

Documento DE ELICITACIÓN DE REQUERIMIENTOS DE SOFTWARE

Francisco Manuel Mares Solano



12 de agosto de 2017

Universidad Veracruzana

Contenido

[Introducción 2](#_Toc490339040)

[Propósito 2](#_Toc490339041)

[Alcance 2](#_Toc490339042)

[Información general 2](#_Toc490339043)

[Descripción general 3](#_Toc490339044)

[Perspectiva del producto 3](#_Toc490339045)

[Funcionalidad del producto 3](#_Toc490339046)

[Características del producto 3](#_Toc490339047)

[Restricciones 3](#_Toc490339048)

[Supuestos y dependencias 3](#_Toc490339049)

[Requisitos específicos 4](#_Toc490339050)

[Selección de lenguaje y herramientas 4](#_Toc490339051)

[Requisitos funcionales 4](#_Toc490339052)

[Diagrama de Casos de Uso 5](#_Toc490339053)

[Descripción de los casos de uso 6](#_Toc490339054)

[Diagramas de robustez de cada CU 7](#_Toc490339055)

[Diagramas de secuencia de cada CU 7](#_Toc490339056)

[Diagrama de clases 7](#_Toc490339057)

[Plan de pruebas 8](#_Toc490339058)

# Introducción

Este documento fue creado con el fin de establecer los requerimientos funcionales, características y limitaciones del sistema, así como justificar el motivo de existir del mismo.

## Propósito

## Alcance

La versión inicial del sistema permitirá calcular la deflexión y ángulos de las mismas en vigas sometidas a cargas con apoyos isoestáticos. Es necesario que el usuario cuente con datos técnicos básicos para poder completar las entradas del sistema.

El cálculo de los valores será realizado a través del método de funciones discontinúas, para lo cual el usuario contará con una interfaz gráfica que le permitirá determinar el módulo de elasticidad para el material trabajado, así como diseñar el perfil de la viga a través de figuras simples o escogerlo de tablas normalizadas; posteriormente indicará las cargas, apoyos y longitudes que podrá visualizar en una representación gráfica para finalmente determinar los resultados correspondientes.

Para la correcta compresión y lectura de este documento se sugieren los siguientes términos descritos en la tabla siguiente.

Tabla Tabla de definiciones, abreviaciones y acrónimos

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| Grid | Una tabla en la interfaz maneja con coordenadas. |
| Perfil | La forma y área de la sección transversal de una viga |

## Información general

Un juego estilo batalla naval desarrollado de forma de multijugador, con sistema de puntuación y registro de usuarios

# Descripción general

## Perspectiva del producto

1. Se pretende que el juego pueda hacer partidas mediante el servidor, uniendo a un cliente y un servidor.
2. Que los jugadores disfruten del juego
3. Que los jugadores tengan enfrentamientos competitivos por medio del sistema de puntuaciones.

## Funcionalidad del producto

1. Guardar partidas en tiempo real.
2. Sistema de Puntación.
3. Juego de conexión entre clientes.

## Características del producto

Al principio de cada partida, los jugadores deberán colocar los barcos en su tablero; los barcos podrán girar y ser acomodados de la manera que el usuario prefiera siempre que sea válida. Al final de que cada jugador asigne sus 6 barcos, se procederá a iniciar la partida, es decir, cuando ambos jugadores estén listos. El jugador primero en tirar será seleccionado al azar; al jugador que le toque tirar, podrá ver el tablero enemigo en el cual seleccionará la casilla donde quiere tirar, mientras el otro jugador verá su propio tablero, seguido de eso, será turno del otro jugador, donde él verá el tablero del contrincante y el contrincante pasara a su propio tablero.

Al momento de tirar se le notificara al usuario si acertó o si fallo al momento de realizar su jugada.

Gana aquel jugador que logre eliminar con éxito a todos los barcos enemigos.

El sistema de puntuación será conforme a los puntos acertados, de ambos jugadores, al ceder o retirarse de la partida, se sobreentiende que se le darán los puntos restantes al jugador contrincante.

## Restricciones

1. El sistema no puede trabajar con datos incompletos

## Supuestos y dependencias

1. El sistema cuenta con Windows 10 Aniversary Edition o superior.

# Requisitos específicos

## Selección de lenguaje y herramientas

Debido a la experiencia previa con el lenguaje C#, se ha decidido hacer uso de este lenguaje junto al entorno de desarrollo (IDE) Visual Studio, así como al lenguaje de marcado XAML para el desarrollo del sistema pues gracias a las herramientas de revisión de código, bibliotecas y la amplia comunidad de personas que hacen uso del lenguaje, el desarrollo puede alcanzar una madurez, seguridad y velocidad adecuadas para cumplir con los plazos estipulados para la realización del proyecto.

C# ofrece una compatibilidad completa con el sistema operativo Windows, haciendo uso del API directo del sistema siendo, además, el lenguaje recomendado por Microsoft para el desarrollo en su plataforma, por tanto es un lenguaje que nos proporciona robustez, compatibilidad y una considerable velocidad de desarrollo, además, ofrece características avanzadas de programación las cuales facilitan el cálculo matemático, así como el manejo de la interfaz gráfica así como una amplia documentación, estabilidad y gran compatibilidad con extensiones de terceros.

Por su parte, Visual Studio es una herramienta bastante completa al momento del desarrollo de aplicaciones, con una integración perfecta con C#, así como herramientas avanzadas exclusivas para el mismo como análisis estáticos y dinámicos del código desarrollo, repositorio universal centralizado de bibliotecas, además como un manejo completo del patrón MVC (Modelo-Vista-Controlador).

Además de todo lo anterior, XAML como lenguaje de marcado para la interfaz ofrece facilidades a la hora del desarrollo como elementos visuales personalizados, enlace rápido de variables y una separación adecuada del código y la interfaz, lo cual permite una mejor modularización, una mejor mantenibilidad y un escalamiento más sencillo.

## Requisitos funcionales

1. Se necesita un sistema de puntuación para los usuarios.
2. Se requiere una lista de usuarios conectados al juego
3. Un chat global para los jugadores
4. Se debe permitir el registro a nuevos usuarios.
5. Autoguardado por parte del servidor.
6. El servidor debe estar orientado a eventos.
7. La base de datos debe ser externa y puede correr junto con el servidor

# Diagrama de Casos de Uso

# Descripción de los casos de uso

# Diagramas de robustez de cada CU

Figura Diagrama de robustez CRUD Colaboradores

# Diagramas de secuencia de cada CU

# Diagrama de clases

# Plan de pruebas