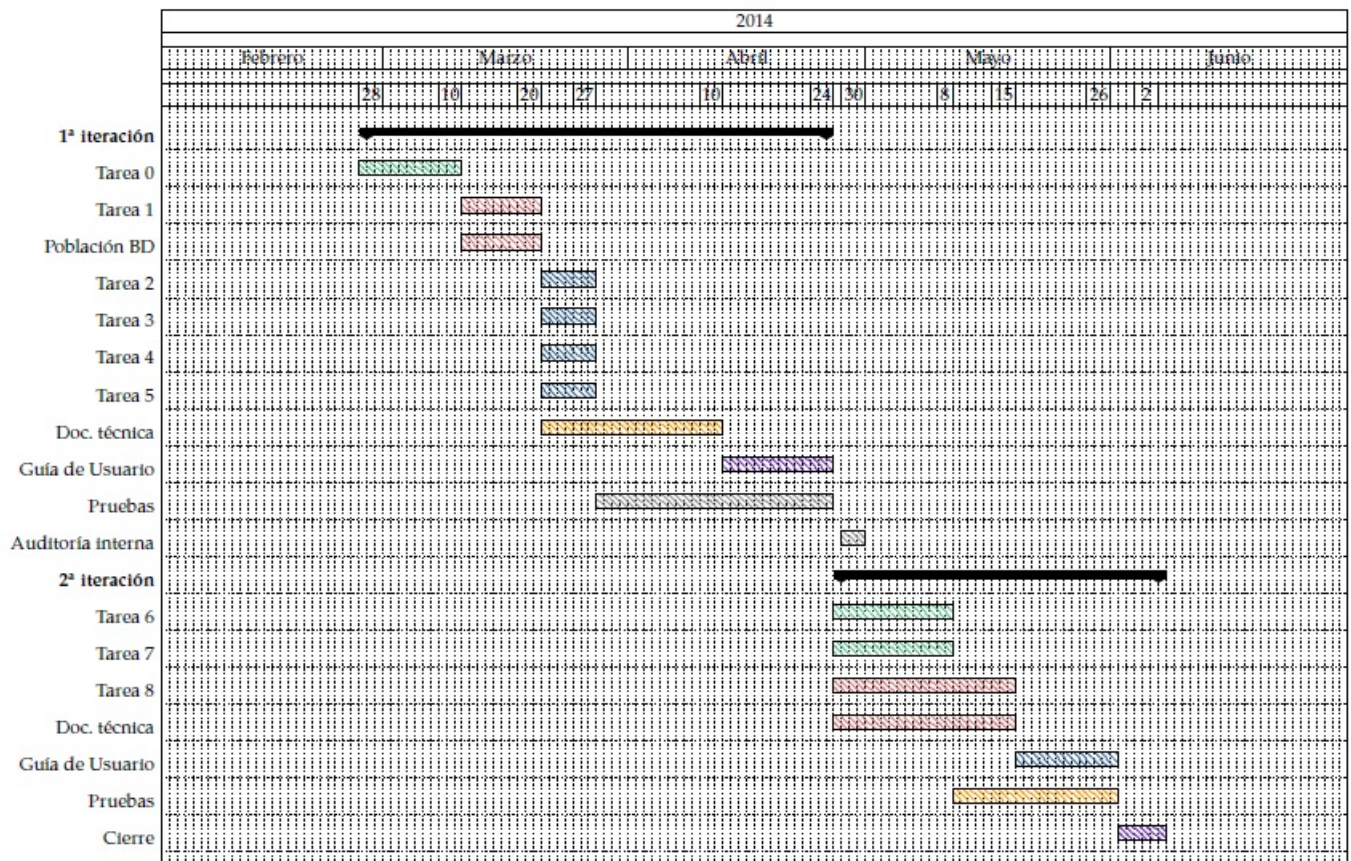


Figura 16: Cronograma inicial del proyecto

C.2. Cronograma final



E20: BITPARTY - Proyecto μ Search

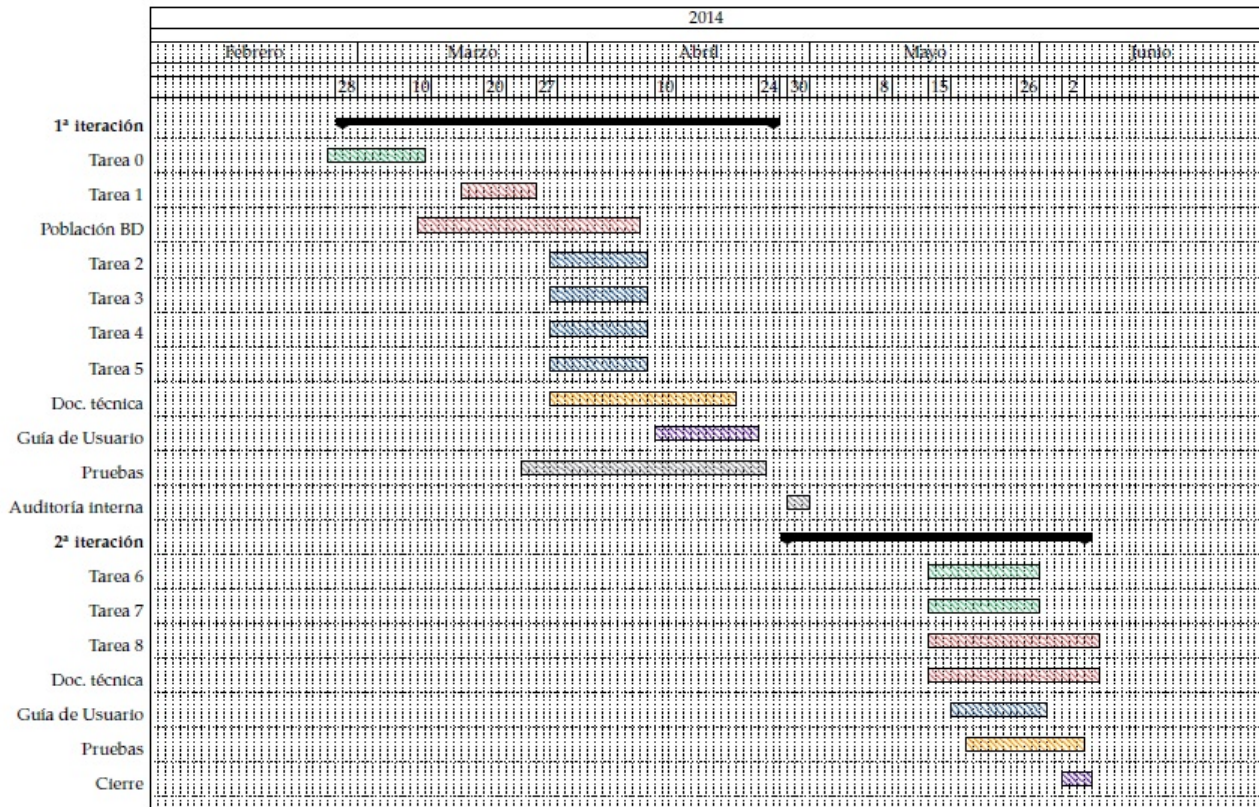


Figura 17: Cronograma real del proyecto

D. PRESUPUESTO

E. INFORMES DE PRUEBAS

En el [siguiente enlace](#) se puede acceder al informe completo de pruebas de sistema.

En el [siguiente enlace](#) se puede acceder al informe de las pruebas realizadas sobre la vista de la aplicación con el *W3C Validator*.