



# BITPARTY

Proyecto  $\mu$ Search

Versión 1.0

03/2014

## **Resumen**

## Índice

<b>1. Identificación del proyecto</b>	<b>4</b>
<b>2. Datos del licitador</b>	<b>4</b>
2.1. Denominación de la empresa . . . . .	4
2.2. Historial del equipo . . . . .	4
<b>3. Objeto de la propuesta</b>	<b>5</b>
<b>4. Requisitos del sistema</b>	<b>5</b>
<b>5. Descripción técnica de la solución</b>	<b>6</b>
5.1. Esquema de la arquitectura . . . . .	7
5.2. Tecnologías . . . . .	8
<b>6. Metodología de gestión y planificación del proyecto</b>	<b>8</b>
6.1. Fases y actividades del proyecto . . . . .	8
6.2. Recursos humanos . . . . .	9
6.3. Cronograma . . . . .	16
6.4. Presupuesto . . . . .	16

## 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*”

## 2 Datos del licitador

### 2.1 Denominación de la empresa

### 2.2 Historial del equipo

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.
- Amplia experiencia en el ámbito de la web.
- Conocimiento de entornos cliente-servidor.
- Creación de interfaces web.

### 3 Objeto de la propuesta

### 4 Requisitos del sistema

**RF01.** 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)

**RF02.** Insertar un elemento en el carro de compra.

**RF03.** Eliminar un elemento del carro de compra.

**RF04.** Modificar un elemento del carro de compra. Por modificar se entiende alterar el número de unidades de los elementos.

**RF05.** Se podrá generar en cualquier momento un listado de todos los elementos del catálogo.

**RF06.** 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.

**RF07.** Se permitirá realizar búsquedas de productos en base a un único campo de búsqueda (una y solo una de las características de los elementos).

**RF08.** Los resultados de la búsqueda se presentarán como un listado (sin paginación) que mostrará, de cada elemento, todos sus campos en columnas.

**RF09.** Los listados de elementos del catálogo estarán ordenados en base al campo arquitectura del elemento.

**RF10.** 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.

**RF11.** 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.

**RF12.** 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.

**RF13.** Se contará con una vista diferente para la administración del catálogo a la que no podrán acceder los clientes (se ejecutará solamente en local), sólo los administradores. En ésta vista se podrán realizar las acciones de:

- 1 - Insertar un elemento en el catálogo.
- 2 - Eliminar un elemento del catálogo.
- 3 - Modificar un elemento del catálogo (cambiar cualquiera de sus características).

## 5 Descripción técnica de la solución

Se propone una solución 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 propuestas para el desarrollo del proyecto son:

- HTML 5

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

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

- Google Chrome  $\geq 30$
- Internet Explorer  $\geq 10$
- Mozilla Firefox  $\geq 27$
- Opera  $\geq 12$

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

## 5.1 Esquema de la arquitectura

El patrón que vamos a utilizar para el diseño arquitectural de nuestro catálogo va a ser el de Modelo-Vista-Controlador, en concreto, la variante Modelo-Vista-Presentador (véase la figura 1).

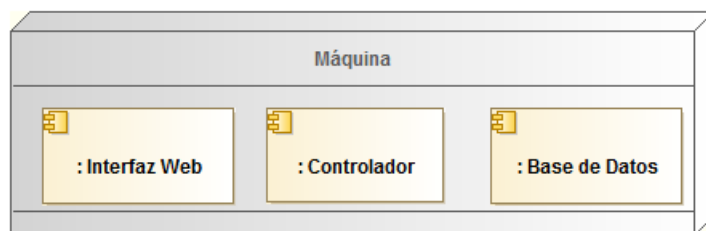


Figura 1: Diagrama de despliegue del sistema

**Interfaz web.** Es nuestro componente vista, lo que utilizan los usuarios para interactuar con la aplicación y visualizar los resultados que producen dichas interacciones. Las acciones del usuario que impliquen el acceso o la modificación de los datos del modelo 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. Cuando la vista notifica el presentador que el usuario ha hecho algo (por ejemplo, hacer clic en un botón), el presentador a continuación, actualiza el modelo y sincroniza los cambios entre el modelo y la vista.

**Es Base de datos.** Es nuestro componente modelo, se encarga de encapsular los datos y ofrecer operaciones para su acceso y procesamiento. Solo el componente Controlador interactúa con este componente.

## 5.2 Tecnologías

# 6 Metodología de gestión y planificación del proyecto

## 6.1 Fases y actividades del proyecto

El proyecto se divide en tres fases principales:

1. Lanzamiento del proyecto.
2. Primera iteración del proyecto.
3. Segunda iteración del proyecto.

Se desarrollan a continuación las actividades o tareas que componen cada una de las fases:

### 1. Lanzamiento del proyecto.

- Proceso de obtención de los requisitos funcionales y no funcionales de la aplicación.
- Diseño de un prototipo sin funcionalidad de la interfaz web de la aplicación.
- Diseño de la base de datos con la que trabajará la aplicación.
- Instalación y puesta a punto del sistema sobre el que funcionará la aplicación.
- Planificación del datagrama de actividades para las dos iteraciones del proyecto y realización de un presupuesto.

### 2. Primera iteración del proyecto.

- Población de la base de datos.
- TAREA 1: Implementación básica de la interfaz web de la aplicación.
- TAREA 2: Implementación del control para insertar, modificar y eliminar elementos en el catálogo.



- TAREA 3: Implementación del control para mostrar listados de elementos del catálogo.
- TAREA 4: Implementación del control para el carrito de compra de la aplicación.
- TAREA 5: Implementación del control para generar, en texto plano, pedidos de compra.
- Documentación técnica de todo lo implementado (documentación de código y mantenimiento de una *Wiki*.)
- Documentación del manual y guía de usuario de lo implementado hasta el momento.
- Preparación y realización de pruebas.

### 3. Segunda iteración el proyecto.

- Población de la base de datos.
- TAREA 6: Implementación del control para realizar búsquedas en el catálogo.
- TAREA 7: Implementación del control para generar pedidos de compra en formato PDF.
- TAREA 8: Mejora de la implementación inicial de la interfaz web de la aplicación
- Documentación técnica de todo lo implementado (documentación de código y mantenimiento de una *Wiki*.)
- Documentación del manual y guía de usuario final.
- Preparación y realización de pruebas.
- Cierre y puesta a punto final del proyecto.

Se puede observar una clara relación entre los requisitos del sistema definidos anteriormente y las tareas "TAREA X" aquí definidas. Pues es de esa forma como se ha abordado la planificación de tareas y actividades.

Además, las tareas de documentación y pruebas son comunes a ambas iteraciones y se realizan paralelamente a la implementación con la intención de adelantar trabajo y no olvidar cosas por el camino.

## 6.2 Recursos humanos

# Curriculum Vitae

**Nombre completo:** Alberto Berbel Aznar

## 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
-------------	---

13 de marzo de 2014

# Curriculum Vitae

**Nombre completo:** Alejandro Gracia Mateo

## 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 <b>Ingeniería informática</b> en la rama de <b>computación</b> |
|-------------|---|

13 de marzo de 2014

# Curriculum Vitae

**Nombre completo:** Javier Briz Alastrué

## Lenguajes de programación utilizados

Java, C, C++

HTML CSS, JSP, JavaScript

MySQL

## Experiencia

- 2013 Cofundador de Prototyp3D y FaryNozzle.
- Impresión 3D, construcción de impresoras y creación de piezas para impresoras RepRap.
- Automatización de impresoras 3D RepRap.
- Administración de sistemas y desarrollos centrados en Raspberry Pi.
- Colaboración en el desarrollo de Octoprint: Software anfitrión para impresión 3D.
- 2007 actualidad Presidente en Púlsar. Púlsar es la asociación de Software Libre de la Universidad de Zaragoza.
- Certificado de servicios distinguidos de la Universidad.
- 2008 actualidad ISC Secretario. ISC es una junior empresa. Es una asociación que tiene como fin acercar el mundo de la empresa a estudiantes universitarios.
- 2008 actualidad Administrador de sistemas y clusters de computación en el grupo de Fluidodinámica Numérica de la Universidad de Zaragoza becado.
- 2008 Administrador de servidores y estaciones de trabajo en el Área de Mecánica de Fluidos de la Universidad de Zaragoza. Becado (2008 mayo 2012) y Personal de Administración y Servicios (junio 2012-febrero 2014).

## Formación

- 2007 a 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 "Fernando Lázaro Carreter".<sup>en</sup> Zaragoza, julio 2009. Tres semanas en la Universidad de Hull, julio de 2007. Trinity College London Spoken English grade 7, julio 2007

13 de marzo de 2014

# Curriculum Vitae

**Nombre completo:** Daniel García Páez

## 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.

13 de marzo de 2014

.svg

.svg

### **6.3 Cronograma**

### **6.4 Presupuesto**

Se presenta a continuación la propuesta de presupuesto u oferta económica: