**Proyecto: CiberMeal**

**Plan de proyecto del software**

**(Formato Pressman)**

**Miembros del equipo:**

|  |  |
| --- | --- |
| Javier García González | Guillermo Aguilar |
| Gerald Lima | Wu Liu |
| Marta Rodenas | Álvaro de Frutos |
| Daniel Reyes | **Grupo: CiberMeal** |

**Control de cambios**

|  |  |  |  |
| --- | --- | --- | --- |
| **Número de versión** | **Fecha** | **Autores** | **Descripción** |
| V.1 | 27/07/2014 | Javier Gracia González | Creación del documento, recopilación de datos, primera versión del documento completa. |
| V.1.1 | 27/07/2014 | Gerald Lima | Análisis del riesgo (plantilla RSGR) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice

1. Introducción

1.1 Propósito del plan

1.2 Ámbito del proyecto y objetivos

1.2.1 Funciones principales

1.2.2 Aspectos de rendimiento

1.2.3 Restricciones y técnicas de gestión

1.3 Modelo de proceso

2. Mecanismos de Calidad y Control

2.1. Gestión de Calidad

2.2 Gestión y control de cambios

3. Plan de Gestión de Riesgos

3.1 Lista de posibles riesgos:

3.2 Análisis de los riesgos

3.3 Priorización del riesgo. Con Tabla del SQUAS-SEF

3.4 Análisis del riesgo (plantilla RSGR)

3.4.1 Introducción

3.4.2 Priorización de riesgos del proyecto

3.4.3 Reducción supervisión y gestión del riesgo

4. Recursos del proyecto

4.1 Personal

4.1.1 Equipo de Desarrollo

4.2 Hardware

4.2.1 Hardware de desarrollo

4.2.2 Hardware del cliente

4.3 Software

4.3.1 Software de desarrollo

4.3.2 Software del cliente

5. Organización de personal

5.1 Forma de organización del equipo

5.2 Estimaciones

6. Planificación temporal

6.1 Estructura de descomposición del trabajo o Planificación temporal

6.2 EDT

6.3 Red de tareas

7. Apéndice

7.1 Fuentes

**1. Introducción**

**1.1 Propósito del plan**

El plan de proyecto está enfocado en marcar los objetivos a los que se pretende llegar.

El objetivo del proyecto es facilitar el seguimiento médico de los pacientes realizando una dieta, anteriormente, para mantener un seguimiento eran necesarias constantes visitas médicas, si el paciente deseaba cambiar la dieta de alguna forma no le quedaba más remedio que concertar una cita con el médico, proponer el cambio, explicar el porqué del cambio, esperar a que el médico lo realice y después este entrar en contacto de nuevo con el paciente. CiberMeal será un software el cual, una vez contratado, permitirá el acceso de un médico, un dietista (tan solo encargado de realizar cambios en las dietas y crear nuevas dietas), y del paciente, este software permitirá la comunicación entre los usuarios y la fácil gestión de la dieta que el paciente esté realizando o vaya a realizar.

**1.2 Ámbito del proyecto y objetivos**

**1.2.1 Funciones principales**

Las funciones de la aplicación serán:

Creación de dietas personalizadas: El dietista, ciñéndose a las limitaciones impuestas por el médico y el paciente, creará una dieta única para el paciente en cuestión, la cual siempre estará abierta a modificaciones.

Seguimiento médico: El médico podrá mantener un seguimiento sencillo del paciente centrado en los aspectos alimenticios, así como los objetivos de la dieta, masa muscular, IMC, peso, etc.

Modificación de dietas: Si el médico lo cree necesario o si el paciente lo solicita, el dietista podrá realizar modificaciones en los elementos de la dieta, si el médico está satisfecho con los cambios, estos se registrarán en la dieta a la que el paciente podrá tener acceso.

Bibliotecas para consultas de alimentos: Todos los usuarios tendrán acceso a la biblioteca de la base de datos y utilizando las herramientas del buscador podrán consultarse las propiedades de cada alimento y las recetas de cada plato.

Herramientas para cálculo de calorías: Cualquiera de los usuarios puede utilizar calculadoras las cuales, utilizando los buscadores de platos y alimentos, calculen las calorías totales de los elementos seleccionados.

Consulta de Dietas: El paciente puede consultar la dieta que está realizando en ese momento, se le presentarán con forma de calendario, y el tamaño dependerá de la longitud de la dieta en cuestión.

**1.2.2 Aspectos de rendimiento**

Para evitar sobrecargas de trabajo, CibelMeal ha decidido contratar los servicios de un servidor diferente por cada hospital.

El único problema real que podría establecerse debido al rendimiento es a la hora de almacenar los datos en la Base de Datos, se ha propuesto el comprar un disco duro de 100 Terabytes por cada servidor contratado, a pesar de las apariencias, cuando se miles de pacientes por cada hospital, con varias dietas cada uno, el almacenamiento se convierte en un serio problema, por lo que se ha decidido utilizar elementos de compresión de datos, los cuales tan solo se podrán recuperar bajo demanda, cuando el usuario ha estado inactivo durante más de un año, o si la dieta se ha descartado.

El tiempo de respuesta del servidor, siempre y cuando no provoque conflictos no será un problema, debido a que servicio de nuestro programa no es inmediato, sino que se lleva a cabo mediante solicitud. De todos modos, si el servidor pone en cola el servicio demandado de cualquiera de los usuarios durante más de dos minutos, los cambios se ejecutarán en un servidor backup el cual realizará los cambios automáticamente en cuanto la cola se vacíe. Esto puede provocar varios minutos de espera para que el mensaje o la modificación lleguen a su destino, pero no es un problema a tener en cuenta por el formato de nuestro programa.

**1.2.3 Restricciones y técnicas de gestión**

● Las instalaciones informáticas son responsabilidad del Hospital, si el hospital no está equipado con el equipo necesario CiberMeal no se hace responsable.

● Mantenimiento de los servidores.

● Mantenimiento de una conexión a internet constante de al menos 100MB/s, para ello se contratará un servicio de cable y además uno vía satélite como backup.

Técnicas de gestión: Un modelo intermedio entre Descentralizado Democrático y DC.

**1.3 Modelo de proceso**

Debido a la forma en la que nuestro grupo está organizado, y a la naturaleza de nuestro proyecto, hemos decidido mantener el modelo que hemos mantenido hasta ahora, el modelo de Cascada.

**2. Mecanismos de Calidad y Control**

**2.1. Gestión de Calidad**

En un proyecto centrado en varios clientes en diferentes situaciones (Hospital, oficina, el hogar, etc.) es muy importante mantener siempre una sincronización entre los componentes y ratio de refresco muy frecuentes, debido a que el contacto entre los usuarios, la Base de Datos y el servidor es internet.

Los Gestores de Sección son los encargados de realizar los controles de calidad de sus respectivas secciones, cada cierto tiempo (a elección del gestor, normalmente antes de las reuniones de gestores) estos entraran en contacto con sus subordinados y pondrán una fecha para la cual todos los miembros de su sección tendrán que subir todo lo que hayan hecho organizado según se haya decidido en la última reunión, ese mismo día el gestor recopilará todos estos elementos de trabajo y los revisará para comprobar si hay algún retraso o si no es satisfactorio el resultado.

Estas medidas permitirán a los gestores documentar la evolución de las secciones y coordinar el trabajo para la siguiente entrega con mayor eficiencia, así como poder entrar en contacto con un empleado el cual no está cumpliendo con el trabajo que se le ha sido asignado, y tratar el problema.

Las reuniones de los gestores serán de aproximadamente una hora y media, y se realizarán semanalmente.

-Método de IS utilizado para el desarrollo de este software: Proceso Unificado.

**2.2 Gestión y control de cambios**

Cada gestor de sección se encargará de asegurarse de que los documentos y elementos de desarrollo del software mantengan una guía de estilo establecida por el mismo.

El programa ideal para la transferencia de documentación, tras probar varios programas a lo largo del primer cuatrimestre, ha sido DropBox, un software el cual almacena los últimos cambios realizados en una carpeta compartida, y sincroniza a través de internet todos a todos los usuarios.

En nuestro caso hemos estado realizando los cambios y almacenándolos en carpetas individuales antes de trasladarlos (una vez finalizados) a la carpeta común, donde se recogía el trabajo total. Los cambios realizados se apuntaban en la hoja del subsistema, junto con la autoría del que realizó el cambio y la fecha.

También barajamos la opción de utilizar un software de gestión conocido como Gitt (o su extensión gráfica”TostoiseGit”), pero nos pareció innecesariamente complejo para un proyecto tan pequeño.

En pocas palabras, creemos que el sistema de control de cambios que estamos utilizando, separando los cambios en documentos de subsistemas, y los cambios mayores en un documento general de todo el programa es el más sencillo y rápido.

**3. Plan de Gestión de Riesgos**

**3.1 Lista de posibles riesgos:**

-Organización del proyecto.

-Desincronización o pérdida de datos.

-Falta de conocimientos para llevar adelante el proyecto.

-Estimación poco realista en la gestión temporal.

-Bajas imprevistas del personal (caos).

-Enfermedades o indisponibilidad de algunos miembros (del personal).

-Perdida del trabajo actual (no hay copia de seguridad).

-Cambios en el diseño, por aumento de requisitos, que podrían descolocar el proyecto (rehacer).

-Ineficiencia del código (estructura) que se está implementado.

**3.2 Análisis de los riesgos**

-Organización del proyecto: El problema con el que más nos hemos encontrado, sobre todo debido a la falta de experiencia previa, ha sido la organización del equipo. La posibilidad de que este problema suceda es media.

-Desincronización o pérdida de datos: Una programación incorrecta del servidor, un problema de internet o un error de hardware puede llevar a una pérdida de datos o una caída de los servidores. La posibilidad de que este problema suceda es media.

-Estimación poco realista en la gestión temporal: Debido a la inexperiencia del equipo, pueden establecerse estimaciones incorrectas, lo cual puede hacer que se retrasen partes del programa. La posibilidad de que este problema suceda es media.

-Bajas imprevistas del personal: Debido a razones personales, algún miembro del equipo se puede ver obligado a abandonar permanentemente el Equipo. La posibilidad de que este problema suceda es baja.

-Enfermedades o indisponibilidad de algunos miembros (del personal): Bajas por enfermedad, ingreso hospitalario, u otras razones personales pueden suceder de forma imprevista, retrasando el trabajo del empleado durante toda esa jornada. La posibilidad de que este problema suceda es media.

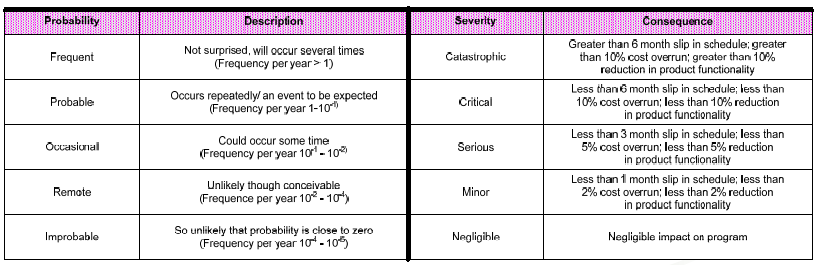
-Perdida del trabajo actual (no hay copia de seguridad): Debido a errores del equipo puede darse la situación en la que partes del trabajo recopilado se pierda, siendo imposible de recuperar. Aun así, este riesgo es muy improbable, debido a que existen varios lugares donde se guarda el trabajo, y varias formas de recuperar un punto del BackUp. La posibilidad de que este problema suceda es muy baja.

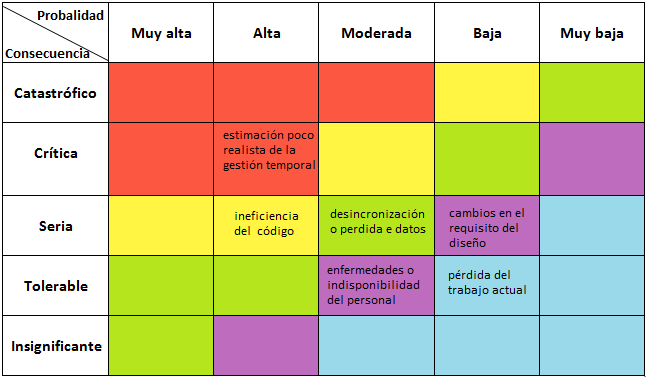
-Cambios en el diseño, por aumento de requisitos, que podrían descolocar el proyecto: Si el cliente decide cambiar el proyecto, o si en una reunión se decide que interesaría cambiar una parte del proyecto, rehacer todo el material necesario llevará tiempo de trabajo. La posibilidad de que este problema suceda es baja.

-Ineficiencia del código (estructura) que se está implementado: Si por problemas de diseño o errores en la programación la eficiencia en la ejecución del programa no es tolerable, el equipo tendría que reunirse y dedicar tiempo extra a la revisión de esa sección del programa. La posibilidad de que este problema suceda es alta (Debido a la falta de experiencia).

|  |  |  |
| --- | --- | --- |
| **Riesgo** | **Probabilidad** | **Efecto** |
| Organización del desarrollo del proyecto | Moderada | Tolerable |
| Estimación poco realista de la gestión temporal | Alta | Crítica |
| Enfermedades o indisponibilidad del personal | Moderada | Tolerable |
| Bajas imprevistas del personal | Baja | Seria |
| Cambios en el requisito del diseño | Baja | Seria |
| Ineficiencia del código (estructura) que se está implementado | Alta | Seria |
| Desincronización o perdida e datos (Base de Datos desactualizada) | Moderada | Seria |
| Falta de conocimientos para llevar adelante el proyecto | Moderada | Tolerable |
| Pérdida del trabajo actual (no hay copia de seguridad o no está actualizada) | Baja | Tolerable |

**3.3 Priorización del riesgo. Con Tabla del SQAS-SEI**

****



**3.4 Planificación del control de riesgos (plantilla RSGR)**

**3.4.1 Introducción**

En un proyecto (en nuestro caso El Proyecto Ciber Meal) que ha ido creciendo según se añadían modulo y características puede existir unos riesgos importantes como por ejemplo la organización del proyecto, abandono del personal o bajas por salud, la ineficiencia del código, estimación poco realista de la gestión temporal, desincronización o pérdida de datos, que desataría un caos en el desarrollo del proyecto. Las tareas asignadas y previstas su terminación para una determinada fecha no llegarían a tiempo a manos del cliente, ya sea porque el equipo carezca de tiempo para llevar acabo otras tareas, o porque el presupuesto excedería con la contratación personal nuevo.

**3.4.2 Priorización de riesgos del proyecto**

De acuerdo a los riegos antes mencionados, hemos establecido un orden de prioridad descendente con respecto a la tabla del SQAS-SEI:

1. Estimación poco realista de la gestión temporal.

2. Ineficiencia del código (estructura) que se está implementado.

3. Desincronización o pérdida de datos.

4. Abandono del personal o bajas por salud.

5. Organización del proyecto.

**3.4.3 Reducción, supervisión y gestión del riesgo**

Según las alternativas de acción a los riegos, los riesgos se pueden tratar como sigue:

* Evitar el riesgo
* Control de riesgo
* Asumir el riesgo
* Transferir el riesgo

Para tratar el riesgo de **estimación poco realista de la gestión temporal,** podríamos determinar una gestión más disciplinada de las fechas y reuniones del equipo a lo largo del proyecto (control de riesgo).

Otro riesgo no menos importante **la ineficiencia del código,** recurriremos a nuestros conocimientos en equipo cada vez que tengamos que implementar un código que requiera algoritmos complicados (plan de prevención).

La **desincronización o pérdida de datos** es un riesgo que asumimos puesto que lo máximo se podemos hacer es cerciorarnos de mantener nuestras copias de seguridad lo más actualizadas posible.

Un riesgo que no podemos evitar sería el **abandono del personal o bajas por salud**, solo nos queda asumirlo y reestructurar el reparto de trabajo con los miembros activos del equipo.

Para evitarla **desorganización del proyecto** podemos estructurar la directiva del equipo, establecer una serie de normas y mejorar la comunicación entre los miembros del equipo.

**4. Recursos del proyecto**

**4.1 Personal**

**4.1.1 Equipo de Desarrollo**

El grupo CiberMeal consta de siete miembros, repartidos de la siguiente forma:

* 2 revisores de CUs su labor es asegurarse de que los casos de uso con coherente entre ellos.
* 2 diseñadores de la aplicación cuya labor es encargarse de que el entorno grafico sea coherente y cómodo para el usuario.
* 2 encargados de diseño encargados de mantener orden en la estructura de programación.
* 1 gestor de programación el cual se ocupará de dirigir al resto de programadores, aclarar dudas y ayudar a los nuevos en programación y diseño.
* 5 programadores.

Los programadores se solaparán con otras funciones debido a la falta de personal.

**4.2 Hardware**

**4.2.1 Hardware de desarrollo**

El hardware utilizado para el desarrollo de la aplicación serán ordenadores sencillos de sobremesa, los servidores se simularán.

**4.2.2 Hardware del cliente**

Debido a que los servidores se encontraran en las instalaciones de CiberMeal, los usuarios tan solo necesitarán un ordenador de sobremesa y una conexión a internet de alta velocidad estable.

**4.3 Software**

**4.3.1 Software de desarrollo**

* **Eclipse:** Herramienta versátil de programación avanzada en Java, JavaScript, etc. Y compatible con elementos Web y SQL.
* **DropBox:** Herramienta la cual facilita tremendamente el compartir grandes cantidades de archivos entre un número limitado de personas.
* **Microsoft PowerPoint, Microsoft Word, Microsoft Project:** Documentación.
* **Photoshop CS6:** Desarrollo gráfico de la aplicación.

**4.3.2 Software del cliente**

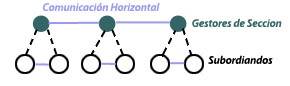
Las aplicaciones de software necesarias para poder ejecutar la aplicación en cualquier ordenador con cualquier usuario son las siguientes:

Java, un explorador de internet actualizado a su última versión (Chrome, Mozilla o iE9 o superior), y gracias al uso del Java, será compatible con cualquier sistema operativo.

**5. Organización de personal**

**5.1 Forma de organización del equipo**

El modelo que se ha decido utilizar es uno basado en los modelos establecidos por Mantei, un punto intermedio entre Descentralizado Democrático (DD) y Descentralizado Controlado (DC).

El grupo tendrá “Gestores de Sección”, los cuales se encargarán cada uno de su campo, y las reuniones y decisiones solo se decidirán entre ellos, la diferencia con el DC es que no existe una figura que gestione todos los grupos, sino que los gestores de cada grupo deciden las cosas de forma democrática. 

Los subordinados de los grupos tendrán una comunicación horizontal, así como los gestores de sección.

Se ha elegido este formato debido a que algunos de los miembros tienen poco tiempo para el proyecto, por lo que se nombrarán como gestores de sección aquellos que puedan estar más envueltos con el mismo, agilizando la toma de decisiones y efectividad de comunicación drásticamente.

**-Razones:**

En principio se pensaba adoptar el modelo DD, pero durante la última mitad del año 2013 comenzaron a presentarse muy serios problemas de comunicación y organización. Los deberes se asignaban de forma arbitraria debido a la falta de tomas de decisiones criticas durante las reuniones y su calidad dejaba que desear en muchos casos, cargando del doble de trabajo a unos pocos.

**-Reparto:**

\*Álvaro Vargas:

Gestor de Documentación.

\*Guillermo:

Documentación, Programador.

\*Wu Liu (Luis):

Revisor de CUs.

\*Daniel:

Gestor de coherencia de los CUs, Programador.

\*Gerald Lima:

Revisor de CUs, Programador.

\*Marta Rodenas:

Documentación, Programador.

\*Javier Garcia:

Gestor de Programación.

*Gestor de Documentación*: Persona encargada de la gestión de la sección de documentación.

*Gestor de Coherencia de los CUs:* Persona encargada de que los Casos de Uso (CRC, descripciones, interacción con otros CUs) tengan coherencia en el conjunto del programa.

*Gestor de Programación*: Persona encargada de diseñar la organización y formatos del código, además de revisar y reunir el trabajo de los programadores.

*Documentación:* Persona encargada de la documentación de legal y técnica del programa.

*Revisor de CUs:* Persona encargada de asegurarse de que los CU sigan siendo compatibles con el resto del programa.

*Programador:* Persona encargada de programar la aplicación.

**5.2 Estimaciones**

**‐ Dificultad del problema:** Media, lo más difícil será la sincronización entre el servidor y los diferentes clientes.

**‐ Duración del equipo:** Se espera que el equipo permanezca en funcionamiento con todos sus integrantes hasta la fecha de entrega del proyecto.

**‐ Fecha de entrega:** Mayo de 2014.

**‐ Calidad y fiabilidad:** Al principio el equipo solo se centrará en la plataforma PC, por lo que la calidad gráfica puede ser cómoda tanto de realizar como de modificar a petición del cliente, la fiabilidad del producto dependerá de la calidad de internet del usuario, por lo que se le impondrá al cliente (hospital en este caso) una calidad mínima, si el fallo es por la conexión, CiberMeal no se hace responsable legal de la posible pérdida de información, vital o no.

**‐ Comunicación:** Se utilizarán varios medios para mantener una comunicación fluida (Watsapp, Skype y llamadas telefónicas), además de dos reuniones semanales de, por lo menos, todos los Gestores de Sección.

**6. Planificación temporal**

**6.1 Estructura de descomposición del trabajo o Planificación temporal**

La forma de organización del proyecto completo será:

* Comunicación

Realización de la documentación, planificación de diseño, requisitos y definición de los usuarios.

* Modelado

Diseño final del entorno gráfico (pantallas) y las características de los diferentes estilos del programa.

* Despliegue

Recopilación de todo el trabajo realizado hasta el momento en un archivo SRS, el cual servirá como guía para el resto del proceso de diseño.

* Construcción

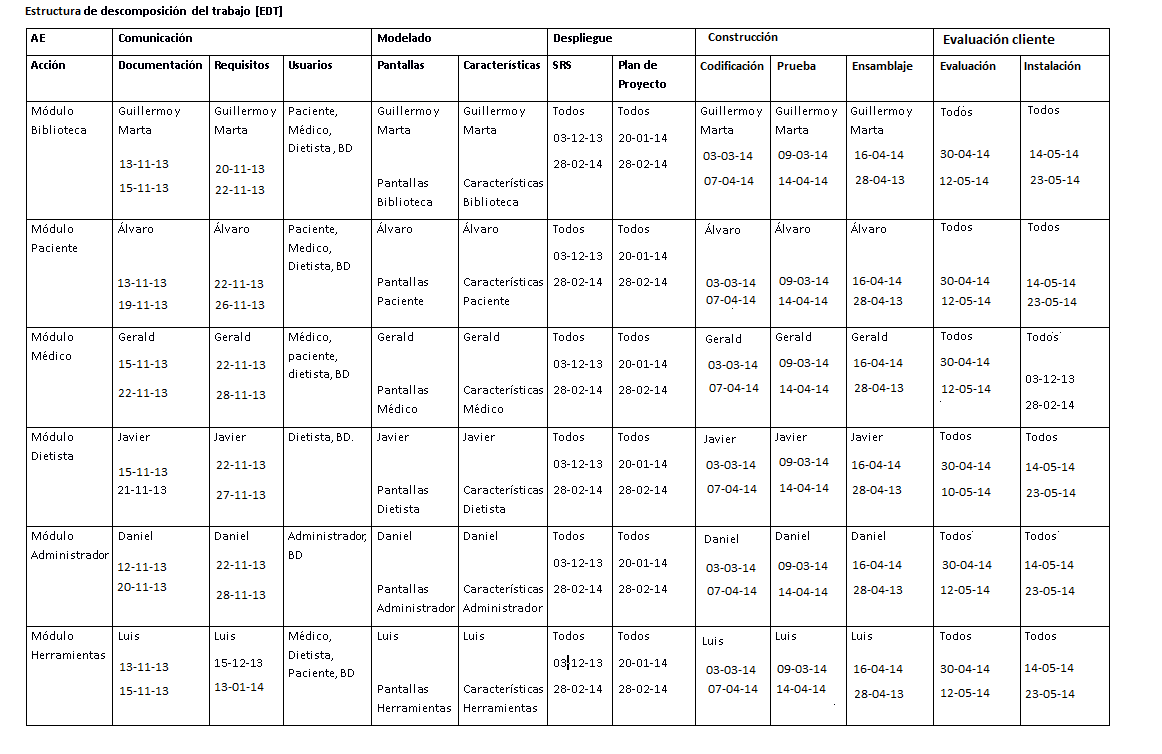
Esta constará de la codificación (programación), prueba (test) y ensamblaje, en el cual se une todo el trabajo hecho hasta el momento.

* Evaluación cliente

El cliente evalúa el producto, y si este está satisfecho, se saca adelante.

Estas fases no se realizarán de forma independiente, algunas se solaparan, o se tendrán que repetir (retrasando el resto del proyecto) como por ejemplo la construcción, si los testeos no son satisfactorios o si se decide hacer un cambio de diseño, habría que repetir todo el proceso.

**6.2 EDT**

****

**6.3 Red de tareas**

La red de tareas nos permite calcular el camino crítico del desarrollo, debido a la organización por secciones y gestores de sección (Ver organización) el camino crítico se determinará por la sección que lleve más tiempo (en las partes que se lleven a cabo en el paralelo), en este caso será el de desarrollo.

**7. Apéndice**

**7.1 Fuentes:**

**Fuente de información para la realización del documento:**

Gestión de proyectos: <https://cv2.sim.ucm.es/moodle/file.php/35573/C1-Teoria/Tema-06-Gestion-de-proyectos.pdf>

Gestión de riesgos: <https://cv2.sim.ucm.es/moodle/file.php/35573/C1-Teoria/Tema-08-Gestion-de-riesgos.pdf>

Gestión de equipos: <https://cv2.sim.ucm.es/moodle/file.php/35573/C1-Teoria/Tema-05-Gesti_n-de-equipos.pdf>

Ministerio de salud y nutrición: <http://www.aesan.msc.es/>

Naturhouse:<http://www.naturhouse.com/es>

**7.2 Glosario:**

DD: *Descentralizado democrático*

DC: *Descentralizado controlado*

CU: *Caso de Uso*

EDT: *Estructura de Trabajo*

SRS: *Software Requirements Specification*

RSGR: *Plan de Reducción, Supervisión y Gestión del Riesgo.*

IS: *Ingeniería del Software*