

# **Proyecto** Truequéalo

## **Grupo** The Red Chicken

### "DOCUMENTO DE PLANIFICACIÓN"

Hito: 1

Fecha entrega: 23-12-2016

Versión: 1

#### Componentes:

- Juan Francisco Bustos Correas
- Pablo Serna Martínez
- Yolanda Torregrosa Hernández
- Alejandro Torres Mateu
- Raquel Yuste Torregrosa

## Contenido

### Contenido

1. Propósito
2. Análisis de riesgos.
  - 2.1. Identificación de riesgos.
  - 2.2. Prioridades de los riesgos.
  - 2.3. Planes de contingencia.
3. Estimación de costes.
  - 3.1. Ley de Parkinson.
  - 3.2. Pricing to Win.
  - 3.3. Puntos Objeto y/o Puntos de Función.
  - 3.4. Comparación y discusión de los valores obtenidos.
4. Agenda del proyecto.
  - 4.1. Relación de actividades (WBS).
  - 4.2. Plan general del proyecto (Hitos).
  - 4.3. Plan detallado del proyecto (2 iteraciones).
  - 4.4. Asignación de tiempo y recursos a actividades.
    - 4.4.1. Recursos humanos.
5. Mecanismos de seguimiento y control.

## 1. Propósito

El propósito de este documento es en primer lugar analizar y detallar los posibles riesgos a los que nuestro proyecto está y estará expuesto, con la finalidad de poder prevenirlos y saber cómo actuar en caso de que sucedan.

En segundo lugar, este documento pretende estimar los diferentes costes que va a suponer realizar y sacar adelante el proyecto, hablando tanto del tiempo que se invertirá como del número de personas que necesitaremos y coste monetario que supondrán.

Por último, en este documento trataremos de establecer una planificación del proyecto: hablaremos de qué se va a hacer, cómo se va a hacer, quién y en qué momento.

## 2. Análisis de riesgos.

### 2.1. Identificación de riesgos.

#### 2.1.1. Tecnología

##### **Desconocimiento de las técnicas a utilizar**

Para la realización del proyecto, hemos tenido que aprender a utilizar nuevas técnicas de programación, así como nuevos programas que implementan estas técnicas. Hasta ahora hemos encontrado todo lo necesario para ello pero la falta de información puede ser un factor importante a la hora de seguir con el desarrollo. Este desconocimiento puede generar un parón importante de la cadena de trabajo y la pérdida de tiempo puede ser sustancial.

##### **Problemas con el equipo informático**

Para la realización del proyecto **Truequéalo**, *The Red Chicken* utiliza diversos equipos informáticos para poder desarrollar el backend y el frontend de la misma. En cuanto a los riesgos que existen podemos encontrar el deterioro o no funcionamiento de estas herramientas que impedirían en gran medida el desarrollo de las tareas, o en el peor de los casos, una pérdida de tiempo grandísima hasta la adquisición de otros equipos de similares características.

##### **Pérdida de datos**

Los datos son la base de funcionamiento más importante de cualquier aplicación, por eso es importante mantenerlos a buen recaudo. En nuestra aplicación tenemos dos tipos de datos que no queremos perder, por un lado la aplicación en sí, es decir el código base y por otro lado los datos generados a partir de la aplicación, la base de datos. El principal riesgo asociado a estos datos sería la pérdida de los mismos, ya sea por un problema externo como puede ser un fallo en el servidor en el que están alojados o un problema interno como por ejemplo un error humano.

### 2.1.2. Personas

Uno de los apartados más importantes para la correcta realización de un proyecto es el buen funcionamiento del grupo de trabajo y su grado de compatibilidad. Existen varios riesgos que pueden provocar que el grupo no rinda como debería o incluso el bloqueo del proyecto.

- Fallecimiento o enfermedad de uno de los integrantes del grupo.
- Pelea sentimental entre dos integrantes de este.
- Inexistencia de amistad o química entre varios integrantes del grupo.
- Desacuerdo en decisiones respecto al proyecto.
- Falta de trabajo de algún integrante.
- Abandono del proyecto por alguno de los integrantes.

### 2.1.3. Organizacional

A lo largo del proyecto pueden surgir problemas financieros que deriven en una reducción del presupuesto del proyecto.

### 2.1.4. Estimación

Es muy importante planificar cómo se utiliza el tiempo y en qué tareas se empeña más o menos tiempo. Durante la realización del proyecto el tiempo es un problema ya que muchas veces no se sabe cuánto tiempo es necesario para desarrollar una parte.

### 2.1.5. Requerimientos del proyecto

A medida que avanza el proyecto pueden cambiar tanto las tecnologías como las necesidades del cliente. Esto supone un cambio en la especificación del proyecto, generando, cambiando o incluso eliminando requisitos.

### 2.1.6. Herramientas a utilizar

Uno de los posibles problemas que nos pueden surgir es que las herramientas que usemos no sean las adecuadas. Por ejemplo, que tanto el rendimiento de los servidores como el de la base de datos no sea el óptimo y de esta manera no podamos ofrecer una experiencia al cliente o usuario final como debería.

## 2.2. Prioridades de los riesgos.

<b>Posibles problemas</b>	<b>Probabilidad</b>	<b>Efectos</b>
Técnicas a utilizar	Alta	Tolerable
Equipo tecnológico	Baja	Serio
Pérdida de datos	Baja	Catastrófico
Personas	Baja	Serio
Estimación	Alta	Serio
Requerimientos del proyecto	Media	Tolerable
Problemas herramientas	Media	Serio

## 2.3. Planes de contingencia.

### 2.3.1. Tecnología

#### Desconocimiento de las técnicas a utilizar

Tenemos varias formas de solucionar este problema. El primer paso que tomaríamos sería investigar por nuestra cuenta usando recursos como internet. En el caso de que no encontráramos una solución o una solución óptima a nuestro problema, consultaríamos a los profesores de la titulación. Si finalmente no podemos resolver el problema, estimaríamos el peso del requisito en el que tenemos el problema y decidiríamos si dejarlo para un futuro o simplemente eliminarlo del proyecto.

#### Problemas con el equipo informático

Para prevenir esto, el grupo de trabajo está realizando un cuidadoso mantenimiento de los equipos y se tiene a disposición otros aparatos de similares características en el caso de que los utilizados comúnmente fallen.

#### Pérdida de datos

Para mantener el código base de la aplicación usaremos un repositorio con control de versiones. Esto nos permitirá volver a cualquier cambio que hayamos hecho en la aplicación en cualquier momento. Además, los desarrolladores usarán un flujo de trabajo en el que para cada tarea harán una rama nueva y al terminar la tarea unirán el código desarrollado a una rama base, evitando así colisión y pérdida de código.

El repositorio estará alojado por un servicio de un tercero, en nuestro caso *Bitbucket* que nos garantiza su replicación y disponibilidad. Aun así, en el caso de que se produzca una pérdida del repositorio remoto, tendremos copias locales de la rama base en cada uno de los equipos, por lo que la pérdida sería mínima.

Por otro lado, y con un mayor peso y riesgo, tenemos los datos generados por la aplicación, la base de datos. Para solventar cualquier problema de pérdida de datos hemos desarrollado un plan en el que haremos una copia de seguridad completa el primer lunes de cada mes. A continuación, el primer martes haremos una copia de seguridad incremental y el resto de días de la semana una diferencial. El resto del mes haremos una copia de seguridad

incremental los lunes y el resto de la semana una diferencial. Por tanto tenemos la base de datos actualizada día a día en nuestra copia de seguridad. Puede ser que tengamos una pérdida de datos si ocurre algún fallo entre copia y copia, pero la pérdida sería como máximo de los datos introducidos entre las 24 horas entre copia y copia. Asumimos esa posible pérdida ya que un menor tiempo entre copia y copia resultaría en un importante costo computacional.

### 2.3.2. Personas

Para evitar estos riesgos lo correcto sería mantener una correcta conducta con el resto de los integrantes para evitar conflictos y crear un ambiente de trabajo adecuado para trabajar.

En nuestro proyecto tratamos de mantener un flujo de información total entre las partes del proyecto para que en el caso de que un componente abandone el proyecto todos podamos saber cómo resolver sus tareas pendientes o en ejecución. Es por eso que hemos decidido usar herramientas como *Bitbucket* y *Jira*.

### 2.3.3. Organizacional

Realizar una buena estimación de costes que se adecue al presupuesto del que disponemos, además de estudiar todas las posibilidades en cuanto a servicios que necesitaremos contratar escogiendo las más adecuadas teniendo en cuenta lo que ofrecen y el coste que supondría analizando bien una buena relación calidad-precio.

### 2.3.4. Estimación

Para prevenir esto estamos utilizando *Jira* para estimar el tiempo de cada tarea, pero dejando un tiempo de margen por si algo falla o la tarea conlleva más tiempo del pensado inicialmente. De esta forma evitamos subestimar las tareas. Puede ser que haya tareas en las que la estimación sea superior finalmente al tiempo real, por eso, hemos decidido que si esto ocurre, el tiempo sobrante se debe emplear investigando futuras tareas y formas de resolverlas.

### 2.3.5. Requerimientos

Debido a esto, nuestro enfoque inicial es incluir los requerimientos mínimos para tener un proyecto funcional y terminado, y en base al tiempo que nos reste para hacer el proyecto ir añadiendo más requisitos. De esta forma nos aseguramos que como mínimo desde el Hito 1 podremos entregar un producto terminado, pero no con la funcionalidad completa. En los siguientes hitos tendremos una evolución del producto, no teniendo que esperar al último día para poder mostrar algo funcional. De esta manera el cliente puede visualizar desde un primer momento qué es lo que va a tener como producto lo que le permitirá saber exactamente qué es lo que quiere y sugerir cambios sin que afecten demasiado al desarrollo del proyecto.

### 2.3.6. Herramientas a utilizar

Si esto sucede, trataríamos primero de optimizar todo lo posible los servidores actuales, y en el caso de que esto no fuera suficiente estimaríamos la posibilidad de contratar un servidor más potente o mover nuestros servicios a otro proveedor.

## 3. Estimación de costes.

### 3.1. Ley de Parkinson.

La duración estimada del proyecto del proyecto es de **191 días laborables** (los domingos son días del señor). La jornada laboral durará 8 horas, lo que nos deja un total de **1528 horas /persona**. Como el proyecto está formado por 5 integrantes a razón de **12 €/hora** el coste de personal por día trabajado será de **480 €**.

Por tanto, el **coste total** del proyecto será de **91680 €**.

### 3.2. Pricing to Win.

Utilizaremos Google AdSense para obtener ingresos a través de banners de publicidad que colocaremos estratégicamente en nuestra plataforma.

<i><b>Páginas vistas</b></i>	<i><b>Ingresos mensuales</b></i>
100.000	300,00 €
300.000	900,00 €
500.000	1.500,00 €
700.000	2.100,00 €
900.000	2.700,00 €
1.000.000	3.000,00 €

### 3.3. Puntos Objeto y/o Puntos de Función.

#### 3.3.1.App Móvil

<i><b>Parámetro de medida</b></i>	<i><b>Contador</b></i>	<i><b>Peso</b></i>
Inicio	1	2
Login	1	1
Registro	1	1
Anuncio	1	2

Filtro de búsqueda	1	2
Resultado de búsqueda	1	2
Perfil	1	3
Editar perfil	1	2
Chat	2	3
Ajustes	5	2
Menú	1	1
Favoritos	1	1
Crear anuncio	1	2
Reportar	1	1

### 3.3.2.Web

<b>Parámetro de medida</b>	<b>Contador</b>	<b>Peso</b>
Inicio	1	3
Login	1	1
Registro	1	1
Anuncio	1	3
Filtro de búsqueda	1	2
Resultado de búsqueda	1	2
Perfil	1	3
Editar perfil	1	2
Chat	2	3
Contacto	1	1
Menú	1	2
Favoritos	1	1
Crear anuncio	1	2
Reportar	1	1
Recuperar contraseña	2	1
Preguntas frecuentes	1	1
Términos y privacidad	1	1

### 3.3.3.Backoffice

<b>Parámetro de medida</b>	<b>Contador</b>	<b>Peso</b>
Métricas	1	1
Anuncios pendientes	1	1
Usuarios reportados	1	1
Buzón de contacto	1	1
Categorías	1	1



Listado de anuncios	1	1
Usuarios baneados	1	1
Listado de usuarios	1	1
Anuncios reportados	1	1

### 3.4. Comparación y discusión de los valores obtenidos.

<b>Recurso</b>	<b>Ingresos mensuales</b>
Personal	91.680,00 €
Dominio	3,20 €
Servidor	87,00 €/año
Gestor de proyectos (JIRA)	10,00 €
TOTAL	91.780,20 €

## 4. Agenda del proyecto.

### 4.1. Relación de actividades (WBS).

Descomposición del proyecto en una lista de actividades a realizar para cumplir con los objetivos del mismo (*Realizado en Project*).

### 4.2. Plan general del proyecto (Hitos).

Agrupación de relación de actividades en los diferentes Hitos (*Realizado en Project*).

### 4.3. Plan detallado del proyecto (2 iteraciones).

Descomposición de dichas actividades en tareas más concretas y estimación de las mismas en periodos de 2 semanas (*Realizado en Project*).

### 4.4. Asignación de tiempo y recursos a actividades.

#### 4.4.1. Recursos humanos.

Asignación de cada una de las tareas programadas a uno o varios de los integrantes del proyecto (*Realizado en Project*).

## 5. Mecanismos de seguimiento y control.

Para el seguimiento y control del proyecto utilizaremos JIRA integrado con Bitbucket, donde almacenaremos el repositorio del proyecto.

*Bitbucket* nos permite crear *pull-requests* en las que cada componente del grupo debe revisar y aprobar, o comentar para una modificación, el código desarrollado por los otros componentes antes de unir ese código con la rama base. Esto permite mantener informado al grupo de trabajo del código nuevo y nutrir de nuevas técnicas de desarrollo a todos los componentes del grupo.

*Jira*, por otro lado, nos permite ver en todo momento qué tarea está haciendo cada componente del grupo. Al ser una herramienta online, todos los componentes tenemos acceso a él y por tanto podemos ver en cada momento cómo de avanzadas van las tareas y quién está haciendo cada cosa. Sirve también para ser resolutivo en disputas entre componentes sobre “*quién ha hecho más que quién*”, ya que podemos ver claramente quién ha hecho cada tarea.