

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 FlexUV, así como justificar el motivo de existir del mismo. Además, incluye los diseños necesarios para el desarrollo del sistema. El documento está basado en el estándar 830-1984 de la IEEE adaptado para cumplir los requerimientos de análisis y diseño planteado por la metodología de ICONIX para desarrollo de software de alta calidad.

Este documento está dividido en 3 secciones principales. En la primera se encuentra una visión general del objetivo del software, así como una descripción inicial de los criterios de aceptación planteados para el sistema e información que permitirá establecer un idioma y entendimiento común entre el lector y el documento. En la segunda sección se encuentran establecidos los requerimientos del sistema, aquí se encuentran los requerimientos de funcionalidad y calidad, se describe a mayor detalle las necesidades del sistema y se plantean diseños que permiten analizar el sistema que se planea desarrollar.

En la 3ra sección de este documento se halla la especificación de diseño. En esta sección se describe a mayor detalle la funcionalidad final que poseerá el sistema, su interacción con los siguientes módulos, así como el cumplimiento con los criterios de calidad, los requerimientos y los criterios de aceptación planteados anteriormente por medio del diseño del sistema y la descripción de pruebas realizadas.

## Propósito

El propósito del sistema que se ha desarrollado es establecer un primer acercamiento a la creación de software especializado para la carrera de ingeniería mecánica de la universidad veracruzana.

## Alcance

* La versión inicial del sistema permitirá calcular la deflexión y ángulos en vigas bajo las siguientes características:
  + Viga simplemente apoyada
  + Viga voladiza
  + Viga empotrada con rodillo y carga al centro
  + Viga empotrada con rodillo y carga a la izquierda
  + Viga doblemente empotrada con carga al centro
  + Viga doblemente empotrada con carga a la derecha
  + Carga distribuida en 3 apoyos equidistantes
* Los valores permisibles estarán determinados en función del tipo de viga y deflexión elegida por el usuario, permitiendo, por medio de buenas prácticas en el desarrollo de software, ampliar estas capacidades para una posible expansión del sistema.
* Toda la información introducida por el usuario en esta versión serán datos numéricos y deberán ser agregados en campos de texto por el usuario, o en su defecto, por medio de tablas en la interfaz, desistiendo de una capacidad gráfica para diseñar el ejercicio a analizar.
* El sistema será desarrollado para la plataforma Windows para facilitar su creación.

## Limitaciones

* El sistema actual no incluirá un sistema administrativo para permitir una expansión sencilla por parte de cualquier usuario.
* El sistema no soportará creación de cuentas de usuario o algún otro mecanismo que permita obtener retroalimentación de análisis realizados con anterioridad en el sistema.
* El sistema no está planeado para una distribución a nivel empresarial o masiva del mismo. Resulta un mero acercamiento a este enfoque.
* La tecnología usada para el desarrollo será UWP para permitir un uso simple del API que ofrece la nueva plataforma de desarrollo, así como las nuevas características visuales que provee.

## Definiciones, abreviaturas y acrónimos

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 |
| S.O. | Sistema operativo |
| UML | Unified Modeling Languaje. Es el lenguaje de diseño unificado para la creación de modelos de software usado como estándar de manera mundial |
| UWP | UWP es el nuevo framework de desarrollo de aplicaciones para Windows. Permite el funcionamiento de la aplicación en diferentes formatos de dispositivos, así como provee acceso a las últimas características del sistema. |
| Actor | Usuario que interactuará con el sistema |

## 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 sistema permita realizar cálculos mecánicos simples y complejos sobre diferentes tipos de vigas.
2. Que el software facilite la validación y compresión del análisis de vigas.
3. Ofrecer un primer acercamiento para el desarrollo interno de software profesional en la Universidad Veracruzana.
4. Que los usuarios posean un método práctico y conveniente para realizar análisis de vigas que faciliten su trabajo en diversos campos disminuyendo la complejidad de los problemas que pueden afrontar durante la carrera.

## Funcionalidad del producto

1. Selección de alguno de los diversos tipos de viga soportados.
2. Sistema amigable para la introducción de datos.
3. Cálculo de vigas por el método seleccionado.

## Características del producto

El sistema mostrará de inicio una interfaz por medio de la cual, el usuario tendrá la capacidad de seleccionar, de entre un número considerable de cálculo de vigas (7), el método que le provea el o los resultados deseados. A continuación, el usuario se encontrará ante una interfaz familiar para los usuarios del S.O. Windows dividida en 3 secciones con diferentes parámetros que el usuario tendrá que incluir para determinar las características del ejercicio que se está analizando. Finalmente, cuando el usuario ha incluidos todos los datos, el sistema se encargará de analizar el cálculo correspondiente de acuerdo con las necesidades descritas por el usuario con la introducción de la información, para después mostrarla en el sistema de manera simple para el usuario. Además, durante el uso del sistema se deben tener en cuenta las siguientes características:

* El sistema informará al usuario sobre cualquier dato introducido incorrectamente.
* Una vez que el usuario termine el análisis del ejercicio, el sistema no permitirá regresar para editar la información introducida.

## Restricciones

1. El sistema no puede trabajar con datos incompletos.
2. El sistema no permite una alta complejidad en las vigas.

## Supuestos y dependencias

1. El sistema cuenta con *Windows 10* *Creators Update* o superior.
2. El sistema permite la instalación de UWP de fuentes procedentes diferente a la Microsoft Store.

# 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 validez adecuada para las necesidades del proyecto.

C# ofrece una compatibilidad completa con el sistema operativo Windows haciendo uso del API principal del sistema, siendo, además, el lenguaje recomendado por Microsoft para el desarrollo en su plataforma, proveen al lenguaje robustez, compatibilidad y una considerable velocidad de desarrollo, así como también ofrece características avanzadas de programación las cuales facilitan el cálculo matemático incluyendo 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) y el API de las aplicaciones UWP, este IDE se convierte en la elección indiscutible para el desarrollo del sistema.

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

Los requerimientos funcionales componen las características que requiere el software para considerarse un programa completo, válido, funcional y capaz de dar solución a una problemática. En cuestiones de este nuevo proyecto, las características funcionales son las siguientes:

1. El sistema requiere un sistema que permita al usuario seleccionar los diferentes tipos de análisis que se podrán realizar
2. Se requiere un sistema que, desde su arquitectura, esté planteado para poder crecer con facilidad
3. Permita el análisis adecuado de cada uno de los diferentes métodos permitidos por el sistema

## Metodología

Para el análisis y diseño de la nueva plataforma se ha escogido la metodología ICONIX como guía de proceso. Esta decisión está fundamentada en la facilidad que proporciona para los principiantes en desarrollo de software, de hacer uso de la misma, en que tal metodología corresponde a las metodologías ágiles y, por tanto, hereda sus características como la capacidad del diseño y desarrollo de un sistema en tiempos cortos de entrega, que permite a equipos pequeños (o a un solo integrante) realizar la gestión completa de todas las actividades de manera simple, en que los diseños propuestos por la metodología están adaptados para cualquier software en general y que proveen una flexibilidad para el diseñador de guiar el proceso en las interfaces, lo cuál permite realizar un software amigable con las personas menos experta en computación.

ICONIX es, además, una metodología ágil, esto implica que sigue los principios del manifiesto ágil, el cuál propone ciertas consideraciones que deben cumplir las metodologías de diseño de software para permitirle un diseñador, realizar diseño y desarrollos en tiempos adecuados cumpliendo con las características de calidad deseadas. Gracias a esto, ICONIX posee la particularidad de permitir que los modelos de diseño creados cambien conforme el propio diseño y el sistema evolucionan.

Para una definición de requerimientos de sistema adecuada, de acuerdo con la propuesta de la metodología ICONIX para la elaboración de software de calidad, se propone la creación de los siguientes artefactos:

* Modelo de dominio
* Modelo de casos de uso
* Prototipos de interfaz de usuario
* Descripción detallada de casos de uso
* Diagrama de clases
* Diagramas de robustez
* Modelos de secuencia

Los cuáles pueden ser agrupados en diseño estático y diseño dinámico. El diseño estático está constituido por el Modelo de dominio, Modelo de casos y Prototipos de interfaz de usuario; Mientras que el diseño dinámico está compuesto por las Descripciones detalladas de casos de uso, Diagrama de clases, Diagramas de robustez y Modelos de secuencia.

El primer grupo de modelos, correspondiente al diseño estático, está planteado con la finalidad de ofrecer un modelado preciso y completo de la información implicada en el mismo sin considerar su interacción o los flujos de información. Para el segundo grupo de modelos, correspondiente al diseño dinámico, la intención es modelar de manera lo más precisa posible, la interacción entre los componentes que compondrán el software final y permitir la refinación de los diferentes mecanismos que permitirán alcanzar los estándares de calidad planteados (características de calidad) así como la funcionalidad correcta.

## Requerimientos del sistema

Como se mencionó, los requerimientos del sistema están conformados por el modelo de dominio, el diagrama de casos de uso y los prototipos de interfaz de usuario. A continuación, se presenta una breve descripción en la cual se detalla, de manera general, los objetivos de cada uno de los diseños presentados en esta sección.

**Modelo de dominio.**

Este modelo es usado por los analistas de software como un medio para comprender el sector en el cual el sistema funcionará. Permite además establecer un idioma común el cuál será usado a lo largo de las diferentes fases del desarrollo del sistema y contribuye a identificar personas, eventos, transacciones y objetos involucrados (Ecured, 2017).

**Diagrama de casos de uso.**

Los casos de uso describen una secuencia de interacciones entre el sistema y un actor externo que resultan en información de valor para el actor. Proveen una vista de alto nivel sobre los requerimientos del sistema. Los casos de uso no indican el flujo de trabajo que se presenta sino las posibles interacciones *atómicas (*que por sí *mismas* son una sola acción completa e indivisible para generar valor*)* (Wiegers, K. 2013)*.*

**Prototipos de interfaz de usuario**

Permite que los requerimientos planteados en el diagrama de casos de uso sean más realistas y las brechas de entendimiento entre el producto que se busca y los requerimientos que se plantean, sean encaminados a hallar una única solución que satisfaga todas las necesidades. Los prototipos permiten plantear la interacción del usuario y descartar o incluir, desde una fase temprana, características del sistema.

En los apartados siguientes se presentan los modelos descritos (Figura 1, Figura 2 e Figura 3).

### Modelo de dominio

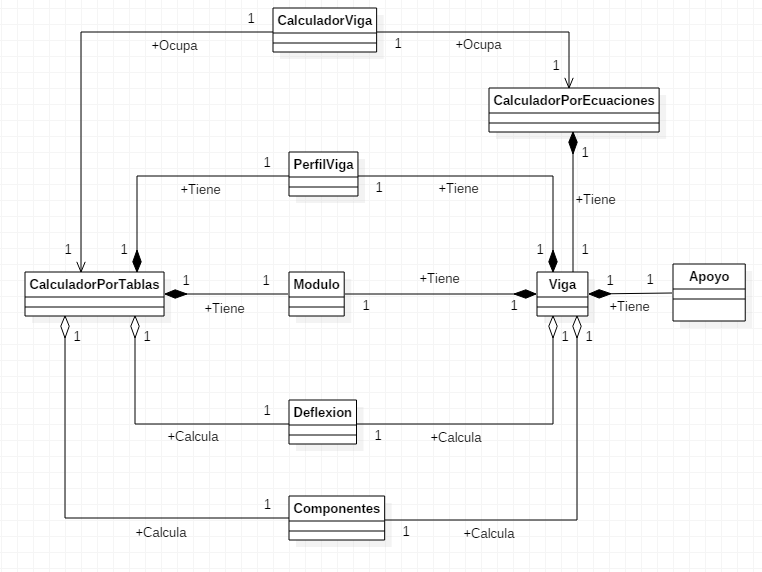


Figura Modelo de dominio

### Diagrama de Casos de Uso

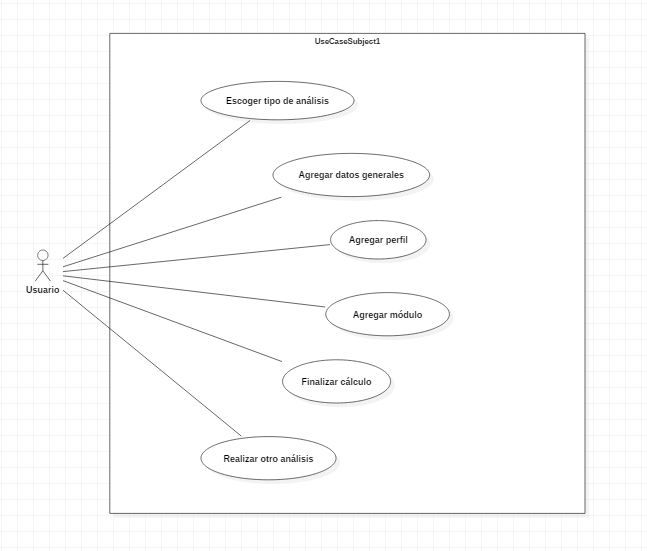


Figura Diagrama de Casos de Uso

### Prototipos de interfaz de usuario

#### Menú principal

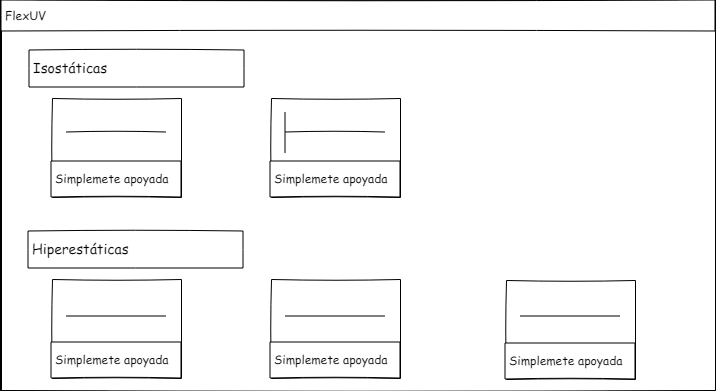


Figura Prototipo Menú principal

#### Viga simplemente apoyada

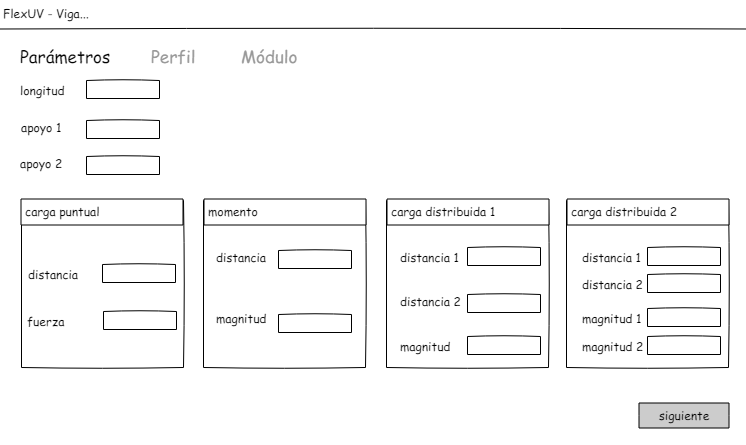


Figura Prototipo Viga simplemente apoyada

#### Viga voladiza

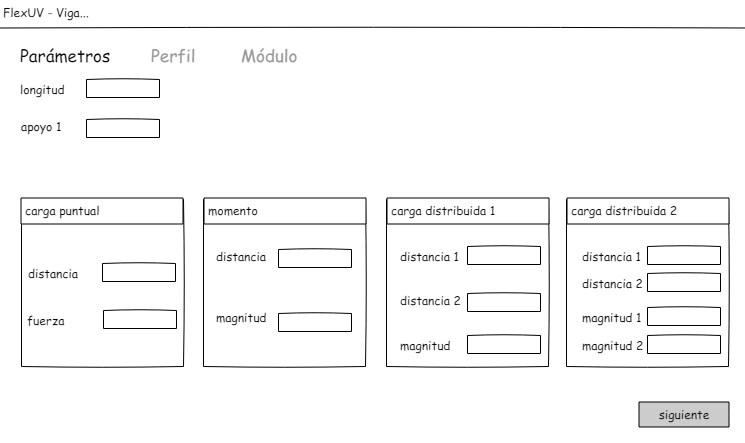


Figura Prototipo Viga voladiza

#### Viga empotrada con rodillo y carga al centro

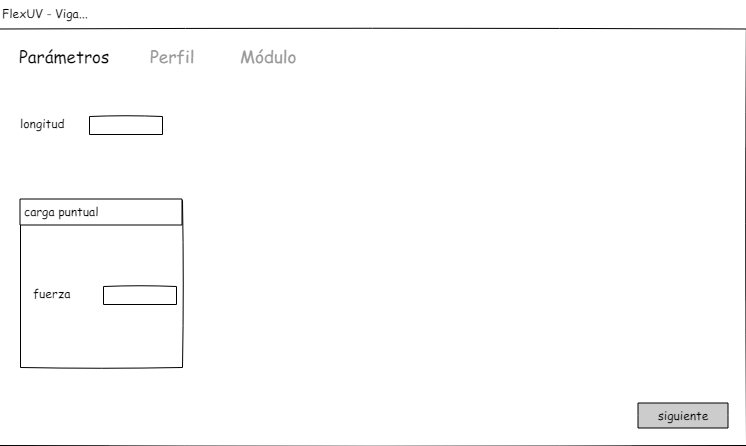


Figura Prototipo Viga empotrada con rodillo y carga al centro

#### Viga empotrada con rodillo y carga a la izquierda

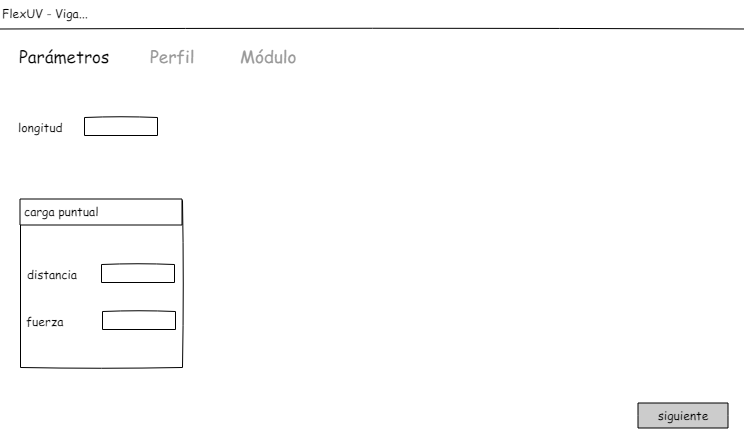


Figura Prototipo Viga empotrada con rodillo y carga a la izquierda

#### Viga doblemente empotrada con carga al centro

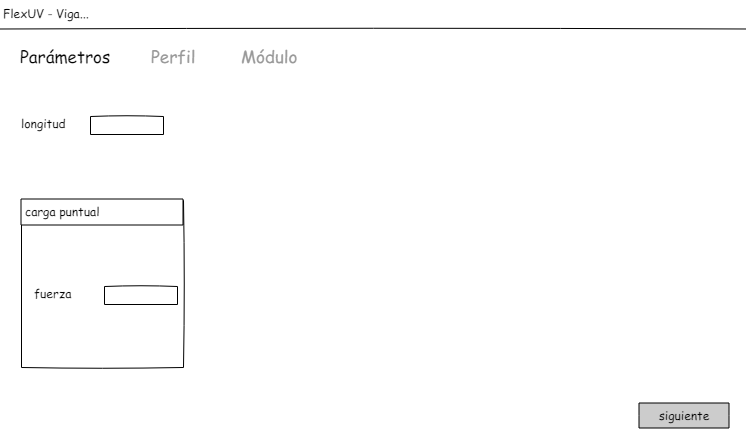


Ilustración Prototipo Viga doblemente empotrada con carga al centro

#### Viga doblemente empotrada con carga a la derecha

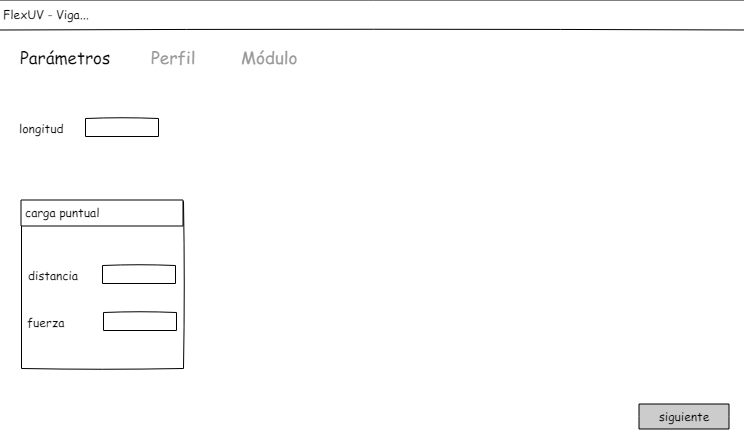


Ilustración Prototipo Viga doblemente empotrada con carga a la derecha

#### Carga distribuida en 3 apoyos equidistantes

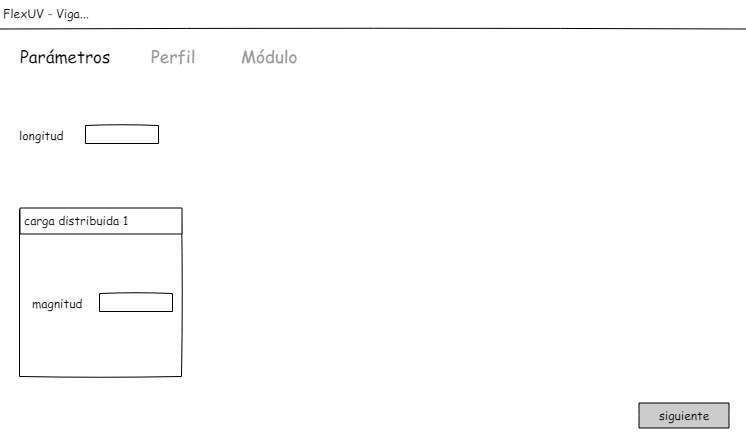


Ilustración Prototipo Carga distribuida en 3 apoyos equidistantes

# Diseño del sistema

Respecto al diseño del sistema, ICONIX propone la creación de los siguientes modelos: Descripción detallada de casos de uso, Diagrama de clases, Diagramas de robustez y Modelos de secuencia. A continuación, se presenta una breve descripción en la cual se detalla, de manera general, los objetivos de cada uno de los diseños presentados en esta sección.

**Descripción detallada de casos de uso.**

Una descripción de casos de uso es una colección de escenarios relacionados usables en el sistema, siendo un escenario, un momento específico del caso de uso. Su objetivo es proveer una descripción más exacta sobre cómo debería ser el flujo de un sistema propuesto para así, producir un diseño cada vez más cercano al código y que cumpla, adecuadamente, con los requerimientos detallados hasta ahora. Debe existir una trazabilidad entre estos y los modelos generados anteriormente pues son indicativo de que los requerimientos funcionales y las características de calidad son cumplidas (Wiegers, K. 2013).

**Diagrama de clases.**

Un diagrama de clases es un método gráfico para representar las clases identificadas durante el análisis del problema, así como sus relaciones. Su enfoque es ser un producto orientado a objetos que permita modelar el problema considerando los requerimientos planteados y la arquitectura que tendrá el software. Este diagrama expresa los atributos de las clases, su comportamiento, sus propiedades, así como las relaciones entre esta y las demás clases y no necesariamente refleja todos los elementos y/o relaciones hallados en el modelo de dominio.(Wiegers, K. 2013)*.*

**Diagramas de robustez**

El objetivo de los diagramas de robustez es añadir nuevas relaciones entre las clases del diagrama de clases. Permite, además, crear un diseño inicial el cuál se apegue a los requerimientos detectados durante el análisis.

**Modelos de secuencia**

Los diagramas de secuencia se enfocan específicamente en las "líneas de vida" de un objeto y cómo se comunican con otros objetos para realizar una función antes de que la línea de vida termine. Es decir, permiten modelar los objetos del modelo de clases, su existencia y sus interacciones para asegurar que el modelo planteado se refine y produzca un desarrollo exitoso.

En los apartados siguientes se presentan los modelos descritos (Figura 1, Figura 2 y Figura 3).

## Descripción de los casos de uso

Escoger tipo de análisis

Agregar datos generales

Agregar Perfil

Agregar Módulo

Finalizar cálculo

Realizar otro análisis

### Escoger tipo de análisis

Tabla 2 Tabla de caso de uso Condicionar el sistema.

|  |  |
| --- | --- |
| **ID:** | CU\_01 |
| **Nombre:** | Escoger tipo de análisis. |
| **Autor(es):** | Mares Solano Francisco Manuel |
| **Fecha de creación:** | 12/08/2017 |
| **Fecha de actualización:** | 15/04/2018 |
| **Actor(es):** | Usuario |
| **Descripción:** | Este caso de uso inicia cuando el usuario proporciona las condiciones del sistema y confirma visualmente el diseño para obtener resultados. |
| **Precondiciones:** | **PRE01-** Tener un perfil hecho. |
| **Flujo Normal:** | **Flujo normal**  1a. El usuario indica las condiciones a analizar en el sistema y terminar al indicarlo con el botón <Vista previa>.  2a. muestra el diseño de la viga con los elementos seleccionados previamente.  3a. El usuario confirma visualmente el sistema con la opción <Aceptar>.  4a. El sistema calcula las deflexiones y ángulos y los muestra en pantalla.  5a. El usuario adquiere la información requerida y decide terminar el proyecto.  6a. El sistema cierra el proyecto. |
| **Flujos Alternos:** | **Flujos alternos**  **General nueva sección**  1b. El usuario desea agregar una nueva sección en la viga con diferente perfil y lo indica con el botón del panel izquierdo <agregar perfil>.  2b. El sistema regresa al caso de uso *Escoger perfil.*  **Modificar sistema.**  3c. El usuario desea modificar el sistema y lo indica con el botón <Cancelar>.  4c. El sistema regresa al flujo normal 1a.  **Calcular otro sistema.**  5d. El usuario indica que desea calcular los datos de un sistema diferente con el botón <calcular otro sistema>.  6d. El sistema regresa al flujo normal 1a. |
| **Excepciones:** | **Campos vacíos.**  1e. El usuario seleccionó considerar campos que no rellenó.  **Valores no válidos.**  1f. El usuario indicó valores numéricos no naturales o caracteres ajenos.  **Valores no operables.**  3g. El usuario proporcionó valores sumamente desproporcionados. |
| **Poscondiciones:** |  |
| **Prioridad:** | Indispensable. |

### Escoger perfil

Tabla Tabla de caso de uso Escoger perfil.

|  |  |
| --- | --- |
| **ID:** | CU\_01 |
| **Nombre:** | Escoger perfil. |
| **Autor(es):** | Mares Solano Francisco Manuel |
| **Fecha de creación:** | 12/08/2017 |
| **Fecha de actualización:** |  |
| **Actor(es):** | Usuario |
| **Descripción:** | Este caso de uso inicia cuando el usuario desea escoger el perfil de la viga con la que trabajará, para escoger entre diseñarlo por su cuenta y escoger un perfil estandarizado. |
| **Precondiciones:** |  |
| **Flujo Normal:** | **Flujo normal**  1a. El usuario inicia el sistema.  2a. El sistema presenta las opciones al usuario.  3a. El usuario escoge una de las opciones.  4a. El sistema envía al usuario a la opción seleccionada. |
| **Flujos Alternos:** |  |
| **Excepciones:** |  |
| **Poscondiciones:** | **POST01-** El usuario puede empezar a diseñar su perfil con el módulo de Young.  **POST02-** El usuario puede escoger el perfil de su viga. |
| **Prioridad:** | Indispensable |

### Escoger módulo de la viga personalizada.

Tabla 2 Tabla de caso de uso Escoger módulo de la viga personalizada.

|  |  |
| --- | --- |
| **ID:** | CU\_02 |
| **Nombre:** | Escoger módulo de la viga personalizada. |
| **Autor(es):** | Mares Solano Francisco Manuel |
| **Fecha de creación:** | 12/08/2017 |
| **Fecha de actualización:** |  |
| **Actor(es):** | Usuario |
| **Descripción:** | Este caso de uso inicia cuando el usuario desea asignar un módulo de elasticidad para la viga personalizada. |
| **Precondiciones:** | **PRE01-** Haber seleccionado una forma de crear el perfil. |
| **Flujo Normal:** | **Flujo normal**  1a. El usuario da nombre a su perfil y escoge asignar el módulo o seleccionar de la tabla.  2a. El sistema habilita la opción **módulo** **de elasticidad**.  3a. El usuario da valor al módulo y selecciona <terminar>.  4a. El sistema envía al usuario la página para diseñar la viga. |
| **Flujos Alternos:** |  |
| **Excepciones:** |  |
| **Poscondiciones:** | **POST01-** El usuario puede escoger las partes que conformarán su viga. |
| **Prioridad:** | Indispensable. |

### Insertar sección del perfil.

Tabla 3 Tabla de caso de uso Insertar sección del perfil.

|  |  |
| --- | --- |
| **ID:** | CU\_03 |
| **Nombre:** | Insertar sección del perfil. |
| **Autor(es):** | Mares Solano Francisco Manuel |
| **Fecha de creación:** | 12/08/2017 |
| **Fecha de actualización:** |  |
| **Actor(es):** | Usuario |
| **Descripción:** | Este caso de uso inicia cuando el usuario desea agregar una sección al diseño de la pieza y le permite escoger tanto la forma como las dimensiones para colocarla, con base en un sistema de puntos de referencias, como parte del perfil de la viga. |
| **Precondiciones:** | **PRE01-** haber seleccionado un módulo para la viga. |
| **Flujo Normal:** | **Flujo normal**  1a. El usuario escoge una de las secciones para insertar.  2a. El sistema pide las dimensiones de la figura.  3a. El usuario da proporción a los valores correspondientes y selecciona la opción <Aceptar>.  4a. El sistema solicita un punto de referencia en la figura.  5a. El usuario selecciona un punto de su conveniencia y continúa con la opción<Listo>.  6a. El sistema pregunta por un punto de referencia para colocar la figura en el Grid.  7a. El usuario selecciona el punto donde desea colocar la sección y da la opción <Aceptar>.  8a. El sistema agrega la sección correspondiente.  9a. El usuario concluye su diseño con la opción <Terminar>.  10a. El sistema envía a la página del diseño del sistema. |
| **Flujos Alternos:** | **Flujos alternos**  **Cancelar dimensiones.**  3b. El usuario desea cancelar la creación de la sección mediante el botón <Cancelar>.  4b. El sistema regresa al flujo normal 1a.  **Cancelar puntos de referencia**  5c. El usuario desea cancelar la creación de la sección mediante el botón <Cancelar>.  6c. El sistema regresa al flujo normal 1a. |
| **Excepciones:** | 3d. El usuario indica valores numéricos no válidos: No naturales u otros caracteres.  4d. El sistema muestra un mensaje de advertencia indicando el problema. |
| **Poscondiciones:** | **POST01-**  El usuario finaliza el diseño del perfil  **POST02-** El usuario decide agregar otra sección. |
| **Prioridad:** | Indispensable. |

### Crear perfil estándar

Tabla 4 Tabla de caso de uso Crear perfil estándar.

|  |  |
| --- | --- |
| **ID:** | CU\_04 |
| **Nombre:** | Crear perfil estándar. |
| **Autor(es):** | Mares Solano Francisco Manuel |
| **Fecha de creación:** | 12/08/2017 |
| **Fecha de actualización:** |  |
| **Actor(es):** | Usuario |
| **Descripción:** | Este caso de uso inicia cuando el usuario desea crear un perfil estándar escogiendo del repositorio, el modelo de viga con las características de su preferencia. |
| **Precondiciones:** | **PRE01-** haber decidido escoger un perfil estándar. |
| **Flujo Normal:** | **Flujo normal**  1a. El usuario selecciona un perfil de viga del menú y lo indica con la opción <Escoger>.  2a. El sistema pregunta si desea escoger el modelo previamente seleccionado.  3a. El usuario lo confirma con la opción <Aceptar>.  4a. El sistema lo lleva a la página de diseño del sistema. |
| **Flujos Alternos:** | **Flujos alternos**  **Cancelar dimensiones.**  3b. El usuario desea cancelar la creación de la sección mediante el botón <Cancelar>.  4b. El sistema regresa al flujo normal 1a. |
| **Excepciones:** |  |
| **Poscondiciones:** | **POST01-**  El usuario decide agregar otra sección con un nuevo perfil.  **POST02-** El usuario proporciona las condiciones de la viga. |
| **Prioridad:** | Indispensable. |

## Diagramas de robustez de cada CU

Para cada uno de los Casos de usos presentados en el apartado *Diagrama de Casos de Uso,* corresponde un diagrama de robustez el cual defina el flujo de trabajo apoyado de las clases detectadas en el apartado *Diagrama de clases*. A continuación, estos diagramas son presentados

### Escoger tipo de análisis



### Agregar datos generales



### Agregar perfil



### Agregar módulo



### Finalizar cálculo



### Realizar otro análisis



## Diagramas de secuencia de cada CU

### Escoger tipo de análisis

### Agregar datos generales

### Agregar Perfil

### Agregar Módulo

### Finalizar cálculo

### Realizar otro análisis

# Diagrama de clases



# Plan de pruebas