

I/O RECORDS

Plan del proyecto del software



Autores

David Arroyo Segovia
Ignacio Cepeda Bajo
Ángel Cruz Alonso
Hao Hao He
Carla Peñarrieta Uribe

Control de cambios

Número de versión	Fecha	Autores	Descripción
1	03/01/2017	Ignacio Cepeda David Arroyo Carla Peñarrieta Hao Hao He Ángel Cruz	Comienzo del Plan de Proyecto
2	04/01/2017	Ignacio Cepeda	Introducción
3	08/01/2017	David Arroyo	Redacción <i>Estrategia de Gestión del Riesgo</i>
4	09/01/2017	Carla Peñarrieta	Redacción <i>Organización del Personal</i>
5	13/01/2017	Hao Hao He	Redacción <i>Mecanismo de Seguimiento y control</i>
6	14/01/2017	Ignacio Cepeda David Arroyo Ángel Cruz	Estimaciones del proyecto Lista de recursos Revisión 1
7	15/01/2017	Ignacio Cepeda Carla Peñarrieta Hao Hao He	Planificación temporal Revisión 2

Índice

1. Introducción	4
1.1 Propósito del plan.....	4
1.2 Ámbito del proyecto y objetivos.....	4
1.2.1 Declaración del ámbito.....	4
1.2.2 Funciones principales	4
1.2.3 Aspectos de rendimiento.....	4
1.2.4 Restricciones y técnicas de gestión.....	4
1.3 Modelo de proceso.....	5
2. Estimaciones del proyecto	6
2.1 Datos históricos.....	6
2.2 Técnicas de estimación	6
2.3 Estimaciones de esfuerzo, coste y duración.....	6
3. Estrategia de gestión del riesgo.....	7
3.1 Introducción: Estudio de los riesgos	8
3.2 Priorización de riesgos del proyecto.....	9
3.3 Plan de gestión del riesgo Reducción, supervisión y gestión del riesgo	11
4. Planificación temporal	13
4.1 Estructura de descomposición del trabajo/Planificación temporal.....	13
4.2 Gráfico Gantt	15
4.3 Red de tareas	16
4.4 Tabla de uso de recursos.....	17
5. Recursos del proyecto	18
5.1 Personal.....	18
5.2 Hardware y software	18
5.3 Lista de recursos	19
6. Organización del personal (Gestión del Equipo)	20
6.1 Estructura de equipo (si procede)	20
6.2 Informes de gestión.....	20
7. Mecanismos de seguimiento y control.....	22
7.1 Garantía de calidad y control (Plan de Calidad).....	22
7.2 Gestión y control de cambios (Plan GCS)	22

1. Introducción

1.1 Propósito del plan

El objetivo de este documento es establecer una planificación detallada y eficaz del desarrollo de nuestro proyecto. Los temas a tratar basan fundamentalmente en las líneas de trabajo a desarrollar junto a los recursos disponibles para efectuar dichas tareas, todo ello establecido dentro de un marco de trabajo realista. No obstante, debido principalmente a los posibles cambios que efectuemos en el proyecto, este documento tendrá un contenido relativamente dinámico y flexible, adaptado a los requisitos establecidos en cada momento, los cuales dependen de las circunstancias concretas y las exigencias que estas planteen.

1.2 Ámbito del proyecto y objetivos

1.2.1 Declaración del ámbito

La siguiente información está también incluida en nuestra especificación de requisitos. No obstante, la aplicación tiene como objetivo gestionar un negocio de venta de discos de música, aunque también sería aplicable a otro negocio minorista similar. El software realizará tareas de gestión del inventario de productos, de ventas y de clientes. Por tanto, no se trata de una herramienta para gestionar tareas como las compras a proveedores o la gestión del personal.

1.2.2 Funciones principales

Nuestra aplicación tiene como principales objetivos las siguientes:

- Facilitar y agilizar la administración del catálogo de la tienda.
- Administrar las ventas de una forma sencilla para que cualquier nuevo empleado pueda incorporarse rápidamente al negocio.
- Posibilitar compras desde la propia aplicación para los clientes que acudan a la tienda.

1.2.3 Aspectos de rendimiento

Dadas las características de nuestro proyecto, y en base al hecho de que se va a utilizarse en un pequeño negocio, el volumen de información que manejará será reducido, lo cual implica un rendimiento aceptable con casi cualquier configuración base de hardware, por lo que el servidor será de gama baja-media y dedicaremos más esfuerzo a realizar una implementación del sistema eficiente. En cualquier caso, la especificación de requisitos desarrolla esta sección con mayor profundidad.

1.2.4 Restricciones y técnicas de gestión

Con respecto a las fechas de entrega, en primera instancia estamos centrados en la correspondiente al 12/01/2017, la cual se compone de los siguientes elementos.

- Plan de proyecto del software (Esquema Pressman)
- Especificación de Requisitos de Software (SRS) de la aplicación.

El equipo se compone de 5 integrantes, y en cuanto a la ingeniería, la idea inicial es programar el proyecto en Java, integrando una solución que permita:

- Establecer un diseño sencillo que comparta dicha cualidad con su implementación para así constituir un esquema intuitivo y agradable, ya que deberá ser utilizada por personas con conocimientos técnicos reducidos.
- Atender peticiones simultáneas al servidor cuando haya varios usuarios conectados en paralelo.
- Establecer restricciones a nivel de usuario para ciertas funciones del sistema.

1.3 Modelo de proceso

Se usará el proceso unificado de RUP (Proceso unificado de desarrollo) que se caracteriza por estar dirigido por casos de uso, naturaleza iterativa e incremental y centrado en la arquitectura.

El modelo de proceso elegido se compone de 5 flujos de trabajo o actividades estructurales (A.E.) que se iteran:

- Requisitos.
- Análisis.
- Diseño.
- Implementación.
- Prueba.

Cabe destacar que no todos los flujos de trabajo tienen el mismo peso en cada fase, existiendo por tanto cierto dinamismo en su realización debido al desarrollo y evolución del proyecto.

2. Estimaciones del proyecto

2.1 Datos históricos

No se dispone de datos históricos para realizar estimaciones, ya que no se realizaron proyectos previos similares y, por tanto, no hay información posible de cada a una recopilación que pueda ser objeto de análisis.

2.2 Técnicas de estimación

La estimación en un proyecto es vital para su correcto desarrollo. Supone una actividad crucial y compleja que puede determinar un punto de inflexión de cara a la búsqueda de resultados concretos. Puede anticipar un fracaso antes de materializarlo, o ayudar a moldear y potenciar un éxito.

Con respecto a nuestro temario, en nuestro proyecto nos centraremos en la estimación basada en la descomposición del proceso, la cual es útil en nuestro caso dado que nos permite identificar un conjunto pequeño de actividades o tareas a realizar, las cuales serán sometidas a un análisis de esfuerzo necesario para ser ejecutadas. Lo adecuado al realizar dicho análisis es obtener un valor de variaciones equivalente al razonable. Dichos valores se encuentran en la franja del 20%, porcentaje por encima del cual las estimaciones carecerían de validez. Por poner un ejemplo, el acceso a la aplicación se descompondría en cliente y empleado, los cuales a su vez tendrían funciones específicas y claramente diferenciadas.

2.3 Estimaciones de esfuerzo, coste y duración

Con respecto a la duración del proyecto, este comenzó el 15/12/2016 y estimamos su finalización para el 25/05/2017, lo cual supone un total de 161 días, de los que se dedicarán semanalmente entre 2 y 3 días de trabajo.

Cabe destacar que el proyecto no va a suponer ningún coste económico, pero si tendrá una repercusión en forma de coste de esfuerzo, ya que consumirá tiempo y desempeño de los miembros del equipo.

En nuestro proyecto existen cuatro módulos claramente diferenciados:

- Usuarios.
- Personal.
- Ventas.
- Discos.

En principio vamos a estimar una distribución de esfuerzo equitativa en cada módulo, ya que todos tienen un número similar de funciones y preferimos operar con cierto margen de maniobra que nos permita cumplir los plazos deseados.

En los 161 días comprendidos en la planificación inicial del proceso hay $161/7 = 23$ semanas, de las cuales vamos a trabajar aproximadamente 3 días por semana, lo

que hace un total de $3 \times 23 = 69$ días de trabajo. Asumiendo un trabajo de 2h/día tenemos $e = 5(p) \times 2(h/d) \times 69(d) = 690(ph)$

Por tanto:

$$690(ph) / 2(h/d) = 345(pd).$$

Al margen de cada módulo, se prevén las siguientes acciones:

- Especificación de requisitos: 16(pd).
- Revisión del proyecto: 9(pd).

Por tanto, quedan 320(pd). Al haber decidido establecer una estimación de esfuerzo equitativa, aplicamos $320(pd) / 4(\text{módulos}) = 80(pd/\text{módulo})$.

En todos los módulos se requiere esfuerzo en cada acción realizada, por lo que se hará la siguiente división.

- Análisis: 15 (pd).
- Diseño: 15 (pd).
- Codificación: 30 (pd).
- Pruebas: 20 (pd).

Con respecto a las iteraciones incrementales, en un principio estimamos 2 por módulo, con una cantidad de esfuerzo similar para cada una, lo cual supondría dividir en 2 partes la relación de esfuerzos arriba indicada, pudiendo modificarse más adelante esta decisión, ya que consideramos esta una estimación compleja de concretar.

En cuanto al número de personas que van a participar en cada actividad de módulo, nuestra idea inicial es que todos participemos en todas las actividades, dado el carácter didáctico de este proyecto y que lo buscamos es enriquecernos y adquirir una gama de conocimientos lo más amplia posible, para lo cual creemos importante que todos nos empapemos de cada fase de trabajo. No obstante, a lo largo del proyecto podrán efectuarse ligeras variaciones en este aspecto, debidas principalmente a las cualidades técnicas de cada miembro del equipo en áreas concretas; en tal caso, dichas modificaciones serán contempladas en este plan de proyecto tan pronto como sean planificadas.

3. Estrategia de gestión del riesgo

Un riesgo es todo aquello que pueda afectar negativamente al proyecto de software. De hecho, *todo* puede afectar negativamente a nuestro proyecto.

Aunque *todo* es preocupante, hay unos riesgos más preocupantes que otros. En este apartado del documento nos encargaremos de estudiar los riesgos más preocupantes y de su gestión.

Dentro de las diferentes estrategias disponibles para gestionar el riesgo vamos a utilizar la estrategia proactiva. Se caracteriza por que comienza antes que los trabajos técnicos,

identificándose los riesgos potenciales. También se evalúa la probabilidad y consecuencia de los riesgos priorizándolos y produciendo un plan de gestión del riesgo. Aunque el objetivo principal es evitar el riesgo también hay que proporcionar planes de contingencia.

3.1 Introducción: Estudio de los riesgos

Utilizaremos la tabla SQAS-SEI para identificar los posibles riesgos indicando la descripción, probabilidad de que ocurra y consecuencias que produciría el mismo.

Riesgo 1: Desmotivación del personal

- **Descripción:** Posible desmotivación del personal por causas relativas al proyecto como externas.
- **Probabilidad:** Ocasional
- **Consecuencias:** Serias

Riesgo 2: Planificación poco realista

- **Descripción:** Planificación con fechas demasiado ajustadas y posibilidad de no haber hecho un buen reparto de las tareas.
- **Probabilidad:** Probable
- **Consecuencias:** Serias

Riesgo 3: Ausentismo

- **Descripción:** Ausentismo de algunos participantes en el proyecto por causas médicas que hacen que sea imposible avanzar en algunos puntos del desarrollo.
- **Probabilidad:** Frecuente
- **Consecuencias:** Catastróficas

Riesgo 4: Poca experiencia en el desarrollo de software

- **Descripción:** Algunos de los integrantes del equipo de desarrollo desconocen el lenguaje de programación que se va a utilizar.
- **Probabilidad:** Frecuente
- **Consecuencias:** Serias

Riesgo 5: Responsabilidades académicas

- **Descripción:** La fecha de entrega de la SRS y del Plan de Proyecto se encuentra muy próxima al período de exámenes de febrero. Esto aumentará la presión sobre los miembros del equipo.
- **Probabilidad:** Frecuente
- **Consecuencias:** Menores

Riesgo 6: Personal poco preparado

- **Descripción:** El personal está poco preparado y tiene poca experiencia para los trabajos asignados.

- **Probabilidad:** Remota
- **Consecuencia:** Serias

3.2 Priorización de riesgos del proyecto

	Frecuente	Probable	Ocasional	Remoto	Improbable
Catastrófico	Riesgo 3				
Crítico					
Serio	Riesgo 4	Riesgo 2	Riesgo 1	Riesgo 6	
Menor	Riesgo 5				
Insignificante					

Exposición al riesgo

1° Ausentismo	
Probabilidad	Frecuente
Consecuencia	Catastrófico
Nivel de riesgo	10

2° Poca experiencia en el desarrollo de software	
Probabilidad	Frecuente
Consecuencia	Serio
Nivel de riesgo	4,5

3° Planificación poco realista

Probabilidad	Probable
Consecuencia	Serio
Nivel de riesgo	3

4° Responsabilidades académicas

Probabilidad	Frecuente
Consecuencia	Menor
Nivel de riesgo	2

5° Desmotivación del personal

Probabilidad	Ocasional
Consecuencia	Serio
Nivel de riesgo	$3 * 10^{-1}$

6° Personal poco preparado

Probabilidad	Remoto
Consecuencia	Serio
Nivel de riesgo	$3 * 10^{-3}$

Dada la poca probabilidad de que ocurra el último riesgo va a ser asumido en nuestro proyecto. Sin embargo, los demás riesgos van a ser tratados activamente debido a su mayor exposición.

3.3 Plan de gestión del riesgo Reducción, supervisión y gestión del riesgo

Riesgo primero. Ausentismo

Reducción

Mediante una supervisión semanal en la que se registre quién ha acudido a las reuniones y horas de laboratorio.

Supervisión

La propia realización de un calendario en el que se recojan los nombres de quiénes han acudido a las reuniones o citas que se hayan acordado. Cuando se alcance un número elevado de veces en las que algún miembro no se presenta habrá que actuar.

Plan de Contingencia

La única forma de actuar que está en nuestra mano es avisando a la persona que no esté cumpliendo con lo acordado. En un caso extremo habría que avisar al profesor que nos ha fijado este proyecto.

Riesgo segundo. Poca experiencia en el desarrollo de software

Reducción

Con la realización de cursos sobre las tecnologías que se vayan a usar en el proyecto a desarrollar.

Supervisión

Controlando la calidad del código escrito, los errores y la productividad que tengan los encargados de la codificación.

Plan de Contingencia

En caso de que llegue a producirse, habría que dedicar la mayor parte del tiempo a estudiar y adquirir experiencia con las tecnologías escogidas. En caso de ser posible, sería una opción cambiar las tecnologías por otras con las que ya se tenga experiencia. De otra manera, habría que retrasar la entrega del proyecto, o entregar un proyecto que no cumpla todos los objetivos de calidad.

Riesgo tercero. Planificación poco realista

Reducción

Mediante una planificación bien meditada y que sea consensuada por todos los miembros del equipo, sería posible reducir de forma significativa este riesgo.

Supervisión

Se revisará si el personal puede cumplir los objetivos que exige la planificación. Cuando el personal no pueda cumplir los objetivos o ciertos aspectos están fuera del alcance realista, habrá ocurrido el riesgo.

Plan de Contingencia

Si el riesgo sucede se procederá a cambiar la planificación de una manera más realista.

Riesgo cuarto. Responsabilidades académicas

Reducción

Con una organización del tiempo más eficaz y adelantando trabajo en los tiempos libres. Aunque la fecha de los exámenes es imposible cambiarla, siempre se puede organizar el tiempo de manera que haya tiempo para ambas cosas.

Supervisión

Realización de un plan semanal con las horas que se va a dedicar a cada actividad para que se puedan llegar a cumplir los objetivos propuestos.

Plan de Contingencia

En caso de que no se pueda cumplir con los objetivos habría que limitarlos y priorizar actividades en la medida que afecte lo menos posible al proyecto.

Riesgo quinto. Desmotivación del personal

Reducción

Mejorando el entorno de trabajo y el ambiente entre los miembros del grupo de manera que transmita la mayor positividad posible y eliminando cualquier estímulo negativo que pueda influir negativamente al trabajo de algún compañero.

Supervisión

Midiendo la productividad de los miembros del equipo de trabajo y estudiando la calidad del trabajo realizado. También, con la realización de reuniones semanales en las que cada compañero exponga su evolución semanal y exponga su visión del proyecto.

Plan de Contingencia

Planificación de reuniones para volver a repasar los objetivos y comprobar si son los adecuados. Intentar solucionar cualquier estímulo externo que pueda influir negativamente en la motivación de los miembros del equipo.

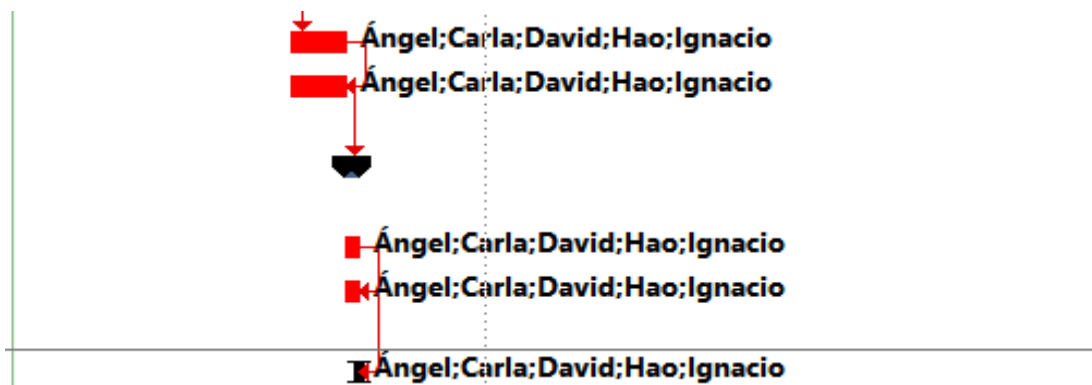
4. Planificación temporal

4.1 Estructura de descomposición del trabajo/Planificación temporal

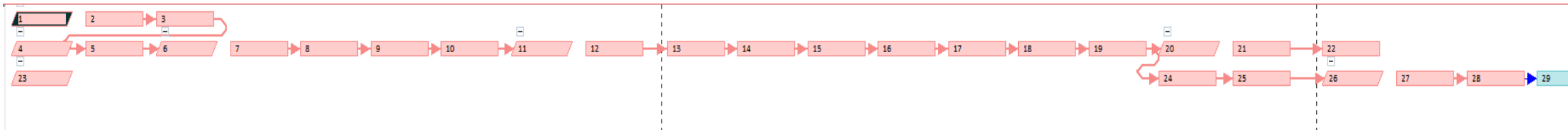
AE	PLANIFICACIÓN Y GESTIÓN DEL RIESGO		INGENIERÍA	CONSTRUCCIÓN Y ADAPTACIÓN		EVALUACIÓN CON EL CLIENTE	
Acción	Obtener SRS. Fecha de inicio: 15/12/2016. Fecha de finalización: 18/01/2017. Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Plan de proyecto Fecha de inicio: 12/01/2017 Fecha de finalización: 18/01/2017. Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Análisis global del sistema Fecha de inicio: 20/01/2017. Fecha de finalización: 10/02/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Diseño general del proyecto. Fecha de inicio: 10/02/2017. Fecha de finalización: 15/03/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Codificación	Pruebas	
Módulo usuario				Inicio del módulo. Fecha de inicio: 10/02/2017. Fecha de finalización: 15/03/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Codificación del módulo. Fecha de inicio: 16/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Depuración del módulo. Fecha de inicio: 17/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	
Módulo personal				Inicio del módulo Fecha de inicio: 10/02/2017. Fecha de finalización: 15/03/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Codificación del módulo. Fecha de inicio: 16/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Depuración del módulo. Fecha de inicio: 17/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	
Módulo ventas				Inicio del módulo Fecha de inicio: 10/02/2017. Fecha de finalización: 15/03/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao	Codificación del módulo. Fecha de inicio: 16/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla	Depuración del módulo. Fecha de inicio: 17/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla	

				<ul style="list-style-type: none">• Ignacio	<ul style="list-style-type: none">• Hao• Ignacio	<ul style="list-style-type: none">• Hao• Ignacio	
Módulo disco				Inicio del módulo Fecha de inicio: 10/02/2017. Fecha de finalización: 15/03/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Codificación del módulo. Fecha de inicio: 16/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Depuración del módulo. Fecha de inicio: 17/03/2017. Fecha de finalización: 30/04/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	
Proyecto	Revisión de SRS Fecha de inicio: 15/05/2017. Fecha de finalización: 20/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Revisión de plan de proyecto Fecha de inicio: 15/05/2017. Fecha de finalización: 20/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio			Cierre de código Fecha de inicio: 30/04/2017. Fecha de finalización: 20/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Cierre de depuración Fecha de inicio: 30/04/2017. Fecha de finalización: 20/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	
Proyecto	Fin de SRS Fecha de inicio: 20/05/2017. Fecha de finalización: 25/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio	Fin de plan de proyecto Fecha de inicio: 20/05/2017. Fecha de finalización: 25/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio					Entrega del proyecto Fecha de inicio: 25/05/2017. Fecha de finalización: 25/05/2017 Realizado por: <ul style="list-style-type: none">• Ángel• David• Carla• Hao• Ignacio

4.2 Grafico Gantt



4.3 Red de tareas



4.4 Tabla de uso de recursos

▲ Ignacio	848 horas		Trabajo	354h	462h	18h
Obtener SRS	40 horas		Trabajo	26h		
Obtener Plan de proyecto	16 horas		Trabajo	16h		
Análisis global del sistema	30 horas		Trabajo	30h		
Inicio del módulo usuario	40 horas		Trabajo	40h		
Inicio del módulo personal	40 horas		Trabajo	40h		
Inicio del módulo ventas	40 horas		Trabajo	40h		
Inicio del módulo disco	40 horas		Trabajo	40h		
Codificación del módulo u	48 horas		Trabajo	14h	34h	
Depuración del módulo u	48 horas		Trabajo	14h	34h	
Codificación del módulo i	50 horas		Trabajo	16h	34h	
Depuración del módulo p	48 horas		Trabajo	14h	34h	
Codificación del módulo v	50 horas		Trabajo	16h	34h	
Depuración del módulo v	48 horas		Trabajo	14h	34h	
Codificación del módulo c	50 horas		Trabajo	16h	34h	
Depuración del módulo d	52 horas		Trabajo	18h	34h	
Revisión de SRS	48 horas		Trabajo		42h	6h
Revisión de plan de proyecto	56 horas		Trabajo		44h	12h
Cierre de código	30 horas		Trabajo		30h	
Cierre de depuración	30 horas		Trabajo		30h	
Fin de SRS	18 horas		Trabajo		18h	
Fin de plan de proyecto	18 horas		Trabajo		18h	
Entrega del proyecto	8 horas		Trabajo		8h	

Al haber elegido de forma inicial una distribución totalmente equitativa de trabajo, la tabla es exactamente igual que la de la fotografía para todos los miembros del equipo.

5. Recursos del proyecto

5.1 Personal

El personal del que se compone el proyecto es el siguiente. Con respecto a las funciones a ocupar por cada miembro, se va a buscar una participación global de todos los miembros del equipo en todas las partes del proyecto, si bien podrán tener lugar ciertos matices detallados a continuación.

- Ángel Cruz Alonso: Posee competencias avanzadas en programación en C++, Java, Python y el ecosistema Linux.
- David Arroyo Segovia: Posee conocimientos de programación en C++, Java, HTML y PHP.
- Carla Peñarrieta Uribe: Posee destrezas en programación Android, C++ y Java.
- Hao Hao He: Además de C++ y Java, posee conocimientos de diseño gráfico en Photoshop y After Effects.
- Ignacio Cepeda Bajo: Posee conocimientos de C++, Java, HTML y nociones avanzadas de administración y finanzas, marketing y logística.

En caso de que haya ciertas partes del trabajo que requieran conocimientos estrictamente técnicos y sean de un carácter muy avanzado, es probable que David y Ángel asuman un mayor protagonismo en su ejecución.

5.2 Hardware y software

Algunas herramientas software que serán utilizadas durante el desarrollo del proyecto son las siguientes:

- Eclipse. (Java)
- IBM Rational Software Architect.
- TortoiseSVN
- CSV de la facultad.
- GitHub.
- Microsoft Visual Project.
- Google Drive.
- Microsoft Office 2013

En lo que respecta al hardware, los ordenadores personales de los miembros del equipo y aquellos disponibles en los laboratorios de la facultad de informática serán el equipo utilizado.

Como red de comunicación para los diferentes integrantes del grupo utilizaremos WhatsApp y Skype, que nos permitirán planificar acordar decisiones concernientes al proyecto en cualquier momento y lugar.

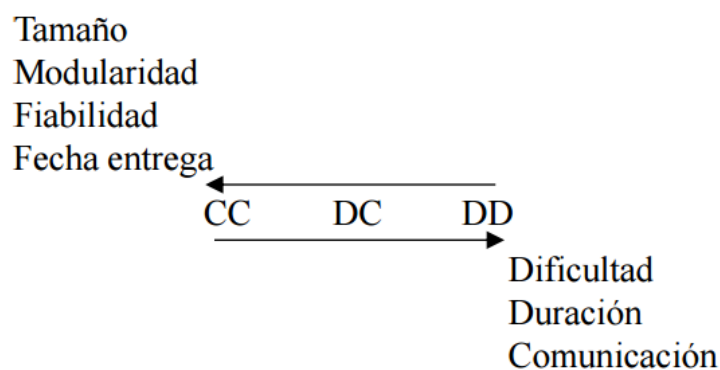
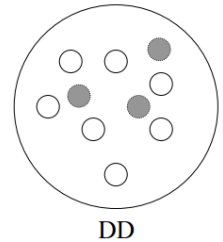
5.3 Lista de recursos

Personal	Software	Hardware
Ángel Cruz Alonso	Eclipse (Java).	Ordenadores personales y del laboratorio
Carla Peñarrieta Uribe	Microsoft Office 2013	
David Arroyo Segovia	Microsoft Project	
Hao Hao He	WhatsApp y Skype	
Ignacio Cepeda Bajo	Dropbox y Google Drive	
	IBM Rational Software Architect.	
	TortoiseSVN	
	CSV de la facultad.	
	GitHub.	

6. Organización del personal (Gestión del equipo)

6.1 Estructura del equipo

La estructura del equipo se basa en una estructura **Descentralizada Democrática (DD)** en la que dicha estructura ha sido elegida por los integrantes del grupo fundamentándonos en las **estructuras de equipo de Mantei**, la cual consta de siete factores para determinar la estructura a elegir: dificultad del problema, tamaño en líneas de código (LDC) o puntos de función (PF), duración del equipo, modularidad del problema, calidad y fiabilidad, fecha de entrega, comunicación requerida en el proyecto.



La estructura de equipo en la que nos basamos no tiene un líder de grupo permanente, aunque en nuestro caso, **David Arroyo**, ha realizado el papel de gestor del proyecto, encargado de la organización, motivación, capacidad de la resolución de problemas durante el desarrollo del proyecto. Por esa razón, se nombran coordinadores de tareas a corto plazo y se sustituyen por otros para diferentes tareas. Las decisiones sobre problemas y los enfoques se hacen a consenso del grupo. La comunicación entre los miembros del equipo es horizontal, gracias a la cual la resolución de dudas o problemas se ve limitada al compartir información y al apoyo entre los integrantes del grupo.

Por otro lado, conviene resaltar que la **cohesión** del grupo es necesaria. Un equipo está cohesionado cuando todos sus miembros comparten un objetivo como un enfoque común y luchan por conseguir los resultados deseados por todos por encima de los intereses individuales. Además, el factor humano, así como el respeto, la confianza y la tolerancia son básicas para alcanzar la cohesión necesaria, aceptando otros puntos de vista que nos permite enriquecernos y a facilitarnos las dificultades que surjan. Por esa razón, para cada uno de los integrantes se requiere compromiso, esfuerzo y la existencia de un proyecto común.

6.2 Informes de gestión

Los proyectos de Software se componen de participantes que pueden clasificarse en una de cinco categorías:

- **Gestor superior y Gestor técnico del proyecto: David Arroyo**, es el encargado de definir los aspectos del negocio que a menudo tienen una influencia significativa en el proyecto. Así como, la capacidad de resolver problemas, motivación, planificación y organización del proyecto. Además de controlar al resto de integrantes que realizan el trabajo de software.
- **Profesionales:**
 - **Ángel Cruz:** su labor está enfocado a la parte más social del proyecto, al diseño de las aplicaciones para compartir y su desarrollo, además de facilitar la unión de las diferentes partes del proyecto.
 - **Ignacio Cepeda:** encargado de desarrollar la parte más administrativa del proyecto, es decir, actualizar la base de datos y lo relacionado al almacenamiento de los datos.
 - **Carla Peñarrieta:** se centra en programar las partes que engloban al cliente, es decir, el mantener contacto con este para la correcta realización de la aplicación de acuerdo a las necesidades del cliente.
 - **Hao Hao He:** se centrará la búsqueda de errores en la codificación, en el diseño o en los requisitos, así como la verificación de los algoritmos que formarán la aplicación.

Por consiguiente, todos los integrantes del grupo deberán colaborar en las diferentes partes del desarrollo de las diferentes partes del proyecto, además del cumplimiento de los plazos establecidos para cada apartado.

- **Los clientes**, los cuales especifican los requisitos de la aplicación. Mantienen contacto con los profesionales, al poderse modificar, añadir algún otro requisito.
- **Los usuarios finales**, estarán formados por el dueño de un establecimiento (tienda de discos), los trabajadores y los clientes que acudan al establecimiento los cuales interactuarán con el software una vez que se ha entregado.

7. Mecanismos de seguimiento y control

Se va a utilizar el Google Drive para el control de las versiones, con una serie de acrónimos expuestos en el apartado 7.2.

7.1 Garantía de calidad y control (Plan de Calidad)

En nuestro proyecto, se van a llevar a cabo una serie de revisiones técnicas continuas. Para ello, utilizaremos la siguiente tabla:

Tarea	Productos	Técnicas y Prácticas	Participantes
Revisión de seguridad, para comprobar la seguridad del software	-Equipo encargado de Seguridad. -Plan de acción	-Sesiones de trabajo -Planificación	-Grupo asignado para control de seguridad. (Hao)
Revisión de calidad, para ver que éste funciona en perfectas condiciones.	-Equipo encargado del control de calidad.	-Sesiones de trabajo	-Grupo asignado para control de calidad (Nacho y Carla)
Revisión de fallos y mejoras del producto.	-Equipo encargado del control de fallos	-Sesiones de trabajo	-Grupo asignado para control de fallos (David y Ángel)

7.2 Gestión y control de cambios (Plan GCS)

Elementos que van a constituir una configuración software:

Para esta tarea nos facilitará el trabajo el software google drive ya que guarda todos los cambios realizadas, así recordando la fecha y los cambios. Y cumple los siguientes puntos:

Todos los documentos software y otros elementos de cada configuración son mantenidos como parte de los cambios guardados por ingeniería ya establecida (Google Drive).

Criterios que se van a seguir para nombrar las versiones de cada elemento de la configuración.

Para nombrar las versiones utilizaremos los siguientes acrónimos:

XXX identificador de la empresa de software

YYY identificador del proyecto

Z es un identificador del elemento

P Plan

R Especificación de Requisitos

D Documento de diseño

S Listado fuente

T Documentación de prueba

U Manual del usuario

I Guía de instalación

M Manual de mantenimiento

RL es el nivel de revisión

NNN código atributo que usamos para determinar la fecha.

La idea es que siga la siguiente estructura a la hora de ordenar las versiones:

XXX-YYY-Z-RL-NNN

Políticas de control de cambios y versionado:

Todos se quedan guardadas en el google drive para facilitar los cambios que hizo cada miembro del equipo. Para que posteriormente se pueda verificar los cambios indeseados.

- Plantillas usadas para realizar un cambio.
- Pasos a dar en caso de realizar un cambio.
- Protocolo para llegar a un acuerdo sobre el cambio.
- Componentes del comité de cambios.
- Herramientas de gestión de versiones- repositorios-Gestión de proyectos.

IO-001-P-0-03/1/17	Este es el plan del proyecto de la empresa "I/O Records" puesto bajo control de cambios el día 3 de enero de 2017.
IO-001-P-1-14/1/17	Esta es la revisión 1 del plan del proyecto. Puesto bajo control de cambios el día 14 de enero de 2017.
IO-001-P-2-15/1/17	Esta es la revisión 2 del plan de proyecto, puesto bajo control el 15 de enero de 2017.

En caso de realizar un cambio, se utilizarán los siguientes parámetros, escribiendo primero el nombre de la empresa: el número del proyecto, identificando el elemento modificado, la versión de la revisión y la fecha. A parte de utilizar las siglas establecidas, también se redactará por escrito.

Se llevará a cabo un Control Individual en el que el técnico responsable del cambio realizará las modificaciones en el documento.

Al ser un proyecto pequeño, el Comité de Cambios estará formado por una sola persona.

La herramienta para llevar a cabo la gestión de versiones va a ser Google Drive.