**Proyecto**

“ALPHA-SCHOOL”

**Grupo**

“Cerberus”

"DOCUMENTO DE PLANIFICACIÓN"

Hito:

Fecha entrega: dia-mes-año

Versión: 1, 2, 3...

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:

* **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 que el ordenador personal de uno o varios de los integrantes del equipo fallara, por ejemplo.
* **Riesgo de personal:** Debido a nuestra inexperiencia es muy posible que nos encontremos ante situaciones para las que no tenemos un entrenamiento funcional, y tengamos que apoyarnos únicamente sobre conocimientos teóricos.
* **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.
* **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.

# Prioridades de los riesgos.

|  |  |  |
| --- | --- | --- |
| POSIBLE RIESGO | PROBABILIDAD | EFECTOS |
| Fallo del equipo personal de uno de los miembros | BAJA | SERIO |
| Personal inexperto | ALTA | TOLERABLE |
| Planificación errónea | MEDIA | TOLERABLE |
| Error de integración entre frameworks | BAJA | CATASTRÓFICO |
| Obtención de requisitos y análisis erróneos | MEDIA | TOLERABLE |
| Error a la hora de realizar las estimaciones de tiempo | MEDIA | SERIO |

# 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.

*<Tendréis que indicar claramente cuál es el esfuerzo del proyecto, el tiempo necesario para completarlo, número de personas a contratar y coste monetario (para ello debéis asignar un sueldo según creáis conveniente>*

# Ley de Parkinson.

# Pricing to Win.

# Puntos Objeto y/o Puntos de Función.

# Comparación y discusión de los valores obtenidos.

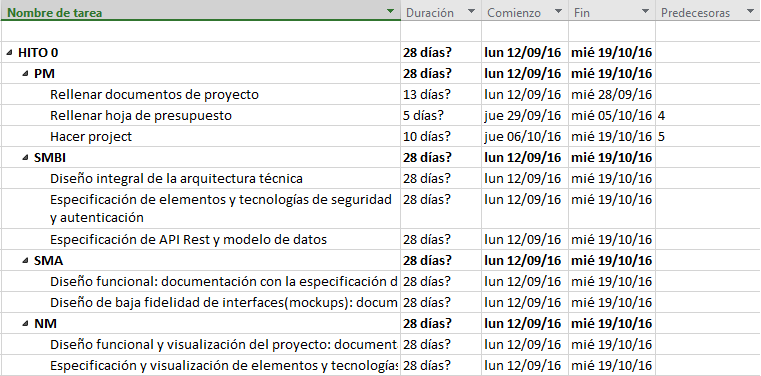
# Agenda del proyecto.

# 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 multiples formatos, salida en modo texto | Guión |
| Hacer project | Especificación de API Rest y modelo de datos | Diagrama de flujo de interfaces | Definir metricas | Cartel del proyecto | Visualizador OpenGL 4.X simple, con datos por programa. Shader básico. | Guión 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 | Trailer 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 accesibliidad) | 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 documentacion | 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 transformacion |  |
| 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 KPI'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 |  |

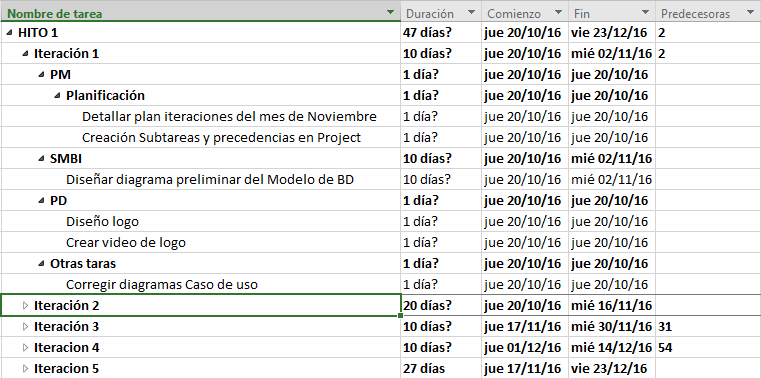
# Plan general del proyecto (Hitos).

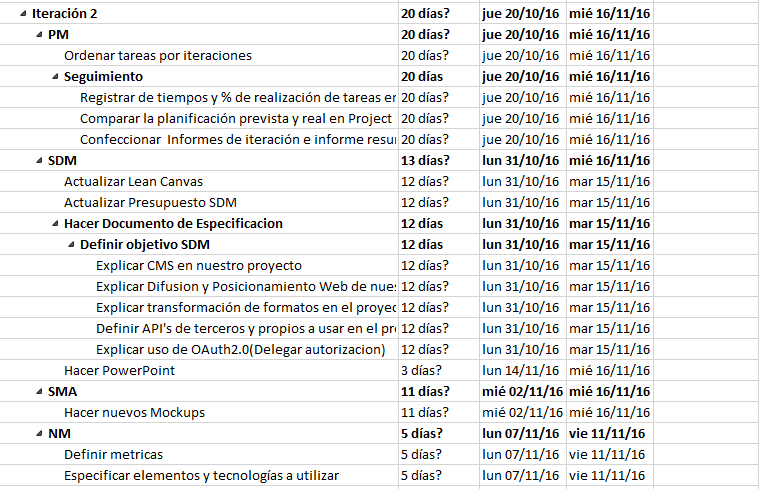
**Hito 0**

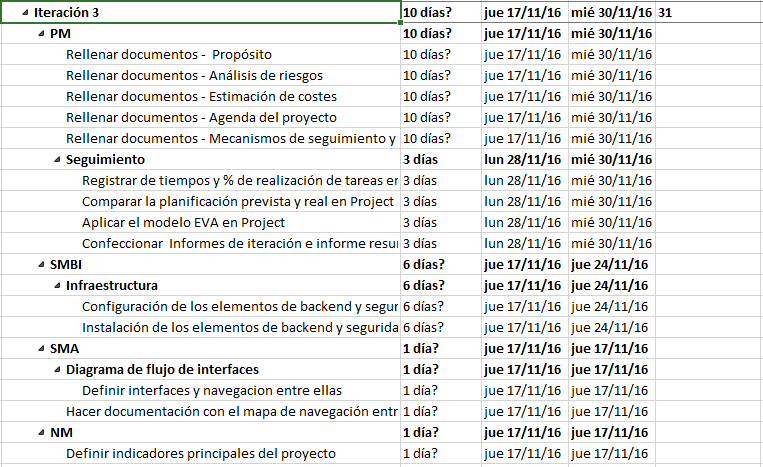
****

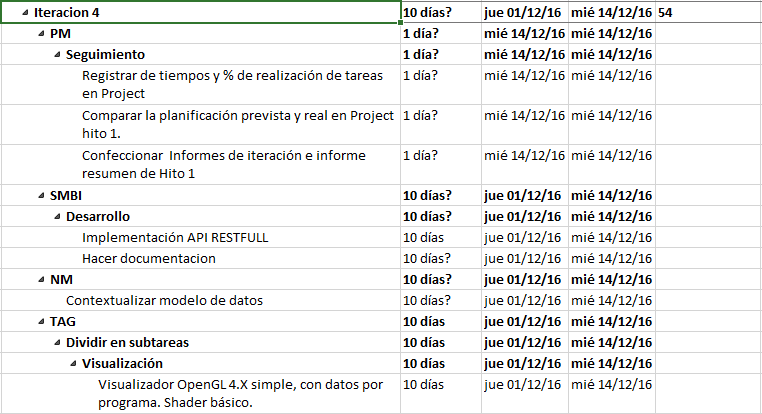
**Hito 1**

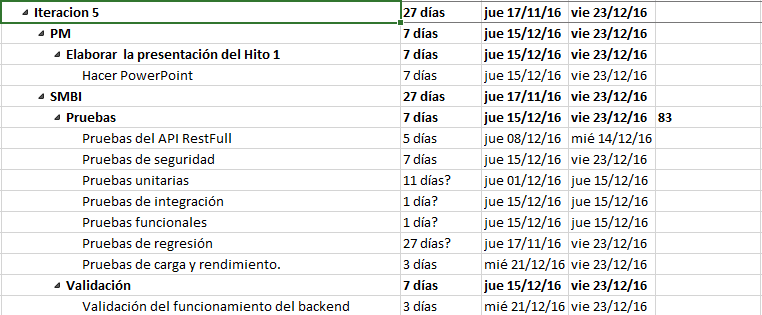
Iteración 1

****

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 - Propósito
* PM-Registrar de tiempos y % de realización de tareas en Project
* SMA-Definir interfaces y navegacion entre ellas

Ricardo

* PM-Rellenar documentos - Mecanismos de seguimiento y control
* PM-Comparar la planificación prevista y real en Project hito 1
* SMA- Hacer documentación con el mapa de navegación entre interfaces

Nahiara

* PM-Rellenar documentos - Análisis de riesgos
* PM-Aplicar el modelo EVA en Project
* SMBI-Instalación de los elementos de backend y seguridad

Antonio

* PM-Rellenar documentos - Agenda del proyecto
* SMBI-Configuración de los elementos de backend y seguridad
* NM-Definir indicadores principales del proyecto

Manuel

* PM-Rellenar documentos - Estimación de costes
* PM-Confeccionar Informes de iteración e informe resumen de Hito 1
* SMA-Hacer documentación con el mapa de navegación entre interfaces

**Iteración 4**

Jorge

* PM- Registrar de tiempos y % de realización de tareas en Project
* TAG- Visualizador OpenGL 4.X simple, con datos por programa. Shader básico

Ricardo

* PM- Comparar la planificación prevista y real en Project hito 1
* NM- Contextualizar modelo de datos
* TAG- Visualizador OpenGL 4.X simple, con datos por programa. Shader básico

Nahiara

* SMBI- Hacer documentación

Antonio

* SMBI- Implementación API RESTFULL

Manuel

* PM- Confeccionar Informes de iteración e informe resumen de Hito 1

# 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 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* |