**Proyecto**

“ALPHA-SCHOOL”

**Grupo**

“Cerberus”

"DOCUMENTO DE PLANIFICACIÓN"

Hito: 1

Fecha entrega: 30-11-2016

Versión: 2

Componentes:

* Jorge Cabanes
* Manuel Romero
* Nahiara Latorre
* Antonio Martínez
* Ricardo Espí

Contenido

[Contenido 1](#_Toc466288125)

[1. Propósito 2](#_Toc466288126)

[2. Análisis de riesgos. 2](#_Toc466288127)

[2.1. Identificación de riesgos. 2](#_Toc466288128)

[2.2. Prioridades de los riesgos. 2](#_Toc466288129)

[2.3. Planes de contingencia. 2](#_Toc466288130)

[3. Estimación de costes. 2](#_Toc466288131)

[3.1. Ley de Parkinson. 2](#_Toc466288132)

[3.2. Pricing to Win. 2](#_Toc466288133)

[3.3. Puntos Objeto y/o Puntos de Función. 2](#_Toc466288134)

[3.4. Comparación y discusión de los valores obtenidos. 2](#_Toc466288135)

[4. Agenda del proyecto. 2](#_Toc466288136)

[4.1. Relación de actividades ( WBS). 2](#_Toc466288137)

[4.2. Plan general del proyecto (Hitos). 2](#_Toc466288138)

[4.3. Plan detallado del proyecto (2 iteraciones). 2](#_Toc466288139)

[4.4. Asignación de tiempo y recursos a actividades. 3](#_Toc466288140)

[4.4.1. Recursos humanos. 3](#_Toc466288141)

[4.4.2. Otros recursos. 3](#_Toc466288142)

[5. Mecanismos de seguimiento y control. 3](#_Toc466288143)

# Propósito

El propósito del documento de planificación software es reunir toda la información necesaria para controlar el proyecto. Describe la forma en la que realizaremos el desarrollo de la aplicación web y es el documento que especifica el *planning* a más alto nivel que emplearemos para dirigir el desarrollo. Los detalles de cada iteración serán descritos más en profundidad en los planes de iteración.

# Análisis de riesgos.

# Identificación de riesgos.

Para identificar los riesgos a los que nuestro proyecto puede enfrentarse durante su fase de desarrollo, encontramos seis posibles campos en los que un riesgo puede producirse:

* **Riesgos de elaboración en la planeación:** la planificación no incluye las tareas necesarias o que las áreas desconocidas del producto lleven más tiempo del esperado.
* **Tamaño del producto:** que el tamaño del producto sea mayor del planteado.
* **Riesgo tecnológico:** Este es el tipo de riesgo más improbable ante el que nos podemos encontrar. Se refiere a posibles errores por parte del software o hardware que empleamos. Podría ser, por ejemplo, que el ordenador personal de uno o varios de los integrantes del equipo fallara. Otro ejemplo sería que la BBDD seleccionada no soporta el número de transacciones requeridas.
* **Riesgo de personal:** en este tipo de riesgo hay que tener en cuenta 3 cosas:
  + Inexperiencia del personal: es muy posible que nos encontremos ante situaciones para las que no tenemos un entrenamiento funcional, y tengamos que apoyarnos únicamente sobre conocimientos teóricos.
  + Enfermedad: que alguien del personal se enferme, lo cual significaría que el resto de personal tuviera que hacer su trabajo, retrasando de esta forma el proyecto.
  + Falta de moral y motivación en el equipo de trabajo.
* **Riesgo organizacional:** En el caso de que nuestra identificación de riesgos sea errónea y se presente una casuística inesperada para la que no tenemos un plan de contingencia nos encontraríamos ante un riesgo de este tipo.
* **Riesgos de herramientas:** Debido al amplio abanico de tecnologías que vamos a emplear nos podríamos encontrar ante problemas de integración entre ellas.
* **Riesgos de requerimientos:** En el caso de que hayamos fallado en la fase de análisis del proyecto, nos encontraríamos ante una situación en la que las tareas que debíamos desarrollar cambiarían. Que los requisitos no se hayan sido definidos correctamente.
* **Riesgos de estimación:** A la hora de realizar estimaciones sobre el tiempo que requiere realizar una tarea, la experiencia a la hora de realizar tareas similares nos da un *feedback* sobre el coste de las mismas. Otra vez, debido a nuestra inexperiencia, existe la posibilidad de que el cálculo de costes sea erróneo.
* **Riesgos legales:** retraso que sufriría la realización del proyecto en caso de que una operación tenga algún error de interpretación jurídica o alguna omisión en la documentación.
* **Riesgo de desempeño**: Grado de incertidumbre de que el producto satisfaga los requisitos y se ajuste al uso que se pretende darle.
* **Riesgos impredecibles**: algunos problemas que pueden ocurrir sin advertir.

# Prioridades de los riesgos.

|  |  |  |
| --- | --- | --- |
| POSIBLE RIESGO | PROBABILIDAD | EFECTOS |
| Error de integración entre frameworks | BAJA | CATASTRÓFICO |
| Errores inesperados | ALTA | SERIO |
| Error a la hora de realizar las estimaciones de tiempo | MEDIA | SERIO |
| Fallo del equipo personal de uno de los miembros | BAJA | SERIO |
| Problemas legales | BAJA | SERIO |
| Más número de usuarios | ALTA | CATASTRÓFICO |
| Personal inexperto | ALTA | TOLERABLE |
| Tamaño estimado demasiado pequeño | ALTA | TOLERABLE |
| Personal enfermo | MEDIA | TOLERABLE |
| Falta de moral y motivación del personal | MEDIA | TOLERABLE |
| Obtención de requisitos y análisis erróneos | MEDIA | TOLERABLE |
| Planificación errónea | MEDIA | TOLERABLE |
| Producto final diferente del esperado | MEDIA | TOLERABLE |

# Planes de contingencia.

|  |  |
| --- | --- |
| RIESGO | ESTRATEGIA |
| Fallo del equipo personal de uno de los miembros | Crear una máquina virtual de backup que se pueda iniciar rápidamente en uno de los ordenadores de la facultad |
| Personal inexperto | Entrenamiento con software nuevo a emplear |
| Planificación errónea | Contar con una tabla de planificación por ciclos en la que veamos las probabilidades de terminar una funcionalidad en un tiempo determinado |
| Error de integración entre frameworks | Contar con un framework más sencillo y con menos funcionalidades, pero más adaptable, de reserva |
| Obtención de requisitos y análisis erróneos | Emplear una metodología de desarrollo software *agile*, que permite en iteraciones cortas tener una funcionalidad acabada. En el caso de que el análisis haya sido erróneo, no habremos perdido mucho tiempo. |
| Error a la hora de realizar las estimaciones de tiempo | Dejar unos márgenes de tiempo razonables para posibles eventualidades |

# Estimación de costes.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Empleado** | **Planificación del proyecto** | **Implementación del motor gráfico** | **Edición de contenido audiovisual** | **Programación backend** | **Difusión de contenidos** | **Programación frontend** | **Negocio** | **Diseño de tutoriales** | **Horas totales** |
| Cabanes | 0 | 0 | 0 | 0 | 112.5 | 0 | 150 | 0 | 272.5 |
| Espí | 24 | 94.75 | 90 | 0 | 112.5 | 150 | 150 | 150 | 771,25 |
| Latorre | 24 | 94.75 | 0 | 150 | 112.5 | 150 | 150 | 0 | 681,25 |
| Martínez | 24 | 94.75 | 90 | 150 | 0 | 150 | 150 | 0 | 658,75 |
| Romero | 24 | 94.75 | 90 | 0 | 112.5 | 150 | 150 | 0 | 621,25 |

Total horas: 3005

Número de personas a contratar: 5

Sueldo: 25€/h \*

|  |  |
| --- | --- |
| **Empleado** | **Sueldo** |
| Cabanes | 272.5\*25= 6812.5€ |
| Espí | 771.25\*25= 19281.25€ |
| Latorre | 681.25\*25= 16281.25€ |
| Martínez | 658.75\*25= 16468.75€ |
| Romero | 621.25\*25= 15531.25€ |

\*Esta cantidad está calculada conforme al presupuesto disponible en una gran empresa. El desarrollo de nuestro proyecto no está remunerado.

# 

# Ley de Parkinson.

El coste del proyecto estará en función de los recursos de los que disponemos, utilizando software gratuito en la medida de lo posible. El tiempo estimado para acabar el proyecto está dividido en hitos, y estos a su vez en iteraciones.

# Pricing to Win.

El pago del servicio se dividirá en un primer pago y cuota anual. El coste se ajustará al presupuesto medio del que dispone un centro educativo, teniendo en cuenta el número de usuarios del centro.

# Puntos Objeto y/o Puntos de Función.

Puntos de Función

|  |  |  |  |
| --- | --- | --- | --- |
| **Parámetro de medida** | **Contador** | **Factor de peso** | **Resultado** |
| N° entradas usuario | 24 | x3 | 122 |
| 2 | x4 |
| 7 | x6 |
| N° salidas usuario | 15 | x4 | 170 |
| 15 | x5 |
| 5 | x7 |
| N° peticiones usuario | 13 | x3 | 301 |
| 58 | x4 |
| 5 | x6 |
| N° ficheros | 37 | x3 | 115 |
| 1 | x4 |
| 0 | x6 |
| Nº Interfaces externas |  | x7 | 30 |
|  | x10 |
| 2 | x15 |
| Total contadores |  | | 184 |
| Multiplicador de complejidad | C=0.65+0.01\*N  N=2 | | 0.67 |
| **Puntos de función** | (Suma resultados)\*C  922\*0.67 | | **617.74** |

# Comparación y discusión de los valores obtenidos.

Utilizando la herramienta de <http://www.cuantocuestamiweb.com/> hemos comprobado que una página web de estas características costaría una cantidad aproximada a 4336€, y 10000€ el desarrollo de la app según <https://www.cuantocuestamiapp.com/> . Puesto que no trabajamos para nadie en concreto, el desarrollo de la aplicación lo haremos sin cobrar, y serán los clientes los que pagarán a posteriori para obtenerla. El precio para cada cliente variará conforme a lo estipulado en el punto [Pricing to Win.](#_Pricing_to_Win.)

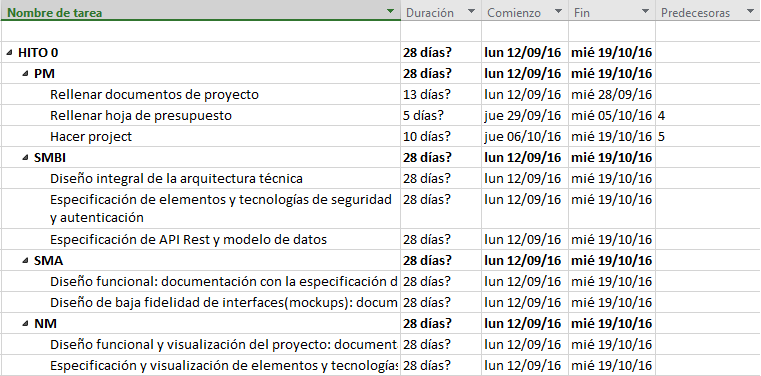
# Agenda del proyecto.

# Relación de actividades (WBS).

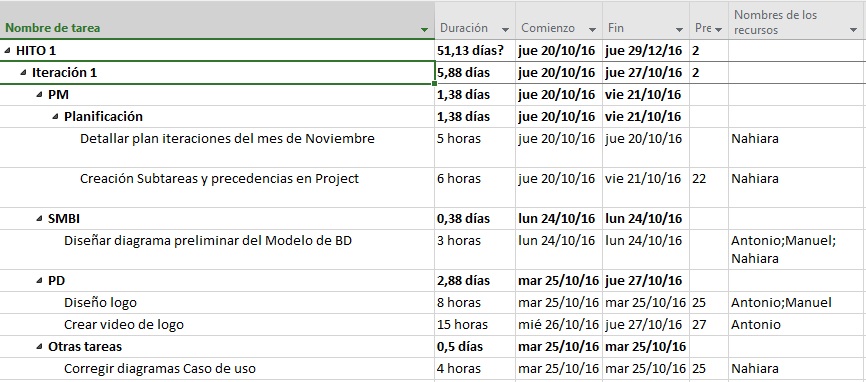
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PM** | **SMBI** | **SMA** | **NM** | **PD** | **TAG** | **E** |
| Rellenar documentos de proyecto | Diseño integral de la arquitectura técnica | Diseño funcional: documentación con la especificación de todas las funcionalidades del sistema | Diseño funcional y visualización del proyecto: documentación con la especificación de todas las funcionalidades del sistema | Diseño logo | Visualizador OpenGL 4.X simple, con datos por programa. Shader básico. | MOOC Lean Canvas |
| Rellenar hoja de presupuesto | Especificación de elementos y tecnologías de seguridad y autenticación | Diseño de baja fidelidad de interfaces(mockups): documentación con todas las interfaces diseñadas mediante mockups | Especificación y visualización de elementos y tecnologías a utilizar, por ejemplo mediante la definición de una infografía del proyecto | Crear video de logo | Parser de objetos en múltiples formatos, salida en modo texto | Guion |
| Hacer project | Especificación de API Rest y modelo de datos | Diagrama de flujo de interfaces | Definir métricas | Cartel del proyecto | Visualizador OpenGL 4.X simple, con datos por programa. Shader básico. | Guion gráfico |
| Ordenar tareas por iteraciones | Diseñar diagrama preliminar del Modelo de BD | Hacer documentación con el mapa de navegación entre interfaces | Especificar elementos y tecnologías a utilizar | Tráiler del proyecto | Tipos de datos para nodos (completo) y entidades (sólo clase padre; clases hijas sin contenido) | Elegir tecnología |
| Registrar de tiempos y % de realización de tareas en Project | Configuración de los elementos de backend y seguridad | Integración Front-Back: documento de relación entre back-end y front-end | Definir indicadores principales del proyecto | Making of del proyecto | Construcción del árbol | Hacer vídeo |
| Comparar la planificación prevista y real en Project hito 1. | Instalación de los elementos de backend y seguridad | Guía de estilos: documento detallando la imagen corporativa del servicio (colores, estilos, esencia para la experiencia de usuario, consideraciones de usabilidad y accesibilidad) | Contextualizar modelo de datos | Créditos | Recorrido del arbol | Grabar audio |
| Confeccionar Informes de iteración e informe resumen de Hito 1 | Implementación API RESTFULL | Documento de SEO: definición de aspectos SEO como son palabras clave, estilos de textos, posiciones de texto, como se va a potenciar esto (landing, blog, etc…), URL amigables | Incorporación de open data, API's, y otras fuentes heterogéneas |  | Salida en modo texto | Montaje |
| Aplicar el modelo EVA en Project | Hacer documentación | Arquitectura tecnológica: documento sobre toma de decisión, versiones, codificación, internacionalización (varias lenguas), instalación sobre servidor de frameworks... | Modelo para la integración de datos |  | Tipos de datos para entidades de tipo transformación |  |
| Elaborar la presentación del Hito 1 | Pruebas del API RestFull | Implementación front-end más documentación | Despliegue de la infraestructura del proyecto |  | Tipos de datos para entidades de tipo cámara y luz (cámaras y luces de dos tipos). Salida en modo texto |  |
| Revisar especificación proyecto. | Pruebas de seguridad | Implementación de las funcionalidades en front-end (relación de las interfaces y funcionalidades que se están implementando y en que momento están implementadas) | Definición de cuadros de mando e implementación |  | Tipos de datos para entidades de tipo malla. Salida en modo texto |  |
| Reestimar proyecto | Pruebas unitarias | Documento de pruebas donde se documenta las pruebas de las funcionalidades principales (por lo menos las interfaces principales) | Definición de KPI's implementación |  | Tipos de datos para entidades de tipo animación (cuadro a cuadro). Salida en modo texto |  |
| Detallar plan iteraciones del mes de Enero | Pruebas de integración | Pruebas de la aplicación demostrando que no hay errores (falta de librerías, archivos que no se descargan, errores javascript) | Valoración de los cuadros de mando y API's definidos e incorporación de nuevos elementos según datos |  | Carga de modelos en formato múltiple en las entidades de tipo malla |  |
| Comparar la planificación prevista y real en Project hito 2 | Pruebas funcionales | Pruebas de seguridad de front end | Validación del funcionamiento |  | Carga de varios ficheros para animación. Salida en modo texto |  |
| Confeccionar Informes de iteración e informe resumen de Hito 2 | Pruebas de regresión | Pruebas de rendimiento (velocidad de carga, que todo carga y no hay errores) |  |  | Visualizador OpenGL 4.X de las entidades tipo malla. Shader básico. Integración con el motor. Sin materiales, texturas, cámara ni luces |  |
| Detallar plan iteraciones del mes de Febrero-Marzo | Pruebas de carga y rendimiento. | Validar las funcionalidades (comprobar que realmente han sido implementadas), las funcionalidades han quedado recogidas en las interfaces implementadas |  |  | Visualizador OpenGL 4.X de las entidades tipo malla. Shader básico. Con cámaras y luces de varios tipos. Visualización con registro de cámaras y luces. |  |
| Comparar la planificación prevista y real en Project hito 3 | Validación del funcionamiento del backend |  |  |  | Carga de materiales y texturas. Salida en modo texto |  |
| Confeccionar Informes de iteración e informe resumen de Hito 3 |  |  |  |  | Visualizador OpenGL 4.X de las entidades tipo malla. Shader básico. Añadiendo materiales y texturas. |  |
| Elaborar la presentación del Hito 3 |  |  |  |  | Creación de una fachada entre el motor y la aplicación |  |
| Exponer la presentación del Hito 3 |  |  |  |  | Visualizador OpenGL 4.X con tres shaders avanzados |  |
| Detallar plan iteraciones del mes de Abril-Mayo |  |  |  |  | Cargador de modelos |  |
| Comparar la planificación prevista y real en Project hito 4 |  |  |  |  | Movimiento del modelo, de las luces y de la cámara |  |
| Confeccionar Informes de iteración e informe resumen de Hito 4 |  |  |  |  | Cambio de modelo, materiales y texturas |  |
| Elaborar la presentación del Hito 4 |  |  |  |  | Animaciones |  |
| Exponer la presentación del Hito 4 |  |  |  |  | Integración con el motor |  |
|  |  |  |  |  | Optimizaciones del motor |  |
|  |  |  |  |  | Efectos visuales |  |

# Plan general del proyecto (Hitos).

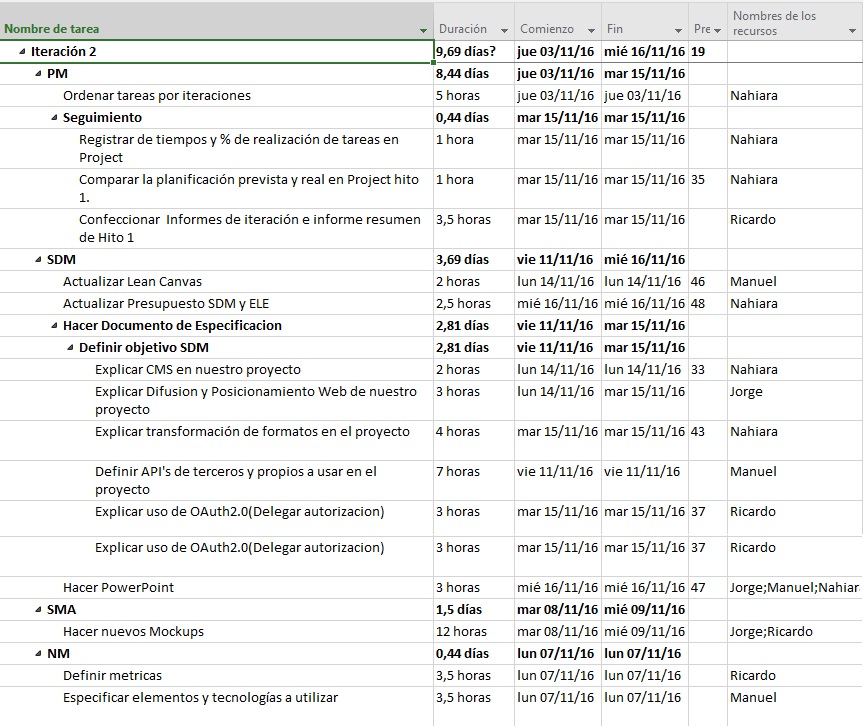
**Hito 0**

****

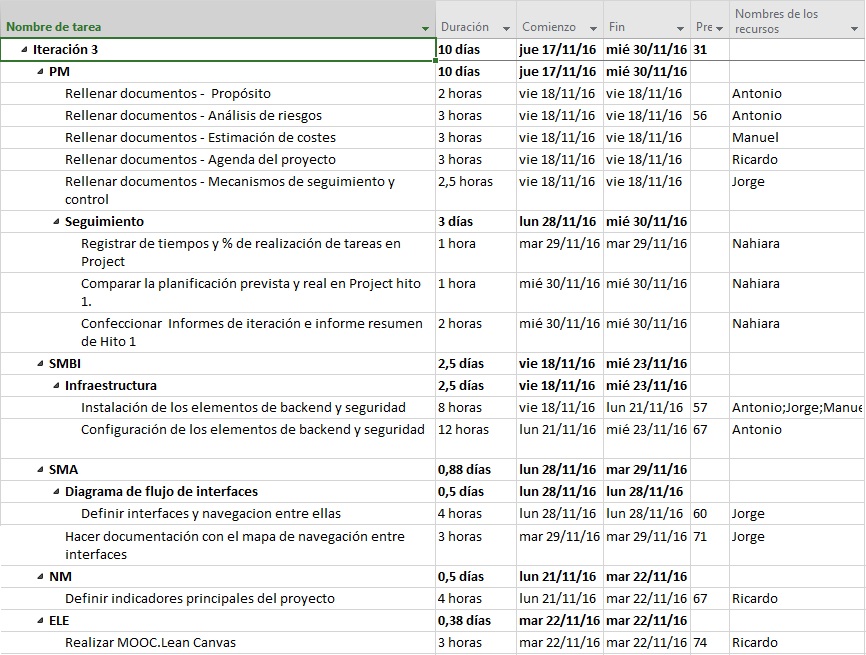
**Hito 1**

Iteración 1

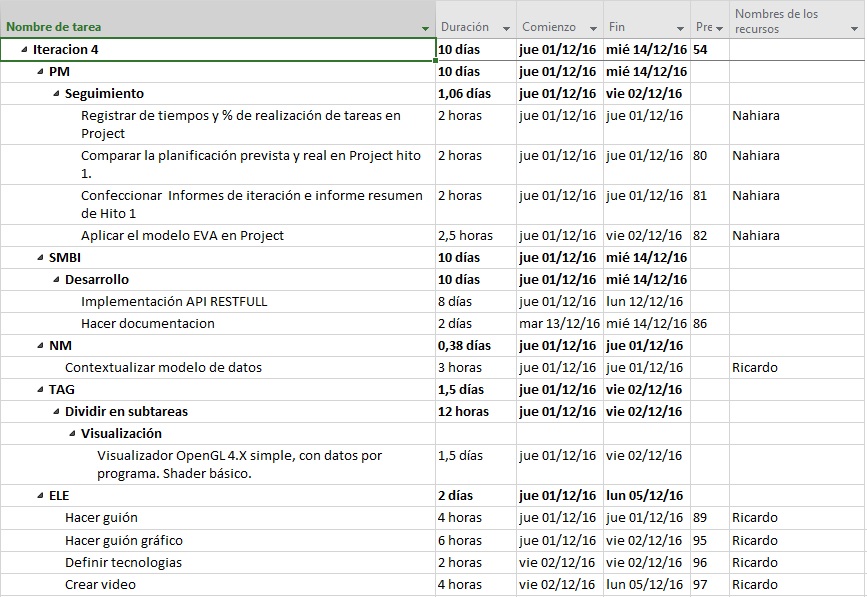
Iteración 2

****

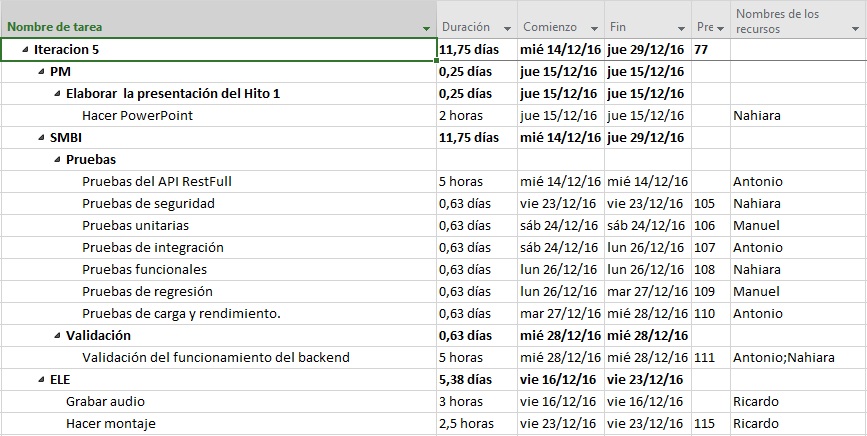
Iteración 3

****

Iteración 4

****

Iteración 5

****

**Hito 2**

**Hito 3**

**Hito 4**

# Plan detallado del proyecto (2 iteraciones).

**Iteración 3**

Jorge

* PM - Rellenar documentos - Mecanismos de seguimiento y control
* SMA - Definir interfaces y navegación entre ellas
* SMA - Hacer documentación con el mapa de navegación entre interfaces

Ricardo

* PM - Rellenar documentos - Agenda del proyecto
* NM - Definir indicadores principales del proyecto
* ELE - Realizar MOOC y Lean Canvas

Nahiara

* PM - Registrar de tiempos y % de realización de tareas en Project
* PM - Comparar la planificación prevista y real en Project hito 1
* PM - Confeccionar Informes de iteración e informe resumen de Hito 1
* SMBI - Instalación de los elementos de backend y seguridad

Antonio

* PM - Rellenar documentos - Propósito
* PM - Rellenar documentos - Análisis de riesgos
* SMBI - Instalación de los elementos de backend y seguridad
* SMBI - Configuración de los elementos de backend y seguridad

Manuel

* PM - Rellenar documentos - Estimación de costes
* SMBI - Instalación de los elementos de backend y seguridad
* SMBI - Configuración de los elementos de backend y seguridad

**Iteración 4**

Jorge

* TAG- Visualizador OpenGL 4.X simple, con datos por programa. Shader básico

Ricardo

* NM - Contextualizar modelo de datos
* ELE - Hacer guion
* ELE – Hacer guion gráfico
* ELE – Definir tecnologías
* ELE – Crear video

Nahiara

* SMBI- Hacer documentación
* PM- Registrar de tiempos y % de realización de tareas en Project
* PM- Comparar la planificación prevista y real en Project hito 1
* PM- Confeccionar Informes de iteración e informe resumen de Hito 1

Antonio

* SMBI- Implementación API RESTFULL

Manuel

* SMBI- Implementación API RESTFULL
* TAG- Visualizador OpenGL 4.X simple, con datos por programa. Shader básico

# Asignación de tiempo y recursos a actividades.

# Recursos humanos.

Nuestra intención es que las funciones y responsabilidades permitentes a RRHH sean llevadas por nosotros. Por lo tanto, somos los encargados de auto asignarnos las tareas y consensuar el tiempo de cada una.

# Otros recursos.

No consta

# Mecanismos de seguimiento y control.

Para el seguimiento del proyecto utilizaremos principalmente Microsoft Project. También, nos apoyaremos en otras herramientas como Trello para la gestión de tareas y posteriormete SeoQuacke y Google Analytics para estudiar el progreso de nuestro proyecto una vez esté funcionando. Por otro lado, también utilizaremos SourceTree y Github para la subida de ficheros y tener un seguimiento issues.

En general, monitorizaremos las siguientes tareas:

|  |  |  |
| --- | --- | --- |
| **TAREA** | **DESCRIPCIÓN** | **SOFTWARE** |
| CONTROL DE COMMITS | Haremos un control de los commits de cada integrante del grupo y se tendrá en cuenta lo que sube cada integrante siguiendo la estructura de carpetas definida a principio del proyecto. | SOURCETREE |
| GESTIÓN DE TAREAS | Para el reparto de las tareas utilizaremos la herramienta trello. De esto modo cada uno de los integrantes del grupo puede ver las tareas que quedan por hacer, que se están haciendo y las que hemos hecho. | TRELLO |
| CUMPLIMIENTO DE LAS TAREAS EN EL TIEMPO ESTIMADO | Con Microsoft Project estaremos controlando en cada momento que las tareas se cumplan como se habían estimado, haciendo las variaciones necesarias en otras tareas para llegar a tiempo. En caso de sobrepasar la estimación inicial se deberá reajustar el timming de tareas para evitar un desplazamiento general de tareas. | PROJECT |
| CONTROL DE SEO | Con la herramienta SeoQuacke tendremos un control total sobre el posicionamiento online de nuestro proyecto. Esto es importante para saber si hemos completado con éxito las tareas de posicionamiento web y sobre la difusión multimedia. | SEOQUACKE |
| REUNIÓN GRUPAL | Todos los domingos por la mañana o por la tarde nos reuniremos los 5 integrantes del grupo para hablar sobre lo que se ha cumplido esa semana, los imprevistos, comparativa de estimaciones y una vista rápida a las tareas de la semana próxima y la asignación a cada uno de los integrantes. | SKYPE |