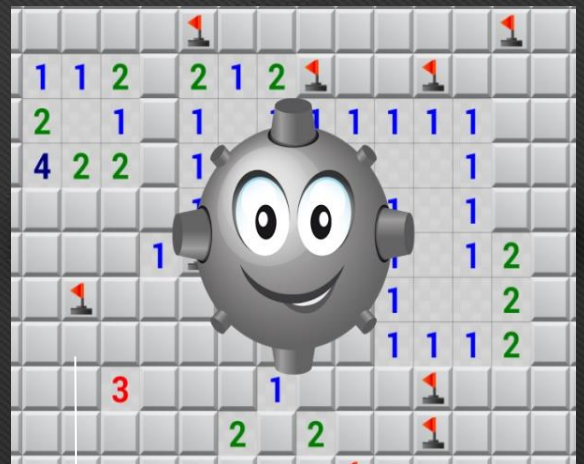


PLAN DE GESTIÓN DE PROYECTO

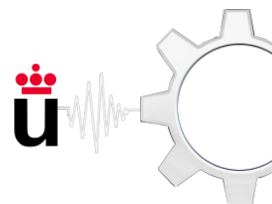
Buscaminas



Álvaro Segura Manzanares
Jorge García Ranera
Carlos Graña Muñoz
Adina Georgiana Onofrei
*Ingeniería
Informática + ADE*

Grupo E

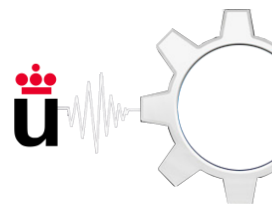
25/04/2018



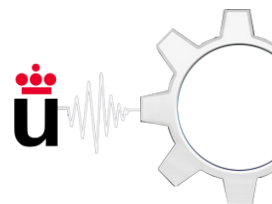
PLAN DE GESTIÓN DE PROYECTO

Tabla de contenidos

Hoja de revisión.....	3
Lista de figuras.....	5
Prefacio	5
1. Introducción.....	7
1.1 VISIÓN GENERAL DEL PROYECTO.....	7
1.2 PRODUCTOS FINALES	8
1.3 EVOLUCIÓN DEL PLAN DE PROYECTO	9
1.4 DOCUMENTOS DE REFERENCIA.....	10
1.5 DEFINICIONES Y ACRÓNIMOS.....	10
2. Organización del proyecto.....	11
2.1 MODELO DE PROCESOS.....	11
2.2 ESTRUCTURA ORGANIZATIVA.....	16
2.3 FRONTERAS E INTERFACES ORGANIZATIVAS.....	18
2.4 RESPONSABILIDADES.....	19
3. Procesos de gestión.....	21
3.1 OBJETIVOS Y PRIORIDADES DE GESTIÓN	21
3.2 SUPOSICIONES, DEPENDENCIAS Y RESTRICCIONES.....	21
3.3 GESTIÓN DE RIESGOS.....	23
3.4 MECANISMOS DE SUPERVISIÓN Y CONTROL	24
3.5 PLAN DE PERSONAL	25
4. Proceso Técnico	26



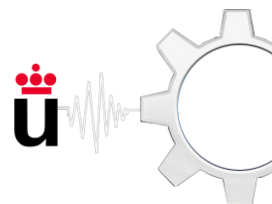
4.1	METODOLOGÍA, TÉCNICAS Y HERRAMIENTAS	26
4.2	DOCUMENTACIÓN SOFTWARE.....	27
4.3	FUNCIONES DE APOYO AL PROYECTO	27
5.	Plan de desarrollo	29
5.1	PAQUETES DE TRABAJO.....	29
5.2	DEPENDENCIAS	30
5.3	RECURSOS	33
5.4	PRESUPUESTO Y DISTRIBUCIÓN DE RECURSOS.....	34
5.5	CALENDARIO.....	35



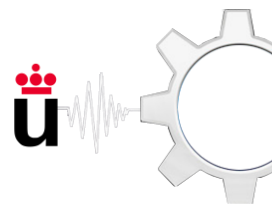
Hoja de revisión

INTEGRANTE	FECHA	DESCRIPCIÓN
Álvaro Segura	23/03/2018	Inicio Plan de Proyecto – Estructura
Adina Onofrei	25/03/2018	Plan de Proyecto – Punto 1
Álvaro Segura	27/03/2018	Inicio Plan de Gestión de Configuración – Estructura y punto 1
Jorge García y Carlos Graña	27/03/2018	Lectura inicial, verificación y creación del menú superior
Adina Onofrei	29/03/2018	Niveles de dificultad
Jorge García	29/03/2018	Mostrar cantidad de minas y tiempo de partida
Álvaro Segura	29/03/2018	Plan de Gestión de Configuración – Punto 2
Jorge García	30/03/2018	Añadir nombre de jugador
Adina Onofrei	2/04/2018	Plan de Proyecto – Punto 2
Carlos Graña	3/04/2018	Plan de Proyecto - puntos 1 y 2
Jorge García	4/04/2018	Guardar mejores tiempos, mostrar los mismos y permitir guardar tiempos en fichero
Álvaro Segura	5/04/2018	Plan de Gestión de Configuración – Punto 4
Carlos Graña	5/04/2018	Plan de Proyecto - punto 3
Jorge García	6/04/2018	Permitir guardar la partida en un fichero y consultar la última partida jugada
Carlos Graña	6/04/2018	Plan de Proyecto – Actualización punto 1
Álvaro Segura	6/04/2018	Plan de Gestión de Configuración – Punto 4

PLAN DE GESTIÓN DE PROYECTO



Adina Onofrei	6/04/2018	Plan de Proyecto – punto 4 y 5
Carlos Graña	9/04/2018	Diagrama de Gantt en formato Excell
Jorge García	9/04/2018	Permitir recuperar la partida guardada de un fichero
Álvaro Segura	9/04/2018	Plan de Gestión de Configuración – Punto 5
Adina Onofrei	9/04/2018	Diagrama de Pert
Carlos Graña y Adina Onofrei	11/04/2018	Revisión Plan de Proyecto y actualización del mismo
Álvaro Segura	11/04/2018	Plan de Gestión de Configuración – Punto 6 y revisión del plan
Jorge García	11/04/2018	Menú para las diferentes opciones del juego
Álvaro Segura	13/04/2018	Inicio PowerPoint y actualización de Plan de Gestión de Configuración de Software
Jorge García	13/04/2018	Validación de código actualizado
Carlos Graña	13/04/2018	Modificación del Plan de Proyecto - Punto 4
Álvaro Segura	16/04/2018	Plan de Proyecto – Modificación punto 5.3 Plan de Gestión de Configuración de Software – Punto 3 y revisión final
Jorge García	16/04/2018	Revisión final código
Adina Onofrei	16/04/2018	Plan de Proyecto – revisión final
Todos	17/04/2018	PowerPoint
Carlos Graña	18/04/2018	Revisión final PowerPoint



Lista de figuras

Figura 1 - Diagrama de Gantt.....	13
Figura 2 - Diagrama de Gantt.....	14
Figura 3 - Diagrama de Gantt.....	14
Figura 4 - Diagrama de Pert.....	15
Figura 5 - Esquema de Reparto de actividades	17
Figura 6 - Esquema de Fronteras e Interfaces.....	19
Figura 7 - Esquema Paquetes de Trabajo	29
Figura 8 - Tabla de Dependencias.....	30
Figura 9 - Tabla de Actividades Principales.....	31
Figura 10 - Diagrama de Pert.....	32
Figura 11 - Calendario de Gantt.....	35

Prefacio

El proyecto indicado por el cliente consta de una serie de partes que deberán ser realizadas por el equipo E de desarrollo de software. Tiene como objetivo la mejora de las funcionalidades del juego *Buscaminas* según las especificaciones del cliente junto con los informes pertinentes para su correcta documentación y ejecución.

Por lo tanto, para conseguir este propósito, se llevarán a cabo un conjunto de acciones necesarias para alcanzar los objetivos y necesidades establecidas en el enunciado de la practica obligatoria. Estos elementos son:

- Mejoras las funcionalidades del programa “Buscaminas”
 - Selección del nivel de dificultad que solicite el usuario o configuración de su propio nivel.
 - Implantación de un menú superior que contenga información de distinta índole. Podemos destacar las siguientes:
 - Contador de minas marcadas.



- Tiempo del juego.
 - Opción de reiniciar la partida.
 - Opción de crear una nueva partida indicando consigo el nivel en el que se quiere jugar.
 - Guardar la partida antes de su finalización.
 - Tabla de los mejores tiempos de cada nivel con el nombre del jugador que haya alcanzado dicho tiempo récord.
 - Archivar el tiempo la partida y el nombre del jugador en el caso de que se haya superado el tiempo récord del nivel jugado.
 - Almacenamiento en un fichero de texto de los mejores tiempos en cada uno de los niveles, así mismo, el nombre del jugador.
 - Almacenamiento en un fichero de texto de una partida sin terminar que quiera guardar el usuario.
-
- Plan de gestión de configuración de software tiene como objetivo maximizar la producción minimizando los errores aplicado a cada fase del proceso de software. Por ello, este plan trata de gestionar los cambios a lo largo de toda la vida del software.
 - Plan de mantenimiento, hace referencia al conjunto de cambios realizados en un software determinado después de su implantación.



1. Introducción

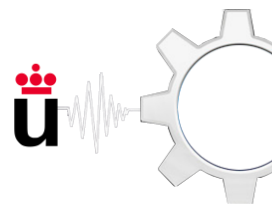
1.1 VISIÓN GENERAL DEL PROYECTO

Como se ha indicado en el prefacio, el objetivo de este trabajo consiste en mejorar las funcionalidades del juego *Buscaminas* a partir de un código ya proporcionado. Para conseguir estas mejoras seguiremos los siguientes pasos:

1. Verificación del código proporcionado: se realizará una lectura activa y se probará el funcionamiento del mismo para comprender mejor su funcionalidad.
2. Validación del código: cada uno de los integrantes dará el visto bueno en relación a la comprobación de los elementos del *Buscaminas* básico proporcionado.
3. Realización de las mejoras
4. Comprobación de errores y mejora del código implementado.
5. Puesta en común y acciones de mejora: en el momento en el que tengamos realizadas todas las funcionalidades requeridas, todos los integrantes del grupo nos reuniremos para comprobar el correcto funcionamiento de las acciones integradas y posibles ideas para mejorar su implantación dentro del código.

Como se acaba de indicar, este proyecto se realizará siguiendo un proceso ordenado y lógico de acción para alcanzar los requisitos y las funcionalidades requeridas por el cliente. Una vez finalizada la construcción de la arquitectura del juego, pasaremos a la fase de mantenimiento en la que iremos resolviendo los problemas que surjan tanto a corto plazo como a largo.

Para poder simplificar el proceso, utilizaremos la plataforma GitHub que nos permitirá tener un repositorio común en el que podamos subir todas las



modificaciones que hagan los integrantes de nuestro grupo además de poder trabajar de forma remota y poder llevar a cabo actividades de control de la calidad.

Por otro lado, utilizaremos la aplicación *NetBeans* para programar las funcionalidades que se nos solicitan e ir probando los cambios que se hagan en el código.

1.2 PRODUCTOS FINALES

Como productos finales podemos identificar, en primer lugar, el Plan de Proyecto que se tratará del informe inicial que indica los elementos iniciales, la forma de implementarlos y el método que seguiremos para alcanzar los requisitos indicados por el cliente.

En segundo lugar, el Plan de Gestión de Configuración del Software con el que conseguiremos mantener la integridad de los elementos, garantizar que no se realizan cambios no controlados y que todos los participantes del proyecto disponen de la versión adecuada de los productos que manejan. De este modo, conseguiremos demostrar al cliente el proceso que hemos seguido y la forma en la que hemos implementado los cambios.

En tercer lugar, el programa *Buscaminas* con las mejoras solicitadas por el cliente definidas en los puntos anteriores.

En cuarto lugar, el Plan de Mantenimiento realizado una vez entregado el proyecto final al cliente y creado con el fin de establecer las prácticas, recursos y secuencias de actividades relevantes para mantener el producto.

Por último, la presentación PowerPoint que se usará en el día concreto para poder exponer el trabajo frente al cliente.

Su fecha de entrega corresponde al 25 de abril de 2018.



1.3 EVOLUCIÓN DEL PLAN DE PROYECTO

El plan de proyecto se llevará a cabo por el conjunto del grupo, cada uno se encargará de uno de los puntos de este informe. Para que uno de los integrantes pase a hacer su apartado, primero se evaluará y aprobará entre todo el grupo las modificaciones sobre este plan y, a continuación, la jefa del Plan de Proyecto, Adina Onofrei, dará el visto bueno en las modificaciones realizadas por los miembros del grupo. Todos estos cambios serán comunicados a través de dos canales distintivos:

- Puesta en común sobre los cambios realizados y confirmación por parte de la jefa de la sección.
- Plataforma *GitHub* en la que iremos subiendo las actualizaciones tanto de los informes como del código. Para simplificar el proceso de puesta en común y mejorar la comunicación entre los integrantes del equipo, cada uno, realizará un "clone" en el repositorio padre de nuestro proyecto. De esta forma, cada vez que se actualice uno de los informes a entregar o el código del juego se realizará un "commit" para guardar la modificación en el repositorio local, a continuación, hará un "push" para subir el archivo actualizado en el repositorio padre. Se utilizará el comando "pull" para que el resto de los integrantes del equipo tengan el archivo actualizado que se acaba de subir al repositorio.

Este plan deberá ser revisado al inicio de cada fase, modificado de acuerdo a lo necesario, aprobado y distribuido al equipo de proyecto.



1.4 DOCUMENTOS DE REFERENCIA

Los documentos sobre los que nos basaremos en el desarrollo de los informes son, principalmente, los siguientes:

- Norma IEEE 1058.1-1977 para la planificación de gestión de proyectos software
 - o Autor: Ismael Caballero Muñoz-Reja
 - o Organización que lo ha publicado:
- Gestión y dirección de proyectos software
 - o Autor: Sin identificar
 - o Organización que lo ha publicado: Universidad Rey Juan Carlos en el aula virtual

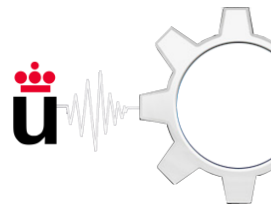
1.5 DEFINICIONES Y ACRÓNIMOS

IEEE: *Institute of Electrical and Electronics Engineers*

PGCS: Plan de Gestión de Configuración de Software

PDP: Plan de Proyecto

CC: control de calidad



2. Organización del proyecto

2.1 MODELO DE PROCESOS

Se establecerán una serie de relaciones, a lo largo del conjunto del proyecto, entre las actividades principales del proceso. Para conseguir este propósito, dividiremos en una serie de tareas el conjunto del proyecto con el fin de poder establecer estas relaciones.

De este modo, algunas de las actividades que identificaremos serán dependientes de otras, es decir, no se puede empezar otra actividad hasta que no se haya finalizado la otra. Haciendo referencia a nuestro proyecto, la actividad de validación del código no puede empezar hasta que no se haya terminado de verificar el código.

Por todo ello, dividiremos el proyecto total en una serie de actividades, que a su vez, éstas se dividirán en un determinado número de tareas para poder esquematizar el tiempo que se dedicará a cada tarea.

De esta forma, el proyecto se divide en:

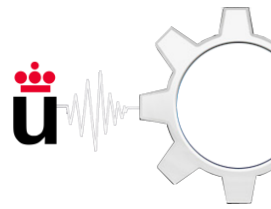
1. Creación de la documentación

Una vez completada esta tarea en el tiempo estimado por el equipo, se llevará a cabo la verificación de código ya que ésta depende de la creación de la documentación.

Tiempo total: 4 días

2. Verificación de código

Una vez realizada la reunión inicial del equipo se pasará a la mejora de las funcionalidades del juego. Una vez hecha la verificación inicial del código



desarrollada en el tiempo estimado, se podrá pasar a la validación del código por todo el equipo.

Tiempo total: 1 día

3. Validación de código

La validación del código se realizará dentro del plazo establecido se podrá empezar con el desarrollo de las funcionalidades indicadas por el cliente en el momento de la presentación del proyecto.

Tiempo total: 1 día

4. Versión actualizada

Esta actividad se divide en las siguientes tareas, para que funcione según las estipulaciones del cliente se pasará a la validación del código actualizado.

Tiempo total: 15 días

- 4.1 Añadir opción para reiniciar el juego → 1 día
- 4.2 Mostrar cantidad de minas → 1 día
- 4.3 Mostrar tiempo de partida → 1 día
- 4.4 Niveles de dificultad → 3,2 días
- 4.5 Guardar 10 mejores tiempos de cada dificultad → 1 día
- 4.6 Mostrar 10 mejores tiempos de cada dificultad → 1 día
- 4.7 Añadir un nombre de jugador → 1 día
- 4.8 Dar la opción de guardar el tiempo o no → 1 día
- 4.9 Permitir guardar la partida actual en un fichero → 3,2 días
- 4.10 Permitir consultar la última partida jugada → 1 día
- 4.11 Recuperar una partida guardada de un fichero → 1 día
- 4.12 Menú para acceder a las diferentes opciones del juego → 1 día

5. Validación de código actualizado

La finalización de esta actividad no implicará el inicio obligatorio de otra actividad.

Tiempo total: 1 día

5.1 Validación del código actualizado → 1 día

5.2 Pruebas del código → 1 día

6. Puesta en común del proyecto

La realización de esta actividad no implica el inicio de otra tarea.

Tiempo total: 1 día

La suma total del desarrollo del proyecto es de 19 días teniendo en cuenta los siguientes elementos:

- Número de actividades definidas anteriormente.
- Relación entre las actividades.
- Número de dependencias de cada actividad.

Para ilustrar de una forma más sencilla realizaremos un diagrama de Gantt con el que, planificaremos programaremos, de forma gráfica, las tareas a lo largo del periodo de desarrollo del proyecto. El diagrama resultante es el siguiente:

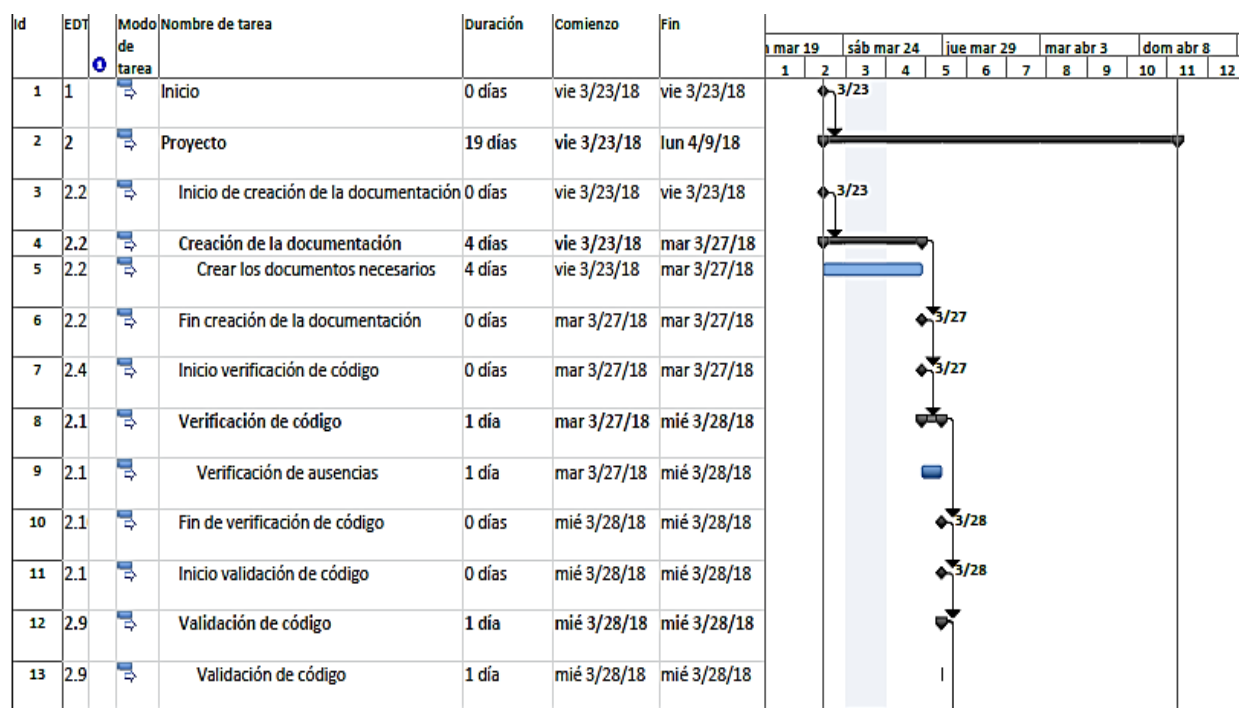


FIGURA 1 - DIAGRAMA DE GANTT



Por otro lado, se utilizará el diagrama de PERT con el objetivo de aclarar las dependencias entre una actividad y otra, el cual es el siguiente:

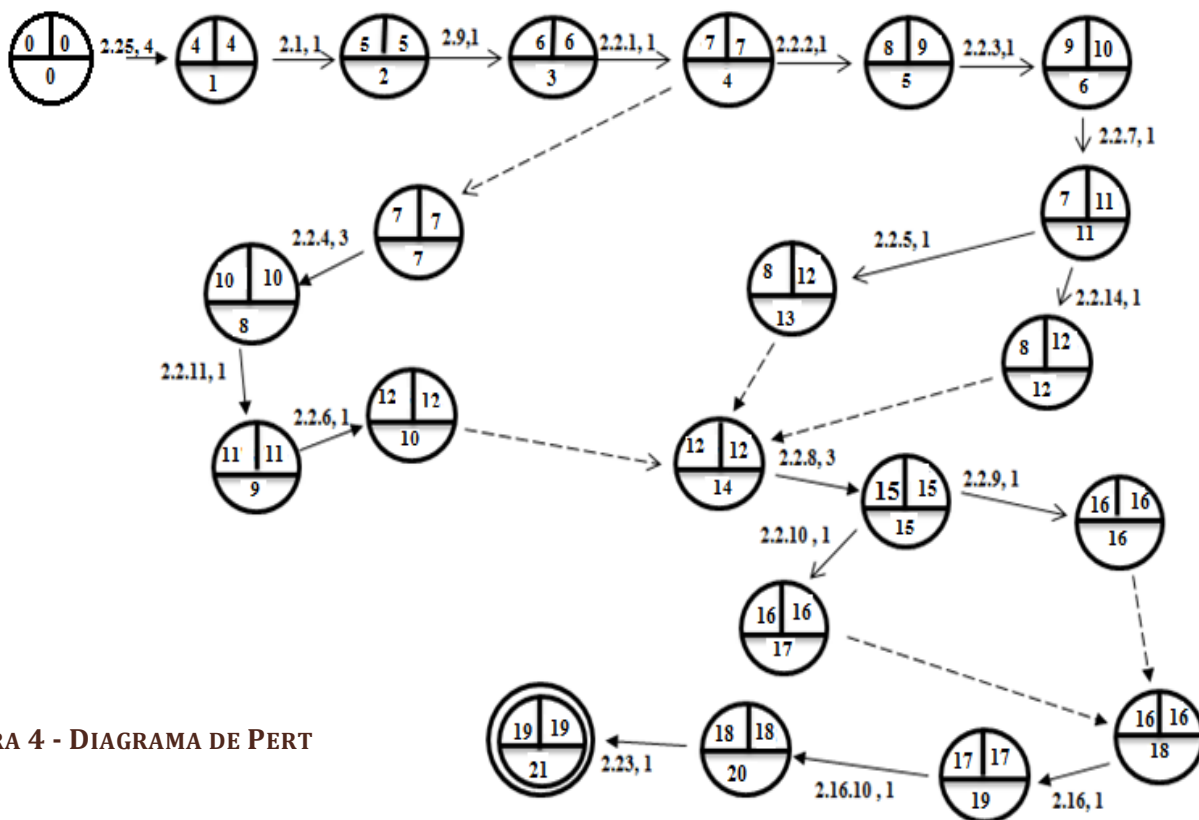
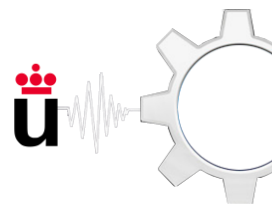


FIGURA 4 - DIAGRAMA DE PERT

El diagrama nos permitirá ver el tiempo *early*, el cual estará representado en la parte superior izquierda del círculo, que nos indicará el tiempo más temprano para comenzar la actividad, y el tiempo *late*, representado en la parte superior derecha del círculo, que nos indica el tiempo más temprano para finalizar la actividad.

Siguiendo el diagrama destacamos que las tareas que se realizan de forma paralela son:

- 2.2.1 (Añadir la opción para reiniciar el juego)
- 2.2.4 (Niveles de dificultad)



Lo mismo ocurre con las actividades:

- 2.2.2 (*Mostrar el número de minas*): representados en la parte derecha del diagrama
- 2.2.3 (*Mostrar el tiempo de la partida*), 2.11 (*Menú con las opciones del juego*) y 2.2.6 (*Añadir nombre de jugador*): representados en la parte izquierda del diagrama

Finalmente, las tareas relacionadas con el diseño del juego, opciones adicionales que no guarden relación con el tiempo de la partida y aquellas actividades relacionadas con guardar o mostrar los 10 mejores tiempos de cada partida, se juntarán en un mismo evento ya que representa la casi finalización de la versión actualizada del código (*suceso 14 en el diagrama*).

Una vez llegado a ese punto, lo restante representado en el PERT serán tareas relacionadas con el fichero que guardará las partidas (*sucesos comprendidos entre 14-17 incluidos*) y tareas relacionadas con la finalización del código, es decir:

- Suceso 19: representa la validación del código actualizado
- Suceso 20: pruebas del código actualizado
- Suceso 21: puesta en común del código

2.2 ESTRUCTURA ORGANIZATIVA

En la reunión inicial del grupo se ha decidido que, al ser un equipo tan reducido, las tareas que componen las actividades principales identificadas en el punto anterior se llevarán a cabo por todos los miembros del grupo indicando el jefe de cada una de las actividades principales del proyecto. Los jefes de cada sección son:

1. Jefa Plan de Proyecto: Adina Onofrei
2. Jefe Plan de Gestión de Configuración: Álvaro Segura
3. Jefe del desarrollo software del juego: Jorge García

4. Jefe de presentación frente a los clientes: Carlos Graña
5. Jefe de la entrega del contenido final del proyecto: Carlos Graña

Cabe destacar que cada uno de los integrantes se dedicará a una actividad en concreto en función de las cualidades en las cuales cada miembro resalta mejor. De este modo las actividades que componen el proyecto se han repartido en función de estas cualidades, para garantizar el desarrollo correcto del mismo con el objetivo de cumplir los plazos establecidos por el cliente, así como las funcionalidades requeridas por el mismo. Además, antes de que el jefe de la sección dé el visto bueno en las modificaciones realizadas se llevará a cabo una reunión grupal para revisar dichas mejoras y cambios.

El reparto de las actividades queda reflejado en el siguiente gráfico:

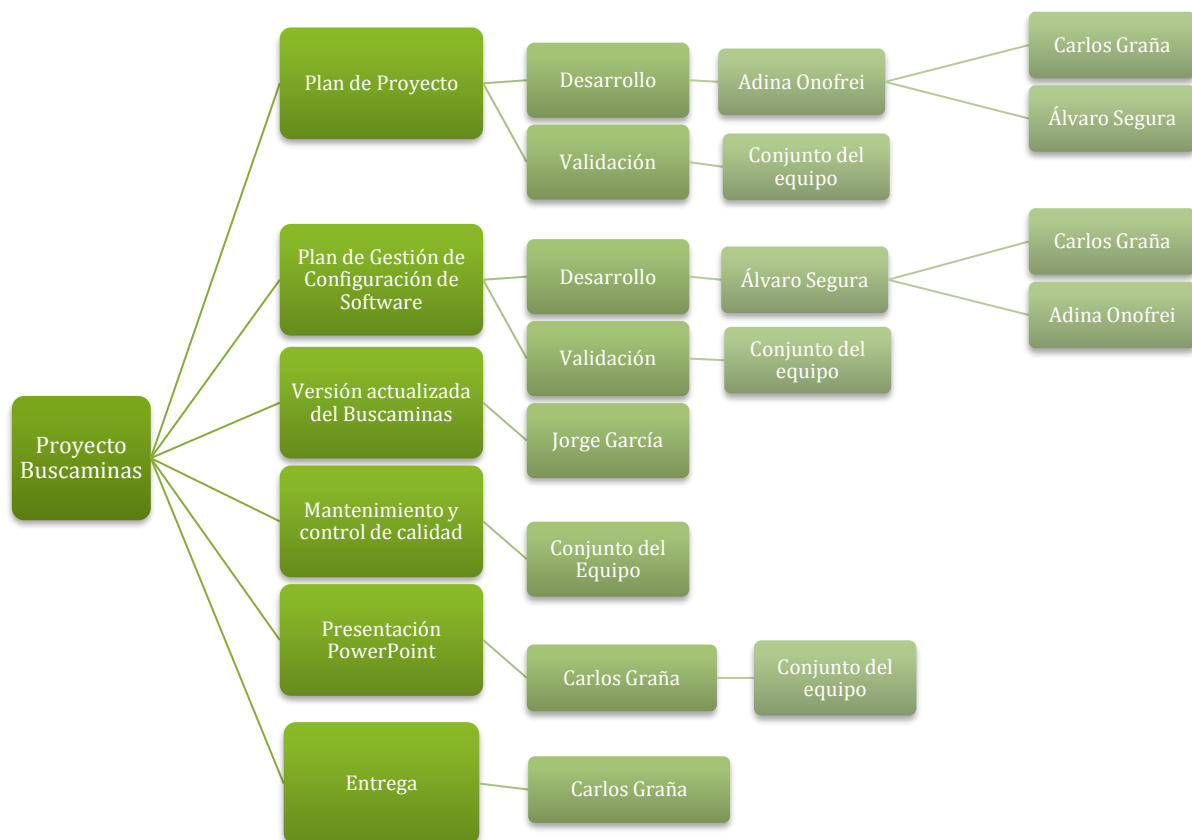
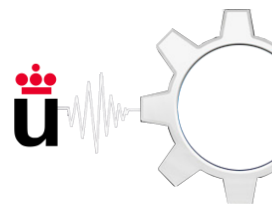


FIGURA 5 - ESQUEMA DE REPARTO DE ACTIVIDADES



2.3 FRONTERAS E INTERFACES ORGANIZATIVAS

En este caso, la organización encargada del proyecto se compone de un pequeño grupo de cuatro integrantes, en concreto el grupo E de la asignatura de Ampliación de Ingeniería de Software.

Todo el proyecto se realizará mediante contribuciones individuales y comunes a los distintos informes, código requerido por el cliente y los controles de calidad pertinentes seguido por la aceptación de los cambios por parte del jefe de sección. Por tanto, nos encargamos del proyecto como tal y la organización cliente, que en este caso serían todas aquellas personas que utilicen el juego y al ser un código abierto (subido a GitHub), será todo aquel que encuentre nuestro proyecto independientemente del sistema operativo que integre en su terminal, como, por ejemplo, Windows, Linux o MacOS.

En nuestro caso, hemos decidido que no es necesario utilizar ninguna organización subcontratada ya que en la fase inicial del proyecto cada integrante del grupo incluía una serie de cualidades que nos permitirá realizar todos los requisitos sin ayuda externa.

Por último, en la primera reunión de grupo se ha decidido llevar a cabo dos momentos concretos en los que se verificará y se validará las implementaciones de los requisitos establecidos por los clientes y son los siguientes:

- a. Inicio del proceso de programación: se realizará una verificación inicial del código proporcionado por el cliente para comprobar su correcto funcionamiento.
- b. Final del proceso de programación: se comprobará la implementación de los elementos requeridos y el grado de optimización de las mismas.

Por último, cabe destacar que el proyecto está relacionado con algunos agentes externos, los cuales quedan representados en el siguiente gráfico:

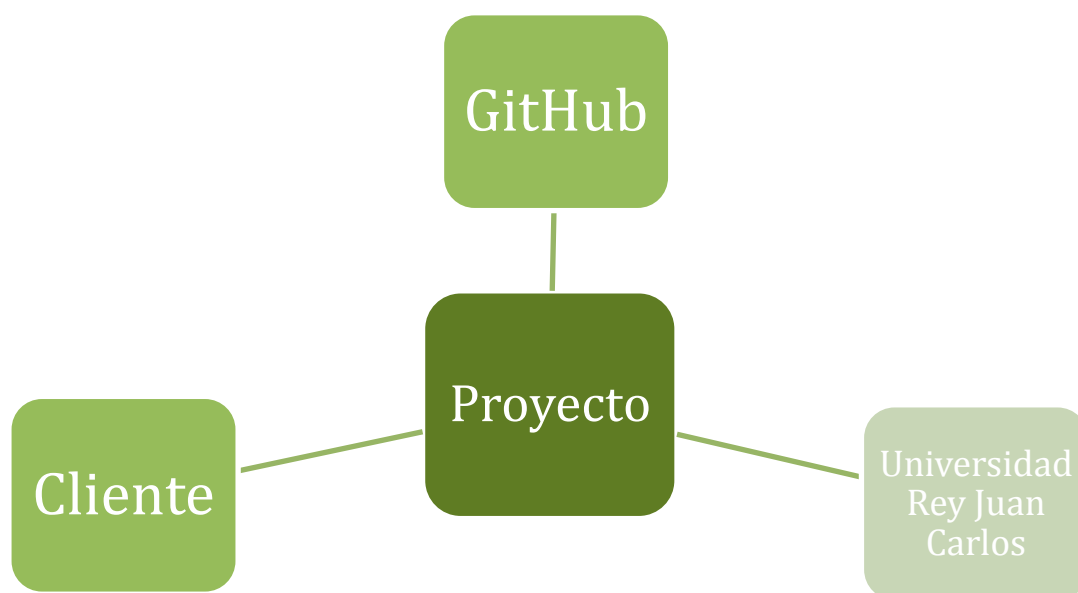
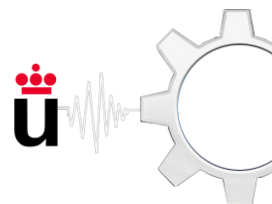


FIGURA 6 - ESQUEMA DE FRONTERAS E INTERFACES

2.4 RESPONSABILIDADES

En un primer momento, antes de comenzar con el proyecto, se decidió de forma conjunta la división del equipo en dos “sub-equipos” que se dedicasen a partes distintas del proyecto trabajando en paralelo para llegar a la fecha de entrega de forma holgada y con la posibilidad de realizar modificaciones en la implantación del código en el caso de que el cliente no estuviese de acuerdo.

Una vez propuesto el trabajo y las funciones requeridas por el cliente, se dividirá el equipo en dos. La primera parte estará compuesta por Jorge García, Jefe de desarrollo software del juego, encargado de implementar las funcionalidades requeridas por el cliente al tener una serie de capacidades más desarrolladas que el resto de los integrantes del equipo en este campo.



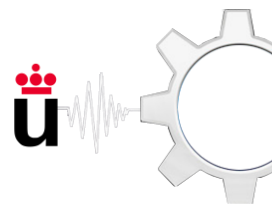
La segunda parte estará compuesta por el resto de los integrantes del equipo (Adina Onofrei, Álvaro Segura y Carlos Graña), los cuales serán los encargados de realizar el Plan de Proyecto y el Plan de Gestión de Configuración del Software los cuales permitirán simplificar el trabajo del grupo encargado de la programación de los requisitos especificados por el cliente.

Por último, la presentación PowerPoint necesaria para presentar el proyecto será realizada en mayor parte por Carlos Graña, aunque será completada por el resto de los integrantes en función de las partes desarrolladas en el proyecto.

Con todo ello, la siguiente enumeración mostrara de una forma rápida y esquemática las responsabilidades de cada integrante del grupo:

1. Mejora de las funcionalidades del juego *Buscaminas*: Jorge García
2. Presentación PowerPoint: Jorge García, Carlos Graña, Adina Onofrei, Álvaro Segura
3. Documentos
 - a. Plan de proyecto: Adina Onofrei, Álvaro Segura, Carlos Graña.
 - b. Plan de Gestión de Configuración: Adina Onofrei, Álvaro Segura, Carlos Graña.

Aunque haya actividades realizadas por varios integrantes, como por ejemplo, los informes o la presentación PowerPoint, se irán dividiendo en función de sus epígrafes para que cada uno de los miembros del grupo realicen el mismo trabajo. Además, cada una de estas actividades tendrá un jefe estipulado que será de dar el visto bueno final tras realizar la reunión de control de calidad en la que participan todos los integrantes del equipo. También, cabe destacar que, aunque cada uno se dedique a realizar una actividad en concreto, se ha decidido que todos nos ayudaremos a solventar las dudas o problemas que nos vayan surgiendo en el trascurso del desarrollo del proyecto. Esto permitirá hacer el trabajo más ameno e incluso favoreciendo el trabajo en grupo a la hora de generar ideas para implementar y solventar de la forma más sencilla, rápida y eficiente los problemas encontrados.



3. Procesos de gestión

3.1 OBJETIVOS Y PRIORIDADES DE GESTIÓN

Después de la reunión inicial en la que se nos presentó el trabajo, se decidió enfocar el proyecto mediante una mentalidad continua, es decir, encontrar un problema, generar las ideas para solventarlo, implementar de la solución, verificación y validación de la solución por todos los integrantes de equipo y, por último, el visto bueno por parte del jefe de la sección en la que se están desarrollando los cambios.

De esta forma, se planteó como prioridad absoluta el correcto funcionamiento de juego *Buscaminas* con las mejoras en la funcionalidad que solicitó el cliente. Todas estas funcionalidades han sido descritas en el apartado 2.1 de Plan de Proyecto de forma más extensa y descriptiva.

Además, se establecerá una serie de controles de calidad cada vez que se completen hitos esenciales para el correcto funcionamiento del programa.

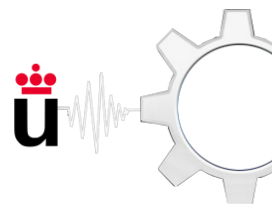
3.2 SUPOSICIONES, DEPENDENCIAS Y RESTRICCIONES

Una vez planteado el problema inicial, en la primera reunión, se identificaron una serie de suposiciones sobre la forma en la que se ejecuta el juego. Estas dependencias y restricciones son las siguientes:

- El jugador deberá de elegir un nivel de dificultad para poder jugar al juego. Podrá elegir entre los siguientes:
 - Principiante: tablero de diez filas por diez columnas con un número total de 10 minas.



- Intermedio: tablero de 16 filas por 16 columnas con un número total de 40 minas
 - Experto: tablero de 23 filas por 23 columnas con un número total de 99 minas.
 - Personalizado: en esta modalidad se permitirá generar una partida únicamente en el caso de que el valor de las filas y las columnas sea el mismo ya que el código proporcionado en el momento inicial del proyecto no se encontraba implementado.
- El nivel experto y personalizado tendrán como restricción básica que el número de filas y columnas sean el mismo. Por tanto, dentro del personalizado, el usuario deberá elegir la misma cantidad tanto en el ancho del tablero (columnas) como en el largo (filas).
- Las opciones que facilita el menú, en el caso del botón de *reiniciar*, siempre cargará el mismo nivel que se ha establecido en la pantalla de selección de nivel. El botón de *juego nuevo* da la opción de volver a elegir un nuevo nivel, es decir, volvería a la pantalla de inicio del juego. Por último, únicamente se podrán mostrar las mejores puntuaciones en el menú de inicio, nunca se podrán consultar mientras que se está jugando una partida.
- El nombre de usuario introducido en el caso de alcanzar un tiempo récord en la dificultad que hayamos elegido no deberá tener espacios ya que la búsqueda realizada por el programa usa los espacios como separador de búsqueda. Si el usuario metiese espacios en el nombre con el que guardase la partida, saldrá un pop up informando al usuario de la restricción correspondiente.
- Si el usuario no supera un récord de tiempo de ese nivel, no se le da la opción de guardar el nombre.
- Solo se podrá acceder a la tabla de los mejores tiempos de cada modalidad desde la pantalla de inicio del *Buscaminas* donde se elige la dificultad en la que se quiere jugar.



- Desde el menú de inicio sólo se podrá cargar una partida guarda anteriormente. Además, desde la pantalla de juego no se podrá ver los mejores tiempos de las dificultades pudiendo, guardar la partida en curso, cargar una partida guardada con anterioridad, reiniciar la partida actual y volver al inicio para cambiar la dificultad del juego.

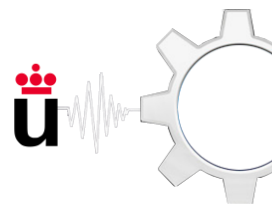
3.3 GESTIÓN DE RIESGOS

Para evitar la pérdida de información en el caso de que uno de los ordenadores personales con los que se está realizando el proyecto, se utilizará la plataforma de GitHub, a través de la cual todos los miembros del grupo podrán ver en cada momento quién ha realizado los cambios sobre los informes a realizar, así como la actualización del código del programa *Buscaminas*.

Sin embargo, exponerlo a una plataforma pública como es GitHub supondrá el riesgo de que otros grupos que realicen las mismas modificaciones sobre el código del juego se aprovechen de la información proporcionada en los informes.

Además, se guardará una copia de seguridad, después de realizar cada cambio en el ordenador de todos los miembros del grupo, copia que se enviará por correo electrónico a todos los miembros.

Por otro lado, como riesgos a los que se enfrentará el proyecto se puede destacar también la falta de experiencia con el software de GitHub, que supondrá un retraso en el desarrollo de las actividades del proyecto ya que se necesitarán unos días para adaptarse al entorno y comprender su funcionamiento completo.



3.4 MECANISMOS DE SUPERVISIÓN Y CONTROL

El proceso que se llevará a cabo para la supervisión y control de las modificaciones sobre el código y las actualizaciones de los informes pertinente tanto a través de la plataforma de GitHub como mediante reuniones grupales.

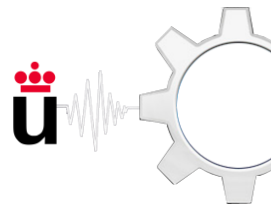
Además, para cada tarea elaborada, como por ejemplo, añadir los niveles de dificultad al juego o mostrar el tiempo, se ejecutará el código a través de unas pruebas para comprobar que todo funcionaba perfectamente. Estas pruebas hay que tener en cuenta que se realizarán siguiendo las restricciones descritas anteriormente, ya que sino, el juego no funcionará de forma adecuada.

Para comprobar si las restricciones se aplican correctamente, se introducirán nombres con espacios pudiendo así, ver si saldrá el pop up correspondiente indicando que el nombre introducido es incorrecto.

Además, en cada nivel de dificultad, después de realizar la tarea que facilitase la elección de dicho nivel, se comprobará que el tablero del juego salía tal y cómo indicaban las restricciones. Hay que añadir que, en el nivel de personalizado, se llevará a cabo un control más exhaustivo para comprobar que el tablero se cree con las medidas especificadas por el usuario.

Por otro lado, una vez añadida la tarea de guardar los 10 mejores tiempos, todos los miembros del grupo jugarán durante media tarde al juego, controlando así la inserción de las partidas dentro del fichero, con el correspondiente nombre de usuario y su tiempo. Gracias a este control, se podrá comprobar que, dentro de los ficheros, los tiempos se organizarán de forma ascendente en función del tiempo de la partida, desde el mejor tiempo hasta el peor.

El mismo control se llevará a cabo después de realizar la tarea de guardar las partidas en un fichero. Se comprobará que el fichero tiene la partida actual guardada correspondiente, así como, se tratará de volver a cargar la partida guardada con anterioridad para comprobar su correcto funcionamiento. Además,



se realizará una prueba en la que guardaremos una partida y probaremos a cargarla el día siguiente.

3.5 PLAN DE PERSONAL

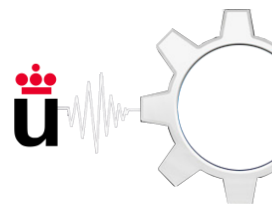
El equipo de trabajo, grupo E, se compondrá de cuatro integrantes, que son los siguientes:

- Jorge García Ranera
- Adina Georgiana Onofrei
- Álvaro Segura Manzanares
- Carlos Graña Muñoz

Cabe destacar que como ya se ha indicado con anterioridad, cada actividad principal que compone el proyecto tiene un Jefe de sección el cual se encargará de dar el visto bueno final cada vez que se realice una modificación sobre cualquier elemento que compone el proyecto. Sin embargo, todos los integrantes colaboran de forma equitativa, fomentando el trabajo en equipo y la lluvia de ideas.

Los cuatro integrantes del grupo, se repartirán el trabajo de tal forma que sus habilidades consigan adaptarse a las tareas del proyecto y conseguir así una mejor eficacia y ahorro de tiempo.

De forma conjunta, se ha estipulado que cada actividad, incluyendo sus tareas, tengan unas duraciones estimadas en función de los conocimientos concretos relacionados con la materia sobre la que se basa el proyecto.



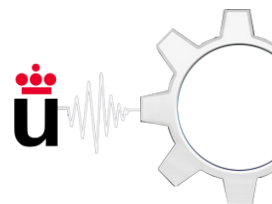
4. Proceso Técnico

4.1 METODOLOGÍA, TÉCNICAS Y HERRAMIENTAS

Para el correcto funcionamiento y posterior implementación del juego, teniendo en cuenta todas las pruebas que van a realizar, se analizará en detalle el código inicial proporcionado para destacar qué partes se van a modificar, en qué método van a ir implementadas las mejoras del juego y qué apariencia va a tener la nueva versión en cuanto a los niveles y opciones del juego.

El ejecutable del juego será desarrollado de tal forma que podrá adaptarse a cualquier sistema operativo. Además de tener todas las librerías pertinentes para poder ejecutarse en sistemas operativos “vanillas”, los cuales, no tienen instalados los elementos que posibilitan ejecutar archivos codificados en el lenguaje de programación solicitado por el cliente. Hay que destacar que el juego de *Buscaminas* será implementado en el lenguaje de programación Java, utilizando ficheros (*incluidos los Buffers necesarios para su futura lectura y escritura*) y JFrames (*sobretudo JBottons y un JMenu para las opciones del juego*) para conseguir implementar de la mejor manera posible todas las mejoras sobre el juego. Todas las modificaciones se llevarán a cabo en el entorno de programación *NetBeans* mediante el cual transformaremos el código proporcionado por el cliente además de poder realizar distintas pruebas prácticas para comprobar el correcto funcionamiento de las mejoras diseñadas e implementadas por el equipo.

Se utilizará la plataforma GitHub con la que conseguiremos llevar un control exhausto de las modificaciones sobre los distintos informes y trabajos que estén llevando a cabo cada uno de los integrantes del equipo. También permitirá poder comprobar y verificar las correctas modificaciones llevadas a cabo en cada una de las tareas del proyecto.



4.2 DOCUMENTACIÓN SOFTWARE

Para la realización de los distintos apartados de este documento se han utilizado los siguientes programas:

- Diagrama de Gantt: se utilizará el programa de Microsoft Project 2010
- Tabla de Precedencias: se basará en el uso de Microsoft Excel 2016
- Plan de proyecto: se usará el Microsoft Word 2016
- Plan de mantenimiento: se usará el Microsoft Word 2016
- Plan de Gestión de Configuración: se utilizará el Microsoft Word 2016
- Diagrama PERT: se utilizará el Microsoft Word 2016 para las figuras y el Paint para hacer los tiempos early y late.
- Buscaminas: se programará en NetBeans usando el lenguaje Java

Además, se presentará también un Plan de Gestión de Configuración que sigue el estándar IEEE 282-2005 en el que se especifican todas las actividades de Gestión de Configuración y Cambios que serán realizados durante todo el ciclo de vida del proyecto.

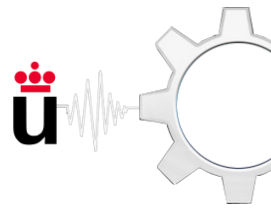
4.3 FUNCIONES DE APOYO AL PROYECTO

El proceso de desarrollo de las mejoras indicadas por el cliente en el juego *Buscaminas* requerirá de la presencia de una serie de funciones de apoyo junto con las cuales se simplificará y organizará el proceso. Estas funciones se dividirán en los siguientes grupos:

- Gestión de la configuración: se compondrá de un conjunto de informes, los cuales, estructurarán el proceso de desarrollo, tanto antes de la implementación de las mejoras como después de su entrega. Se dividirán los siguientes planes en función del momento de uso:



- Plan de Proyecto: este documento describirá en detalle la secuencia de actividades y tareas a seguir durante todo el desarrollo de proyecto. Servirá como punto de partida para el comienzo del proyecto, así como para la organización temporal y la secuencia de pasos que se debe seguir para conseguir la entrega del proyecto en el tiempo estipulado.
- Plan de Gestión de Configuración del Software: este plan hará referencia a la gestión de los cambios a lo largo del ciclo de vida del proyecto, es decir al arte de identificar, organizar, revisar y controlar las modificaciones que sufre nuestro software. Este plan de gestión se aplicará durante todas las fases del proceso con el objetivo de maximizar la eficiencia del *Buscaminas* minimizando sus errores.
- Aseguramiento de la calidad software: se llevarán a cabo una serie de controles de calidad durante todo el ciclo de vida del proyecto. Se comprobará y validará el código inicial proporcionado por el cliente y se comprobará la existencia de algún tipo de error mediante una serie de reuniones grupales. Para que cualquier verificación o control de calidad que se lleve a cabo, deberá ser validado por el jefe de sección. A lo largo de toda la actividad de mejora de las funcionalidades del juego *Buscaminas*, se realizarán una serie de comprobaciones periódicas por parte de todo el equipo. Estas comprobaciones se llevarán a cabo en las reuniones conjuntas comprobando que las mejoras indicadas por el cliente han sido integradas de forma satisfactoria en el juego mejorado. Por último, una vez entregado el proyecto final al cliente se llevará a cabo una serie de procesos de mantenimiento para solventar posibles errores y se modificarán todas aquellas funcionalidades que el cliente quiera cambiar una vez entregado el proyecto. Todos estos controles estarán apoyados por la plataforma GitHub con la que se irán actualizando los elementos al proyecto final.



5. Plan de desarrollo

5.1 PAQUETES DE TRABAJO

Como se estipuló en la primera reunión de equipo, el proyecto en cuestión se llevará a cabo dividiéndolo en una serie de actividades que a su vez, estarán divididas en una serie de tareas las cuales siguen un orden continuo y concreto. Además, varias de las tareas identificadas poseen una relación de dependencia, esto implica que, antes de poder realizar la tarea en cuestión se debe haber completado una anteriormente.

Por otro lado, la descomposición del trabajo servirá para poder representar las actividades que hay que realizar a distinto nivel con el objetivo de identificar la independencia entre las actividades principales del proyecto. De esta forma se usa el esquema siguiente que permite observar los paquetes de trabajo y aporta más claridad de cara a la organización del equipo.

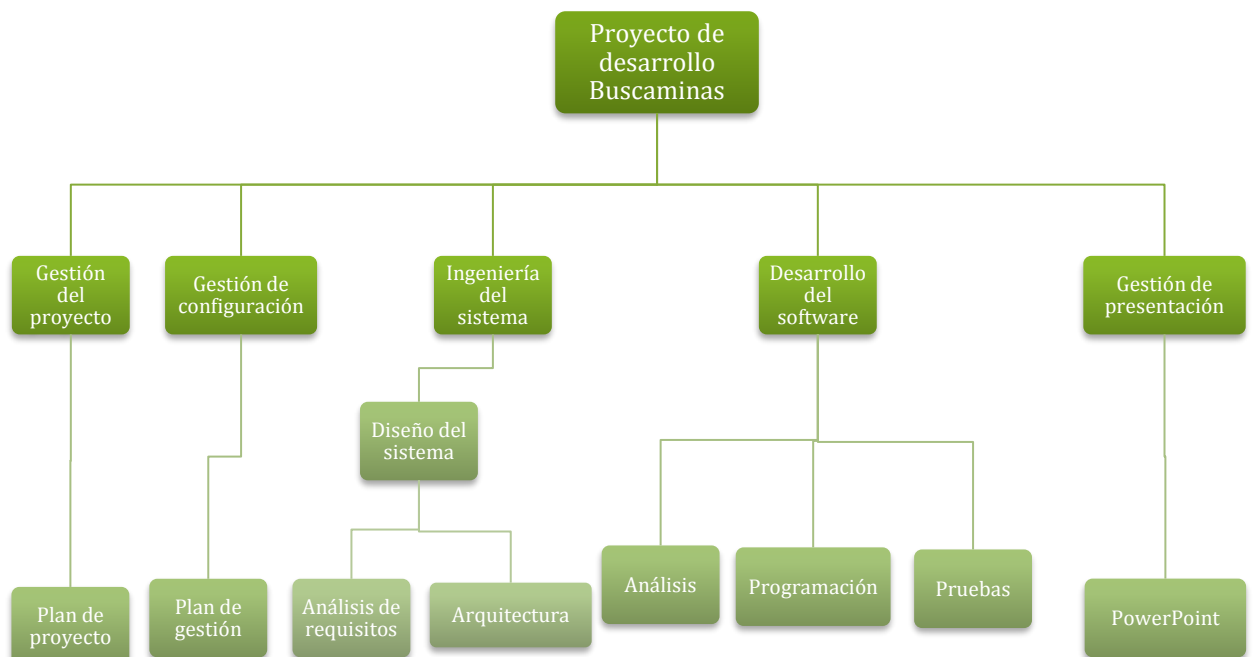


FIGURA 7 - ESQUEMA PAQUETES DE TRABAJO

5.2 DEPENDENCIAS

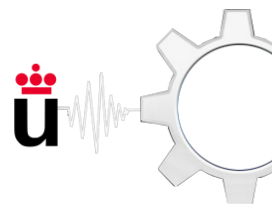
En este proyecto, el equipo ha identificado como actividad principal y primordial a realizar la mejora de las funcionalidades del juego buscaminas con los planes e informes pertinentes para su implementación y su puesta en marcha. De esta forma, se elaborará una matriz de encadenamientos, cuyas dimensiones coinciden con el número de tareas en las que se descompone la actividad principal del proyecto. Esta matriz servirá para reflejar las relaciones entre las diferentes tareas, consiguiendo así una mejor aclaración de cara a la implementación.

Esta matriz permitirá tener una mayor información acerca de los pasos que se deben dar para alcanzar las fechas de entrega solicitadas por el cliente.

Las actividades mostradas en la siguiente tabla serán denominadas con los mismos números que el diagrama de Gantt expuesto en puntos anteriores:

		Actividades Precedentes																
	2.1	2.2.1	2.2.2	2.2.3	2.2.4	2.2.5	2.2.6	2.2.7	2.2.8	2.2.9	2.2.10	2.2.11	2.2.14	2.9	2.16.9	2.16.10	2.23	2.25
2.1																		X
2.2.1														X				
2.2.2		X																
2.2.3			X															
2.2.4	X																	
2.2.5								X										
2.2.6												X						
2.2.7				X														
2.2.8						X	X						X					
2.2.9									X									
2.2.10									X									
2.2.11					X													
2.2.14								X										
2.9	X																	
2.16.9										X	X							
2.16.10															X			
2.23																X		
2.25																		

FIGURA 8 - TABLA DE DEPENDENCIAS



Al haberse identificado cuatro actividades principales dentro de este proyecto, todas ellas tendrán una serie de dependencias unas con otras. Estas actividades ya mencionadas son:

- Plan de Proyecto
- Plan de Configuración de Software
- Mejoras en la funcionalidad del juego *Buscaminas*
- Plan de Mantenimiento

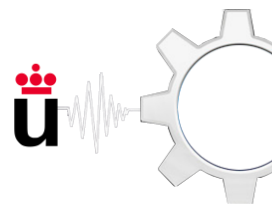
Las dependencias entre las tareas en las que se han dividido la actividad de mejora del juego se han comentado en el subapartado de Organización de Proyecto (2.1), denominado Modelo de Procesos.

Por otra parte, las otras tres actividades dependen unas de otras; la única actividad que no dependerá de las otras identificadas es el Plan de Proyecto ya que es el documento inicial sobre el que nos basaremos para el desarrollo de los requisitos estipulados por el cliente.

De este modo, las dependencias mencionadas se pueden resumir en la siguiente tabla:

Actividad	Actividad de la que depende
Plan de Proyecto	-
Plan de Configuración de Software	Mejora del Juego
Mejora del Juego buscaminas	Plan de Proyecto Plan de Configuración de Software
Presentación PowerPoint	Mejora del Juego, plan de Proyecto y plan de Configuración de Software

FIGURA 9 - TABLA DE ACTIVIDADES PRINCIPALES



Por otro lado, al ser un proyecto que se adapta a cualquier sistema operativo (Windows, MacOS, Linux...) se añadirá al archivo ejecutable del juego el ejecutable del JDK distribuido por parte de la empresa ORACLE ya que el juego se basa en el lenguaje de programación java y requiere de una serie de librerías para que se pueda ejecutar.

Una vez determinadas tanto las dependencias globales como las dependencias de las actividades, se realizará la construcción del diagrama de PERT, según el cual, nos permitirá determinar la duración total del proyecto y mostrar la secuencia más larga de las actividades conectadas a través de una red. Una vez realizado dicho diagrama podremos ver cuál es el camino crítico del proyecto. Hay que destacar que el diagrama de PERT se ha explicado previamente en el punto 2.1 de este mismo plan, el cual es el siguiente:

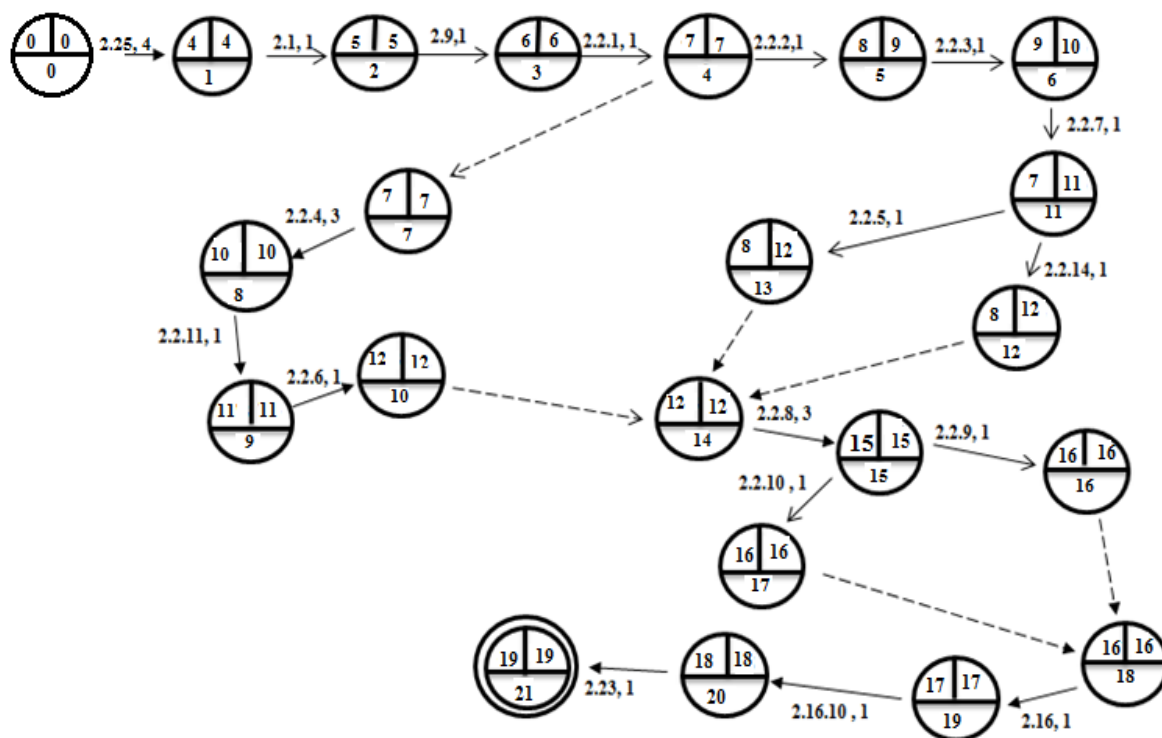
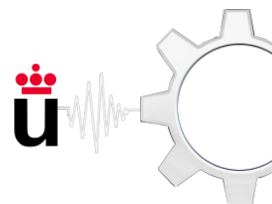


FIGURA 10 - DIAGRAMA DE PERT



Se ha decidido usar el diagrama de PERT para este apartado ya que el proyecto presenta todas las actividades bien definidas, se pueden relacionar unas con otras, están ordenadas de tal forma que simulan una secuencia y además una vez comenzada una actividad, se tiene que continuar hasta su finalización.

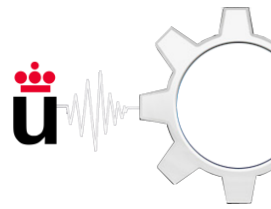
5.3 RECURSOS

Este proyecto de desarrollo del juego Buscaminas se realizará por cuatro alumnos del doble grado de Ingeniería Informática y Administración y Dirección de Empresas de la Universidad Rey Juan Carlos del campus de Vicálvaro.

Este proceso se hará de manera coordinada tanto en reuniones en persona como puestas en común en diversas las plataformas y aplicaciones que a continuación detallaremos, además de en salas de la Biblioteca del campus de Vicálvaro habilitadas para los trabajos grupales. Para el trabajo disponemos de tres portátiles personales y de un ordenador fijo entre los cuatro integrantes del equipo, además de los ordenadores del aula de informática en las sesiones que el profesor habilitará para avanzar el proyecto. Estos ordenadores tendrán unos procesadores i7 o i5 y dispondrán de una memoria RAM entre 8 y 16 GB. En el caso del ordenador de mesa será un procesador i7 con 32 GB de RAM. Para su realización el equipo deberá repasar el lenguaje de programación java y en especial el manejo de ficheros en dicho lenguaje. También, todo el equipo deberá leerse las diapositivas facilitadas por el profesor a través del Aula Virtual en el apartado de su asignatura.

A continuación, se expondrán todas las herramientas software que el equipo utilizará:

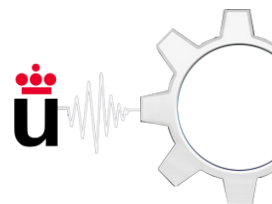
- GitHub: es un repositorio que nos permitirá trabajar de manera remota y poder seguir de primera mano los avances del resto del equipo.



- Netbeans: es un entorno de programación donde se desarrollará todo el proceso software hasta llegar al producto final.
- Microsoft Word: es un entorno de escritura que nos permitirá realizar la documentación del proyecto.
- Microsoft Excel: es un entorno organizado en celdas que nos servirá para realizar el calendario con las tareas a realizar.
- Microsoft Project 2010: entorno que permitirá realizar el diagrama de Gantt de una manera sencilla y muy visual.
- Paint: nos permitirá realizar diferentes figuras que nos harán mas sencillo el diseño del diagrama Pert.
- Skype: es una aplicación que nos permitirá hacer videoconferencias de manera online para aclararnos diferentes dudas que puedan surgir y sea imposible resolver en persona.
- WhatsApp: aplicación de mensajería instantánea que nos permitirá tener un grupo con todos los miembros del equipo para ir informando sobre los avances realizados.
- Aula Virtual: plataforma de la universidad donde tenemos información sobre los procesos Software además de las especificaciones del cliente en un pdf.
- Eclipse: entorno de programación parecido a Netbeans pero consideramos que el manejo de las frame es más sencillo y realizaremos esa parte del proyecto y después copiaremos a Netbeans donde se hará casi todo el desarrollo.

5.4 PRESUPUESTO Y DISTRIBUCIÓN DE RECURSOS

La realización de este proyecto no conllevará ningún coste ya que todo se hace a través de las herramientas ya disponibles. La utilización de recursos ha sido principalmente en tiempo y personal.



5.5 CALENDARIO

Este proyecto empezará el día 23 de marzo de 2018 y acabará el día 25 de abril de 2018. Para cumplir con el plazo establecido, se ha realizado un calendario con las tareas a desarrollar, aunque la mayor parte de éstas están relacionadas con una de las principales actividades del proyecto (*desarrollo de una versión actualizada del juego*) con el objetivo de mejorar la organización de los miembros del grupo y poder cumplir con los requisitos del cliente, también cabe destacar que la primera tarea del calendario, comprendida entre los días 23 de Marzo y 27 de Marzo, es la creación de los documentos necesarios para el proyecto, es decir el Plan de Proyecto y el Plan de Gestión de la Configuración Software.

Nombre	Duración	Comienzo	Fin
Crear los documentos necesarios	4 días	3/23/18 9:00 AM	3/27/18 1:00 PM
Verificación de código	1 día	3/27/18 3:00 PM	3/28/18 10:00 AM
Verificación de ausencias	1 día	3/27/18 3:00 PM	3/28/18 10:00 AM
Validación de código	1 día	3/28/18 10:00 AM	3/28/18 5:00 PM
Validación de código	1 día	3/28/18 10:00 AM	3/28/18 5:00 PM
Versión actualizada	11 días	3/28/18 5:00 PM	4/6/18 4:00 PM
Añadir opción para reiniciar el juego	1 día	3/28/18 5:00 PM	3/29/18 12:00 PM
Mostrar cantidad de minas	1 día	3/29/18 12:00 PM	3/29/18 7:00 PM
Mostrar tiempo de partida	1 día	3/29/18 12:00 PM	3/29/18 7:00 PM
Niveles de dificultad	3.2 días	3/29/18 12:00 PM	4/2/18 12:00 PM
Guardar 10 mejores tiempos de cada dif	1 día	4/2/18 12:00 PM	4/2/18 7:00 PM
Mostrar 10 mejores tiempos en cada dif	1 día	4/3/18 9:00 AM	4/3/18 4:00 PM
Añadir un nombre de jugador	1 día	3/30/18 9:00 AM	3/30/18 4:00 PM
Dar la opción de guardar el tiempo o no	1 día	4/2/18 9:00 AM	4/2/18 4:00 PM
Permitir guardar la partida en un fichero	3.2 días	4/2/18 9:00 AM	4/4/18 2:00 AM
Permitir consultar la última partida juga	1 día	4/4/18 9:00 AM	4/4/18 4:00 PM
Recuperación de partida guardada de un	1 día	4/5/18 9:00 AM	4/5/18 4:00 PM
Menú para las diferentes opciones del j	1 día	4/6/18 9:00 AM	4/6/18 4:00 PM
Validación de código actualizado	1 día	4/6/18 4:00 PM	4/9/18 11:00 AM
Validación del código actualizado	1 día	4/6/18 4:00 PM	4/9/18 11:00 AM
Pruebas de código	1 día	4/6/18 4:00 PM	4/9/18 11:00 AM
Puesta en común del proyecto	1 día	4/9/18 11:00 AM	4/9/18 6:00 PM

FIGURA 11 - CALENDARIO DE GANTT

Los días restantes se dedicarán preparar la presentación PowerPoint para la exposición, así como de corregir aquellos errores que surjan de forma imprevista o modificar algunos pequeños detalles relacionados con la estética de ambos documentos.