

ESPECIFICACIÓN DE REQUISITOS.

JÁBEGA

TEAM



ÍNDICE

	Páginas
Datos de grupo:	1
Integrantes del grupo	1
Sección 1:	1
Introducción	1
Sección 2:	2-9
Requisitos funcionales	2-6
Requisitos no funcionales	6-7
Requisitos de dominio	7

Integrantes

Nombre: Iván Díaz García, Correo: ivandiazuni@gmail.com

Nombre: Ángel Campos Salido, Correo: angelcamsal06@uma.es

Nombre: Adrian Fernandez Vera, Correo: adrianfeve@gmail.com

Nombre: Alberto Sánchez Aparicio, Correo: albertoalual03@gmail.com

Nombre: Youcef Abi Ruiz, Correo: youcefatbi@uma.es

Nombre: Manuel Ruiz Campos, Correo: manuelrc@uma.es

Nombre: Antonio Jesús Díaz Plaza, Correo: antoniojesusdiazplaza@gmail.com

Nombre: Ángel Tobaruela Baños, Correo: angeltoba@uma.es

INTRODUCCIÓN- Sección 1

En este documento el equipo Jábega Team realizará una descripción sobre los requisitos necesarios para la realización del proyecto sobre la carrera de barcas.

Consistirá en una descripción detallada sobre los requisitos objetivos que cubre las necesidades del proyecto. Para su realización tuvimos una entrevista con el cliente, en la cual, realizamos diversas preguntas sobre los requisitos, tanto funcionales, como no funcionales, que nos ayudaron a hacernos una idea más clara del producto final que desea el cliente.

Para la composición de estos requisitos, hemos utilizado herramientas como Visual Paradigm Professional y el word, que nos ayudarán a reflejar todos los requisitos junto a las relaciones que comparten entre ellos.

Visual Paradigm Professional (Yousef (Universidad de Málaga))

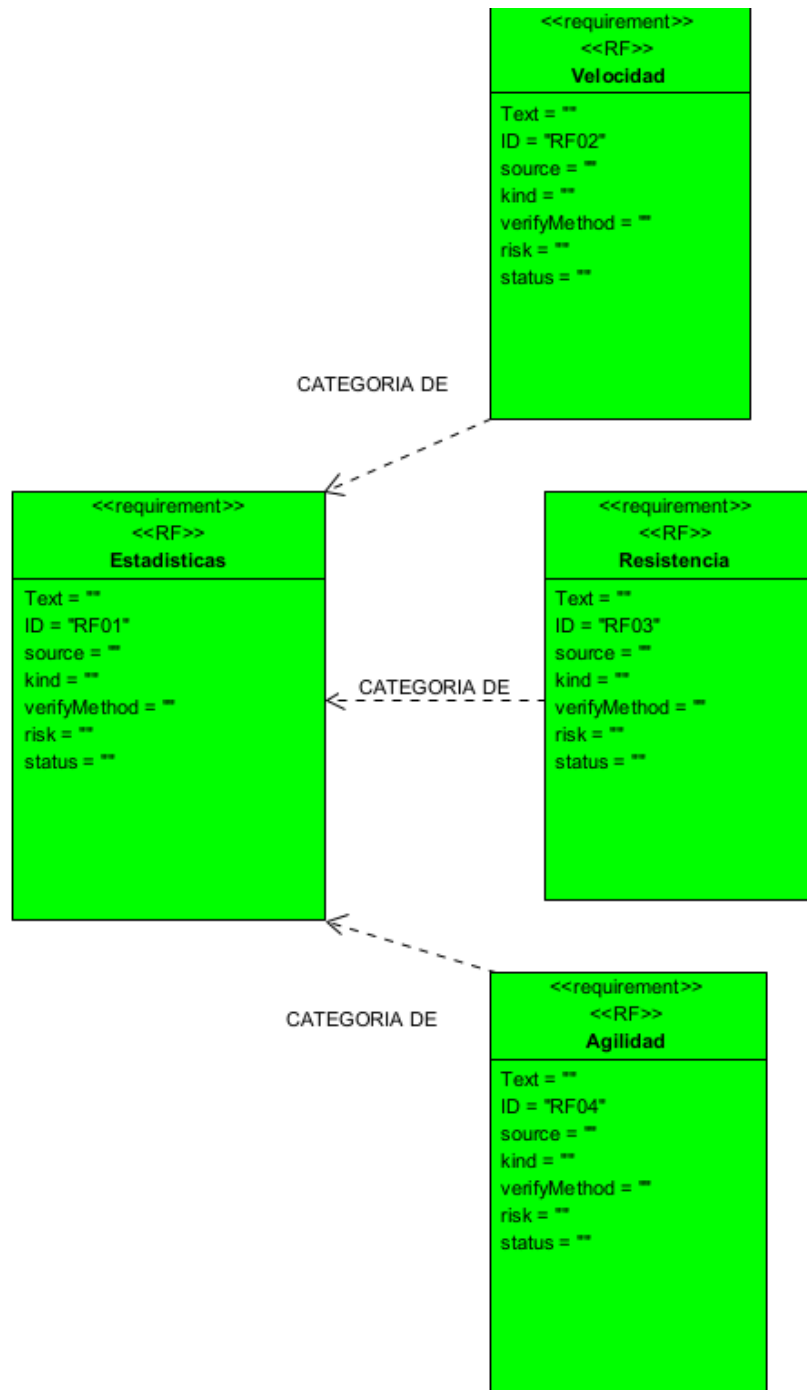


Estadística: Será necesario que las barcas cuenten con estadísticas que proporcionen variedad y jugabilidad al videojuego, dentro de cada categoría de estadística dispondremos de un rango del 1-5, siendo 5 la que cuente con esa mejor categoría de estadística. Las categorías son las siguientes:

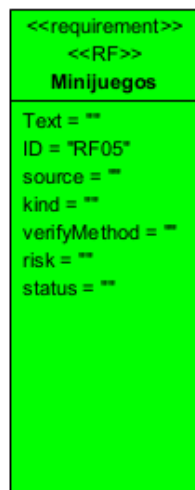
Resistencia: Cada barco tiene una resistencia, cuando un barco choca contra un obstáculo su resistencia irá disminuyendo, cuando dos barcos chocan puede que uno se rompa y otro no según sus

resistencias.(Está relacionado con las colisiones ya que en las colisiones interviene la resistencia).

Agilidad: Capacidad que tiene el barco para desplazarse hacia los lados, siendo cuanto más alta la estadística más rápido el desplazamiento



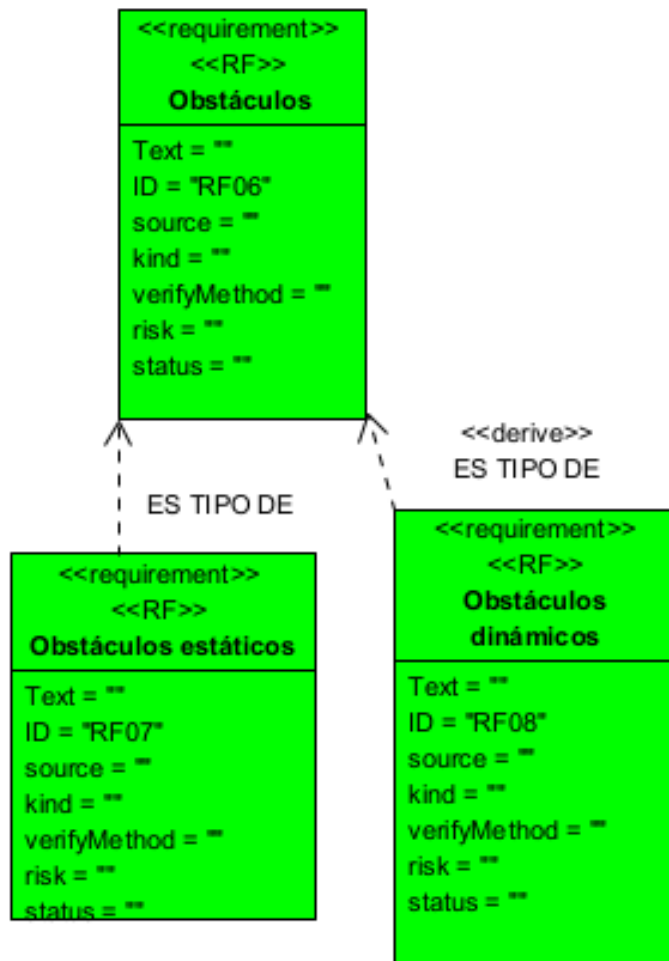
Minijuegos: Entre las diferentes carreras se implementará un minijuego con relación al tema de los dragon boats, el tema a criterio del desarrollador



Obstáculos: Será necesario que a lo largo del carril haya obstáculos con los que el usuario y la cpu interactúen. Se distinguen dos variedades dentro de los obstáculos.

Dinámicos: A lo largo de la carrera encontraremos obstáculos que se estarán moviendo, encontraremos obstáculos con movimiento lateral en el carril, como patos y obstáculos con movimiento vertical que se moverán como si fueran arrastrados por la corriente como ramas.

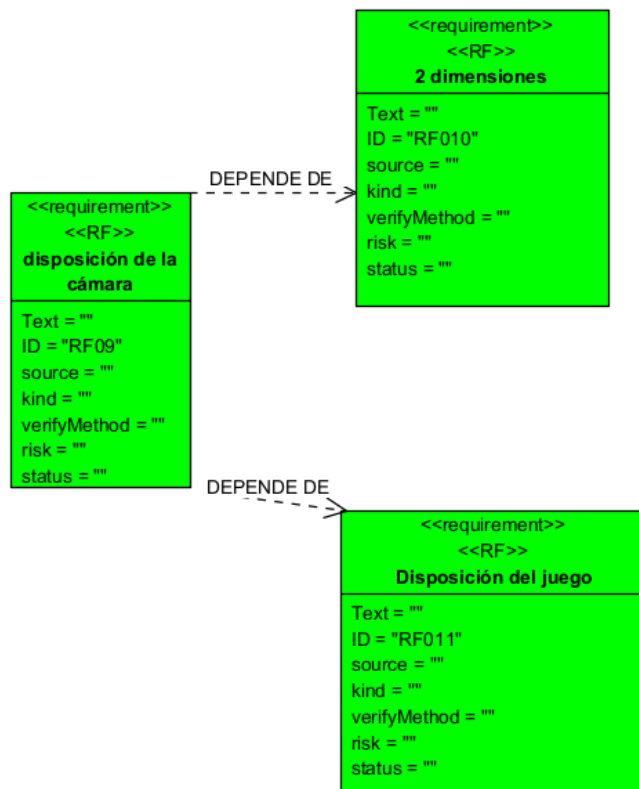
Estáticos: Durante una carrera a lo largo de cada carril encontraremos una serie de obstáculos que no se moverán a lo largo del carril.



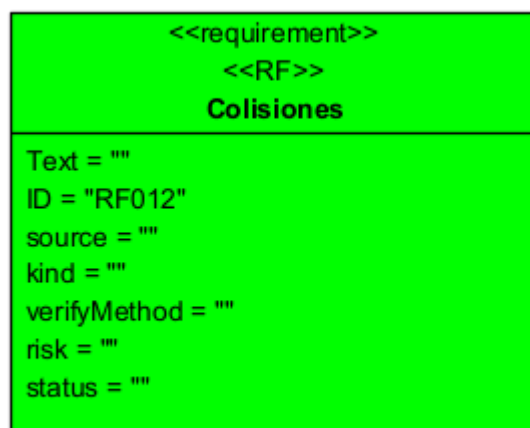
Dimensiones: El juego se hará en 2 dimensiones.

Disposición de la cámara: En forma cenital, los ríos vistos desde arriba. (una relación de dependencia con dimensiones, ya que si fuera en 3d por ejemplo no lo haríamos desde arriba).

Disposición de la pantalla: La pantalla se dividirá en 4 carriles verticales que dividirán el río, también hay relación de dependencia con la disposición de la cámara.



Colisiones: Los barcos al colisionar perderán energía de la barra de energía y además perderán velocidad, los barcos pueden colisionar con los obstáculos o con otros barcos al cambiarse de carril.



Movimiento: Los barcos no se mueven por sí solos, si no apretamos alguna tecla no se moverán, no se podrán desplazar hacia atrás y los barcos girarán determinados grados al cambiar de carril.

<<requirement>> <<RF>> Movimiento
Text = "" ID = "RF013" source = "" kind = "" verifyMethod = "" risk = "" status = ""

Interfaz: El juego tendrá una menú de inicio para empezar el juego y elegir los barcos y tras finalizar la carrera pondrá un posicionamiento.

<<requirement>> <<RF>> Interfaz
Text = "" ID = "RF014" source = "" kind = "" verifyMethod = "" risk = "" status = ""

Requisitos no funcionales

Los requisitos no funcionales son las restricciones o los requisitos impuestos al sistema. Especifican el atributo de calidad del software. Los requisitos no funcionales se ocupan de problemas como la escalabilidad, la mantenibilidad, el rendimiento, la portabilidad, la seguridad, la confiabilidad y muchos más. Los requisitos no funcionales abordan cuestiones vitales de calidad para los sistemas de software. En este artículo, intentaremos comprender los requisitos no funcionales en detalle con algunos ejemplos.

Tasa de refresco(FPS): Tasa de refresco de juego: La tasa de refresco del juego será preferiblemente de 30 FPS.

<div><<requirement>> <<RNF>> Tasa de refresco juego</div>
<div>Text = "" ID = "RNF01" source = "" kind = "" verifyMethod = "" risk = "" status = ""</div>

Mapa: Un gráfico nos mostrará dónde se encuentra cada barco a lo largo de la carrera, ya que la cámara seguirá nuestro barco y los demás pueden quedar atrás.

<div><<requirement>> <<RNF>> Mapa</div>
<div>Text = "" ID = "RNF02" source = "" kind = "" verifyMethod = "" risk = "" status = ""</div>

Requisitos de Dominio

Los requerimientos del dominio se derivan del dominio de aplicación del sistema más que de las necesidades específicas de los usuarios. Normalmente incluyen terminología especializada del dominio o referencia a conceptos del dominio.

Originalidad: Es un requisito legal y de dominio, en el que se busca un juego innovador y original, para que no puedan recibir problemas legales sobre el plagio del software a otro proyecto.

<div><div><<requirement>></div><div><<RNF>></div><div>Originalidad</div></div>
<div>Text = ""</div> <div>ID = "RNF03"</div> <div>source = ""</div> <div>kind = ""</div> <div>verifyMethod = ""</div> <div>risk = ""</div> <div>status = ""</div>

