



# **Pontificia Universidad Católica de Valparaíso**

Facultad de Ingeniería

Escuela de Ingeniería Informática

## **REPROGRAMACION BALANCEADA DE LA CARGA ACADEMICA PARA ALUMNOS CON SANCION DE ARTICULO N° 28.**

Autor:

Sebastián Andrés Cordero Barrientos - Cristoffer Eduardo Morales López

Informe final del Proyecto para optar al Título profesional de

Ingeniero de Ejecución en Informática

Profesor guía:

Jorge Bozo Parraguez

Profesor Co-referente:

Broderick Crawford Labrín

**Diciembre - 2006**

### **Resumen**

En este documento se describe el problema que se genera al aplicar el Artículo 28 del Reglamento General de Estudios, debido al gran número de variables y restricciones que se

deben tener en cuenta para lograr una reprogramación de las asignaturas por cursar, balanceadas y apropiadas para los alumnos que incurrir en esta sanción.

Para la solución de este problema, se investigó, estudió, adaptó e implementó una solución informática, un modelo de programación lineal con la ayuda de librerías, de un software especializado en este tipo de problemas, disponibles para solucionar este tipo de modelos.

## Abstract

This document, describe the issues generated by the application of the 28 Article of the General Studies Regulations, due to the big number of variables and restrictions that must be considered if a balanced and appropriated reprogramming of the pending courses wants to be done, to those students that felt in this sanction.

For the resolution of this problem, an investigation, study, adaptation and implementation by means of a prototype, were made of a linear programming model with the help of libraries, of specialized software in this kind of problems, available to solve this type of models.

# Capítulo 1. Introducción

En la Universidad existen reglamentos que se deben cumplir con respecto a la cantidad de asignaturas que un alumno debe tener aprobadas a partir de un cierto período, la cantidad de créditos por semestre, notas mínimas de aprobación, etc. Cuando éstas no se cumplen, dan origen al causal de eliminación descritos por ciertos artículos en el Reglamento General de Estudios.

El Artículo 28 en particular, indica la cantidad mínima de créditos exigidos, para un determinado período. En dicho artículo se ordena a cualquier alumno que incurre en éste, a reformular o reprogramar su carga académica semestral, paso que generalmente es un proceso resuelto mediante prueba y error, es decir, observar su avance en la malla (asignaturas aprobadas, reprobadas, no cursadas) y probar distintas combinaciones u ordenaciones de asignaturas con el fin de que esta nueva carga académica sea capaz de poner al día al alumno en cuanto a sus créditos y a la vez, que éste sea capaz de cumplir con dicha malla curricular.

El mayor reto en este problema es mantener un balanceo de la carga académica utilizando como punto de referencia la cantidad de asignaturas y de créditos por semestre. Para este balanceo se necesita un acercamiento más cuantitativo que la anteriormente nombrada técnica de prueba y error.

En este documento se define detalladamente éste problema y se realiza un diseño e implementación al método de solución de esta problemática.

En un principio se investigó sobre la programación lineal para la resolución del problema, ya que se basa en un modelo combinatorial y de asignación, para lo que la programación lineal es ideal para estos casos. Este proceso tuvo como resultado un modelo de programación lineal para la reprogramación de la carga académica.

Cabe mencionar que los ejemplos contenidos en este documento se basan en la malla de la carrera de Ingeniería Ejecución en Informática de la Pontificia Universidad Católica de

Valparaíso lo cual no implica que no sea ajustable a otras mallas curriculares de distintas carreras.

## Capítulo 2. Definición de Objetivos

### 2.1 Objetivo General

Implementar, en una aplicación informática, un modelo de programación lineal entero binario, que de manera balanceada reformule la carga académica para los alumnos que tengan sanción por el Artículo 28, del Reglamento General de Estudios de la Pontificia Universidad Católica de Valparaíso .

### 2.2 Objetivos Específicos

Estructurar un modelo de programación lineal completo, capaz de resolver el problema de reformulación de la carga académica.

Diseñar e implementar una solución de manera de obtener los datos del alumno que influyan y sean parte del problema desde una fuente fiable, tales como: asignaturas aprobadas, reprobadas, pendientes, entre otros.

Alivianar el trabajo realizado por los Jefes de Docencia al momento de reformular la carga académica para cada alumno que haya incurrido en el Artículo 28.

## Capítulo 3. Descripción de la Problemática y su posible Solución.

### 3.1 Situación Actual

En cada Jefatura de Docencia de distintas Unidades Académicas, existe un proceso en el cual a cada término de período académico se analizan los casos de los alumnos que incurren en el Artículo 28, establecido en el Reglamento General de Estudios dispuesto por la Vice-Rectoría de Asuntos Docentes y Estudiantiles. Este artículo rige a contar de la mitad mas uno de los semestre que componen el currículo de cada alumno, donde en dicho período se le pide al alumno una cantidad de créditos mínimo aprobados la cual se calcula como la suma de los promedios de créditos aprobados por semestre el que corresponderá al cuociente entre el total de créditos del Plan de Estudios y el número máximo de períodos académicos a que se refiere la letra (d) del Artículo 22° .

#### Ejemplo:

Carrera:	Ingeniería ejecución en informática
Numero de semestres que componen el currículo:	8
Número máximo de períodos académicos:	13
Total de créditos del Plan de Estudios:	145

**Entonces se tiene:**

El Artículo 28 regirá desde el quinto semestre:

$$((8 \div 2) + 1) = 5$$

Promedio mínimo de créditos aprobados por semestre:

$$\frac{\text{Total de créditos del Plan de Estudio}}{\text{Número máximo de períodos académicos}} = \frac{145}{13} = 11$$

Por lo tanto al quinto semestre el alumno tendrá que contar con 44 créditos aprobados.

En el caso que el alumno incurra en dicho artículo, existe la posibilidad que se le reformule su carga académica, solicitando la eximición del artículo. Esta labor consta de procesos no triviales para cada caso ya que cada alumno va a contar con una situación distinta a la de otro alumno, esto va a depender de cuántas y cuáles asignaturas tiene aprobadas, reprobadas o no cursadas, ya que éstas se tomarán como primer punto de consideración para reprogramar la carga académica de cada alumno.

Otro punto a considerar, es la cantidad de créditos en déficit con que se presenta el alumno para que éstos sean distribuidos de la manera más equitativa posible dentro de los períodos restantes de la carrera para que no exista una sobrecarga de asignaturas en un sólo período.

### **3.2 Problemas Identificados**

Como se logra apreciar, en la manera actual de solucionar el problema, se tienen que considerar bastantes puntos para resolver el problema, y aunque si solo se considerase el caso de los prerrequisitos, se vuelve realmente enredado y complicado llegar a una solución en un tiempo considerablemente corto ya que se basa en “prueba y error”.

Otro problema observado es que al encontrar una solución posible se puede generar un período en el que existe una sobrecarga académica, con lo que se tiene que realizar todo el proceso nuevamente.

Sobre el punto anterior, cabe mencionar que el alumno, al momento de hacer la solicitud de eximición del artículo N° 28, cae en un error común, al pensar en sobrecargar algunos

períodos, y así poder quedar al día en el avance curricular, lo que lleva a la gran mayoría de los casos, a una carga irreal y exagerada para poder enfrentar el período.

### **3.3 Situación Futura**

Al existir un caso de que un alumno incurra en el Artículo N° 28, éste solicita la eximición de dicho Artículo a través del Jefe de Docencia, quien hará la reprogramación de su carga académica. El Jefe de Docencia por medio del sistema en cuestión, le podrá entregar al alumno de forma automática un resultado factible y realista.

En el caso que el resultado no sea de la conveniencia del alumno, dado que pretende solicitar el levantamiento de prerrequisitos, en donde una asignatura puede ser cursada sin tener aprobada la que le precede, el sistema será capaz de realizar movimientos de asignaturas en la malla académica hasta que el alumno esté conforme con el resultado entregado.

El alumno también podrá realizar dicha operación en cualquier computador que este habilitado para el sistema, pudiendo analizar previamente su situación antes de la visita con su Jefe de Docencia, para plantear su eximición del Artículo 28.

Incluyendo lo dicho anteriormente y que debido a que todos los cálculos de restricciones serán automáticamente realizados por el sistema, habrá una disminución en el tiempo de realización de esta labor permitiendo disminuir el recurso horas/hombre y alivianar la labor del Jefe de Docencia.

### **3.4 Análisis de Requerimientos**

La solución deberá cumplir con las condiciones reglamentarias comunes de las mallas curriculares como:

<p>Se deberá respetar el orden de las asignaturas y sus prerrequisitos, es decir, no se podrá asignar una asignatura en un semestre previo a su antecesora, entendiéndose por ésta, la asignatura prerrequisito, y respetar asignaturas que se deben cursar en cierto semestre, por ejemplo, anuales sólo del primer semestre o del segundo.</p>
--



Deberá cumplir con restricciones sobre la distribución o balanceo de la carga académica, basándose en el número de asignaturas y el número de créditos por semestre que sería indicado según el usuario.

Se deberá tomar en cuenta la duración máxima de la carrera para entregar los resultados, es decir, el número máximo de períodos en el que se debe completar el currículo.

Sobre el ingreso de los datos: La entrada será, además de las asignaturas de una carrera en particular, las características de la malla: prerrequisitos, asignaturas ya aprobadas, el número del semestre actual del alumno y la cantidad de semestres en que se realizará la reprogramación. Se deberán indicar valores límites para las restricciones tales como número máximo y mínimo de créditos o de asignaturas por semestre.

Sobre el balanceo de la malla: Para el balanceo de la malla curricular se deberá considerar la cantidad de créditos y la cantidad de asignaturas por semestre.

Sobre la solución del modelo: La solución debe ser presentada de manera tal que sea entendible e intuitiva, no debe requerir mayor análisis por parte del usuario, ya que este tipo de problemas suelen entregar soluciones complejas y que requieren mayor análisis que el problema en si.

## Capítulo 4. Programación lineal

La Programación Lineal es una técnica determinista, no incluye probabilidades y utiliza un modelo matemático para describir el problema. El adjetivo lineal significa que todas las funciones matemáticas del modelo deben ser funciones lineales. En este caso, la palabra programación no se refiere a programación en computadoras; en esencia, es un sinónimo de planeación. Así, la programación lineal trata la planificación de las actividades para obtener un resultado óptimo, esto es, el resultado que mejor alcance la meta especificada (según el modelo) entre todas las opciones de solución.

Modelar un problema usando programación lineal, significa describir este problema mediante ecuaciones y variables, las que al tomar cierto valor, hacen posible encontrar una solución a dicho problema. Para ello existen reglas de lo que se puede y no hacer con programación lineal. Esas reglas tratan de asegurar que los siguientes pasos en la resolución del problema (resolver e interpretar) lleguen a buen término.

Las partes claves de la programación lineal son: determinar las variables de decisión, el objetivo y las restricciones.

**Variables de Decisión:** Las variables de decisión representan decisiones a tomar, que pueden corresponder, como en este caso, a asignaciones, es decir, si una variable es o no asignada dependiendo del valor que ésta tenga.

**Objetivo:** Cada problema de programación lineal tiene un objetivo asociado. Ese objetivo debe ser maximizado o minimizado.

**Restricciones:** Cada problema de programación lineal tiene también un conjunto de restricciones que limitan los valores que pueden tomar las variables.

En ciertas ocasiones, solo son coherentes aquellos resultados en los que las variables solo son cifras enteras, en este caso se tiene un problema de programación lineal entera, en otros casos, los valores que pueden tomar las variables son aún más específicos, pudiendo solo tomar los

valores 1 ó 0, donde en la mayoría de los casos estos valores se usan para variables de decisión donde un “1” significa que la decisión se toma y un “0”, que no, estas variables son llamadas binarias.

Al tener variables enteras y/o binarias en un modelo de programación lineal, el número de soluciones se reduce a una cantidad finita, pero que en muchos casos, el número de soluciones a analizar puede ser excesivo.

Llevando esto al caso de la carga académica es fácil darse cuenta que la función objetivo para esta problemática es minimizar la carga académica por período, en este caso, no se puede asignar 16,5 créditos, por lo anterior el método de resolución para la problemática se basará en un modelo de programación lineal entera, y si se considera las variables de decisión, en las que una asignatura puede (1) ó no (0) ser asignada, se puede agregar el adjetivo binario, quedando con un problema de programación lineal entera binaria.

## **4.1 Resolución de Programación Lineal**

Para la resolución del problema, se basará en una de las técnicas elaboradas bajo la programación lineal entera, la cual tiene como objetivo diseñar un modelo de programación lineal para la creación de mallas curriculares balanceadas, dicho modelo ha sido desarrollado por el Dr. Carlos Castro.

Dado que este método es utilizado para generar mallas completas, éste será modificado y adaptado a ésta problemática.

Cabe destacar que cuando se hace mención a carga académica se refiere a la suma de los créditos de cada asignatura de un período. El modelo de la programación lineal para el balanceo de mallas curriculares se define en este capítulo.

## 4.2 Parámetros del Modelo

$m$  : número de asignaturas de la malla

$n$  : número de períodos académicos

$\alpha_i$  : número de créditos del curso  $i$ ;  $i = 1, \dots, m$

$\beta$  : carga académica mínima aceptada por período (créditos)

$\gamma$  : carga académica máxima permitida por período (créditos)

$\delta$  : mínimo número de asignaturas por período

$\varepsilon$  : máximo número de asignaturas por período

## 4.3 Variables de Decisión

$c_j$  : carga académica para el período  $j$  (créditos)

$c$  : máxima carga académica para los períodos (créditos)

$x_{ij} = \begin{cases} 1 & \text{si el curso } i \text{ es asignado al período } j; i = 1, \dots, m \wedge j = 1, \dots, n \\ 0 & \text{en otro caso.} \end{cases}$

## 4.4 Función Objetivo

El objetivo es minimizar la carga académica, por lo tanto la función objetivo se define como:

$$\text{Min } c$$

La carga máxima esta definida por:

$$c = \text{Max } \{c_1, \dots, c_n\} \quad \text{Ecuación 1}$$

Esto puede ser representado por el siguiente conjunto de restricciones lineales:

$$c_j \leq c \quad \forall j = 1, \dots, n \quad \text{Ecuación 2}$$

La carga académica para el período  $j$  es definida por:

$$c_j = \sum_{i=1}^m \alpha_i \times x_{ij} \quad \forall j = 1, \dots, n \quad \text{Ecuación 3}$$

## 4.5 Restricciones

Todas las asignaturas “*i*” deben ser asignadas a un período “*j*”:

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall i = 1, \dots, m \quad \text{Ecuación 4}$$

Asignatura “*b*” tiene a “*a*” como prerrequisito:

$$x_{bj} \leq \sum_{r=1}^{j-1} x_{ar} \quad \forall j = 2, \dots, n \quad \text{Ecuación 5}$$

La carga académica del período *j* debe ser mayor o igual que el mínimo requerido:

$$c_j \geq \beta \quad \forall j = 1, \dots, n \quad \text{Ecuación 6}$$

La carga académica del período *j* debe ser menor o igual que el máximo permitido:

$$c_j \leq \gamma \quad \forall j = 1, \dots, n \quad \text{Ecuación 7}$$

El número de asignaturas del período *j* debe ser mayor o igual al mínimo permitido:

$$\sum_{i=1}^m x_{ij} \geq \delta \quad \forall j = 1, \dots, n \quad \text{Ecuación 8}$$

El número de asignaturas del período *j* debe ser menor o igual que el máximo permitido:

$$\sum_{i=1}^m x_{ij} \leq \varepsilon \quad \forall j = 1, \dots, n \quad \text{Ecuación 9}$$

## 4.6 Adaptación del Modelo de programación Lineal

Hasta aquí estas restricciones propias del modelo encontrado para el diseño de un Currículo Académico, para adecuar el modelo a la problemática, se han agregado ciertas restricciones, que cumplirán con el objetivo de reformular la carga académica y que facilitarán su implementación.

Estas modificaciones son:

Siendo “s”, el semestre actual del alumno, la definición de “c”, ahora es:

$$c = \text{Max} \{c_s, \dots, c_n\} \quad \text{Ecuación 10}$$

En las restricciones debemos agregar:

Para “a” una asignatura no aprobada, y “s” el semestre actual del alumno

$$x_{aj} = 0 \quad \forall j = 1, \dots, s \quad \text{Ecuación 11}$$

Esta restricción cumple con el objetivo de tomar en cuenta las asignaturas que el alumno no ha aprobado en semestres anteriores a contar del punto en donde desea realizar la reprogramación de su carga académica, para que estos sean asignados en los futuros períodos.

Para “b” una asignatura aprobada en el semestre “k”

$$x_{bk} = 1 \quad \text{Ecuación 12}$$

Esta restricción le dice al modelo qué asignaturas ya ha aprobado el alumno en semestres posteriores al período en donde realiza la reprogramación.

Para “a” una asignatura reprobada en semestres anteriores a “s”:

$$x_{as} + x_{as+1} = 1 \quad \text{Ecuación 13}$$

Esta restricción obliga a tomar asignaturas reprobadas en los dos primeros semestres después de la reprogramación, tal como se indica en .

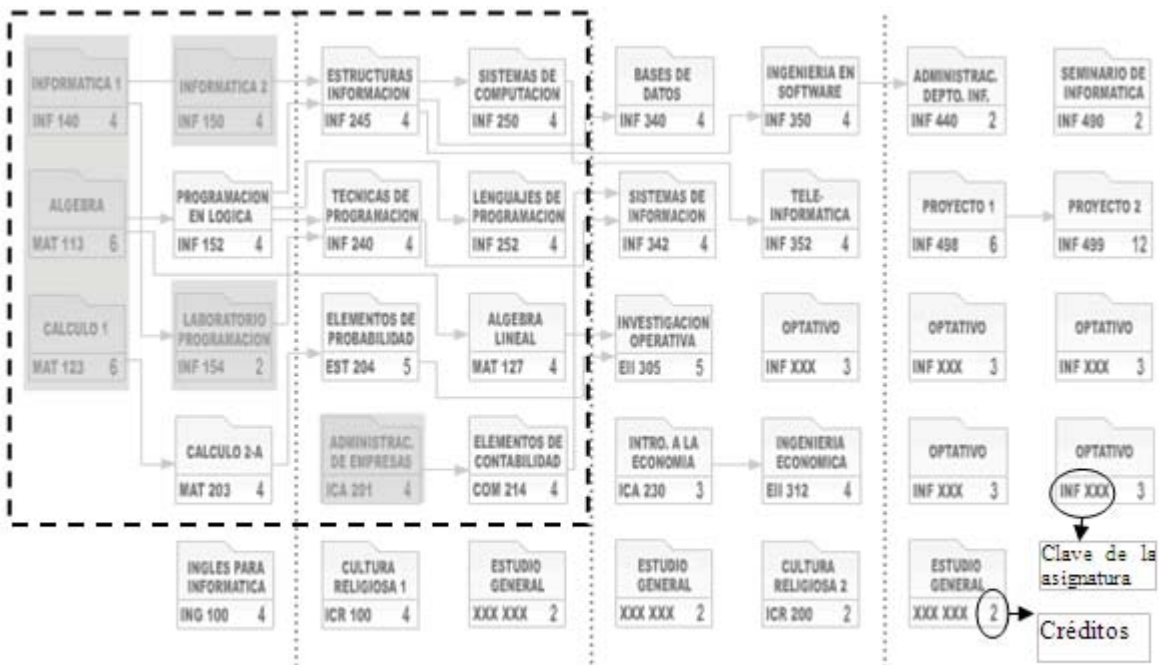
Cabe mencionar que “s”, el número de períodos académicos a reprogramar, deberá ser ingresado por el usuario al momento de hacer la reprogramación, siendo “s” restringida por el valor de “n”.

## 4.7 Ejemplo de resolución

Como ejemplo usaremos una versión de los 4 primeros semestres de la malla de la carrera de Ingeniería de Ejecución Informática de la Pontificia Universidad Católica de Valparaíso.

Para el caso del ejemplo no consideraremos, por no tener relevancia importante, los prerrequisitos o ser prerrequisito de alguna otro asignatura, en el caso de: Ingles para Informática, Cultura Religiosa y un Estudio General correspondiente a estos 4 períodos, además, de no considerar asignaturas anuales, es decir aquellas impartidas solo en semestres que corresponda.

Por lo tanto solo consideraremos las asignaturas dentro del recuadro, las que se impartirán en cualquier semestre.



■ = Asignaturas aprobadas.

### Ilustración 1: Malla Ing. De Ejecución en Informática

Para nuestro ejemplo, consideraremos a un alumno que haya aprobado 6 de las 15 asignaturas de esta malla: Informática 1, Álgebra, Cálculo, Informática 2, Laboratorio de Programación y Administración de Empresas.

Como nos podemos dar cuenta, el alumno debería estar cursando el tercer semestre de la carrera como mínimo, ya que aprobó Administración de Empresas. Quedando así, con 9 asignaturas y 37 créditos, los cuales repartiremos en 3 semestres.

Respetando sólo el orden de los prerrequisitos, en el cuarto semestre, se tomarían 16 créditos: Programación en Lógica, Cálculo 2-A, Álgebra Lineal y Elementos de Contabilidad. En el quinto semestre, 17: Estructuras Información, Lenguajes de Programación y Elementos de Probabilidad. Y al sexto semestre solo 4: Sistemas de Computación.

Entonces resumiendo el cuarto semestre, se tomarían 16 créditos, en el quinto semestre, 17, y al sexto semestre solo 4, todo esto respetando solo la condición de que ciertas asignaturas son prerrequisitos de otros.

4°	5°	6°
PROGRAMACION EN LOGICA INF 152 4	ESTRUCTURAS INFORMACION INF 245 4	SISTEMAS DE COMPUTACION INF 250 4
CALCULO 2-A MAT 203 4	TECNICAS DE PROGRAMACION INF 240 4	
ELEMENTOS DE CONTABILIDAD COM 214 4	LENGUAJES DE PROGRAMACION INF 252 4	
ALGEBRA LINEAL MAT 127 4	ELEMENTOS DE PROBABILIDAD EST 204 5	

### Ilustración 2: Ejemplo de Reprogramación No Balanceada

El problema de esta solución, es que no distribuye la carga académica de manera equitativa, es decir una sobrecarga de créditos y asignaturas innecesarios en el cuarto y quinto semestre que al fin pueden llevar a un fracaso por parte del alumno retrasándolo aún más, debemos



considerar el hecho de que esta malla posee solo 15 asignaturas, y la malla normal de la carrera 37, haciendo que una solución que cumpla solo con la condición de prerrequisitos sea difícil de encontrar de manera manual. En cuanto al balanceo en este caso se puede lograr de manera sencilla moviendo un par de asignaturas al siguiente semestre, pero esto se debe a que esta malla se encuentra reducida, lo que no es tan fácil de visualizar con una malla completa.



Ilustración 3: Ejemplo de Reprogramación Balanceada

Esta solución, consta de los mismos 6 semestres (3 de reprogramación), pero la diferencia se encuentra en la distribución equitativa de la cantidad de créditos y el número de asignaturas por semestre, logrando una cantidad cercana a 12 créditos y 3 asignaturas en cada semestre, respetando el orden de éstos y evitando una sobrecarga en el alumno.

## 4.8 Comprobación del Resultado

La tabla que se muestra a continuación, indica la relación entre asignaturas ( $i$ ) y sus prerrequisitos.

$i$	Nombre del Asignatura	$\alpha_i$	Prerrequisito 1 (Asignatura $i$ )	Prerrequisito 2 (Asignatura $i$ )
1	Informática 1	4	-	-

2	Algebra	6	-	-
3	Calculo	6	-	-
4	Informática 2	4	1	-
5	Programación en Lógica	4	2	-
6	Laboratorio de programación	2	1	-
7	Calculo 2-A	4	3	-
8	Estructuras Información	4	4	5
9	Técnicas de Programación	4	5	6
10	Elementos de Probabilidad	5	7	-
11	Administración de Empresas	4	-	-
12	Sistemas de Computación	4	8	-
13	Lenguajes de Programación	4	5	-
14	Algebra Lineal	4	2	-
15	Elementos de Contabilidad	4	11	-

A continuación se indican los valores que toman los parámetros del modelo para este caso:

$m$	: número de asignaturas de la malla	=15
$n$	: número de períodos académicos	=6
$\beta$	: carga académica mínima aceptada por período (créditos)	=3
$\gamma$	: carga académica máxima permitida por período (créditos)	=16
$\delta$	: mínimo número de asignaturas por período	=1
$\varepsilon$	: máximo número de asignaturas por período	=7

En el siguiente recuadro, se tiene una tabla de ancho 15 y largo 6, que representan las 15 asignaturas en 6 semestres, una casilla representa el valor de asignación (1 si es asignado y 0 si no lo es) de la asignatura en el semestre correspondiente es decir; (1,1)=1 significa que la asignatura 1 fue programada para el semestre 1, así como (2,1)=1, indica que la asignatura 2 lo fue para el semestre 1, (1,2)=0 la asignatura 1 no fue programada para el semestre 2, etc.

El semestre actual es el 3 ( $s=3$ ), por lo que las restricciones 11 y 12 ya se han cumplido, asignando 0 a las asignaturas no aprobadas que pertenecen a estos 3 períodos y 1 a los ya aprobados.

		Asignaturas															Total	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
Semestres	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
	2	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-
	3	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	-
	4	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	3
	5	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	3
	6	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	3
			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

La columna total representa la suma o cantidad de asignaturas programadas en el semestre correspondiente, mientras que la última fila representa el número de veces que fue asignada la asignatura correspondiente. Con este valor podemos darnos cuenta que se cumple con las restricciones 8 y 9.

Los primeros tres semestres no tienen valor ya que estos semestres son previos a la reprogramación de la malla. Las asignaturas con valores 1 indican que éstos han sido aprobados en el semestre correspondiente.

La última fila representa la cantidad de veces que ha sido asignada la asignatura en los distintos períodos, de esta manera, se cumple con la restricción 4.

La siguiente tabla muestra la cantidad de créditos asignados en cada período.

		Asignaturas															Total
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
Semestres	1	4	6	0	0	0	0	0	0	0	0	0	0	0	0	0	-
	2	0	0	6	4	0	0	0	0	0	0	0	0	0	0	0	-
	3	0	0	0	0	0	2	0	0	0	0	4	0	0	0	0	-
	4	0	0	0	0	4	0	4	0	0	0	0	0	0	4	0	12
	5	0	0	0	0	0	0	0	4	0	0	0	0	4	0	4	12
	6	0	0	0	0	0	0	0	0	4	5	0	4	0	0	0	13

En este caso se muestra la carga académica de cada asignatura y semestre, el número de la última columna, indica la carga total asignada en ese período, como podemos ver, la carga es bastante similar en los tres semestres de reprogramación, de esta manera se puede comprobar que se cumplen con las restricciones 6 y 7.

Para la restricción 5 tomemos, por ejemplo, la asignatura Estructuras Información ( $i=8$ ). Esta, posee 2 prerrequisitos, Informática 2 ( $i=4$ ) y Programación en Lógica ( $i=5$ ).

En el caso de Estructuras de Información ( $i=8$ ) con Informática 2 ( $i=4$ ):

$$x_{82} \leq x_{41} \rightarrow 0 \leq 0$$

$$x_{83} \leq x_{41} + x_{42} \rightarrow 0 \leq 0 + 1$$

$$x_{84} \leq x_{41} + x_{42} + x_{43} \rightarrow 0 \leq 0 + 1 + 0$$

**Ecuación 14**

$$x_{85} \leq x_{41} + x_{42} + x_{43} + x_{44} \rightarrow 1 \leq 0 + 1 + 0 + 0$$

$$x_{86} \leq x_{41} + x_{42} + x_{43} + x_{44} + x_{45} \rightarrow 0 \leq 0 + 1 + 0 + 0 + 0$$

$$x_{82} \leq x_{51} \rightarrow 0 \leq 0$$

$$x_{83} \leq x_{51} + x_{52} \rightarrow 0 \leq 0 + 0$$

$$x_{84} \leq x_{51} + x_{52} + x_{53} \rightarrow 0 \leq 0 + 0 + 0$$

**Ecuación 15**

$$x_{85} \leq x_{51} + x_{52} + x_{53} + x_{54} \rightarrow 1 \leq 0 + 0 + 0 + 1$$

$$x_{86} \leq x_{51} + x_{52} + x_{53} + x_{54} + x_{55} \rightarrow 0 \leq 0 + 0 + 0 + 1 + 0$$

Esta restricción verifica si una asignatura prerequisite ha sido programada en cualquier semestre anterior al de la asignatura que solicita el prerequisite.

En este caso, Estructuras Información posee 2 prerequisites, que en el caso de Ingeniería Ejecución Informática es una cantidad común, puede ser que en otras mallas existan asignaturas con más de 2 prerequisites, lo cual complica demasiado obtener un resultado de manera manual.

Esta misma restricción debe aplicarse a cada asignatura que posea prerequisites.

Hasta aquí se ha descrito, en términos de la programación lineal entera binaria, el modelado del problema y como se puede comprobar el resultado. Desde aquí en adelante el problema será abarcado desde el punto de vista de la implementación informática del problema.

## Capítulo 5. Paradigma de Desarrollo

En proyectos informáticos, contar con un paradigma de desarrollo, es fundamental para mantener un orden y control sobre el trabajo, por esto, la elección de un paradigma se debe tener en mente desde un principio.

En este proyecto se optó por una corriente relativamente nueva llamada Métodos Ágiles, dentro de la cuál, un paradigma emergente es Feature Driven Development (FDD), las razones para usar FDD son:

Está pensado para proyectos cortos, establece etapas iterativas de poca duración (2 semanas).

Al ser de características ágiles, necesita un equipo de trabajo pequeño.

Provee una continúa evolución del proyecto.

Si es bien implementado, garantiza funcionalidad en pocas semanas.

Uno de los puntos débiles de FDD y que se debe considerar desde un principio es que ésta no define una etapa de obtención de requerimientos, deja abierta la posibilidad de usar casos de uso y usar Rational Unified Process (RUP) en esta etapa del proyecto.

La principal característica de FDD son los rasgos o funcionalidades (“features”), éstos representan pequeñas partes funcionales del proyecto y de interés del usuario, una vez identificados los rasgos se organizan por la importancia y funcionalidad que aportan, por lo tanto los primeros rasgos desarrollados otorgan la funcionalidad básica del sistema, garantizando así, su funcionalidad.

Se divide en 5 etapas, dos de las cuales son iterativas.

Desarrollo de un modelo general.

Construcción de la lista de rasgos.

Planeación por rasgos.

Diseño por rasgos (iterativo).

Construcción por rasgo (iterativo).

Las tres primeras etapas se realizan una sola vez dentro de un proyecto, las últimas dos son iterativas por cada rasgo identificado, después de cada iteración se debe actualizar el diseño del sistema y la documentación.

La implementación, pruebas por unidad, revisión del código e integración son realizados por cada iteración.

## Capítulo 6. Metodología de Trabajo

Este proyecto consta de 2 grandes partes, la primera, tiene que ver con la investigación necesaria para encontrar el modelo de programación lineal capaz de entregar una solución que satisfaga las necesidades, y una segunda etapa, donde se debe desarrollar una aplicación que implemente el modelo y pueda entregar una solución entendible por el usuario.

### **6.1 Metodología de Investigación**

Para la realización de un proyecto de estas características, orientado más a una investigación que a un desarrollo neto se necesita una manera distinta de realizar el proyecto, por lo tanto se enfrentará al problema dividiéndolo en las siguientes etapas.

#### **6.1.1 Búsqueda de documentación**

En esta etapa se corresponde a la búsqueda de documentación para enfrentar el problema, encontrando como documentación, métodos de resolución a problemas semejantes.

#### **6.1.2 Determinación del Problema**

Se comenzó a comprender el problema determinando y definiendo claramente el problema de la reprogramación de una carga académica según el artículo N° 28, para lo cual se sostuvo reuniones esporádicas con el Jefe de Docencia de la Escuela de Ingeniería Informática, en donde se aclararon las dudas y se captaron los requerimientos para el sistema.

#### **6.1.3 Definición del método de trabajo**

Esto se explicó en el capítulo 5 donde se muestra los recursos a utilizar para llevar a cabo el proyecto



### **6.1.4 Resolución, validación y verificación**

La búsqueda de una solución al problema se dividirá en resolución, validación y verificación; las cuales estarán estrechamente relacionadas con el proceso de diseño del prototipo.

Cada una de estas etapas será como sigue:

Resolución: se pondrá en marcha el método encontrado, el que se acomodará a este proyecto.

Verificación: a través de un diseño de prototipo, el cual entregará un resultado, estos datos serán analizados y comparados con los resultados esperados.

Validación: se tomarán y comprobarán los resultados obtenidos de la verificación.

## **6.2 Metodología de Desarrollo**

La Orientación a Objeto cuenta con una gran cantidad de ventajas como las que se muestran a continuación:

Reutilización. Las clases están diseñadas para que se reutilicen en muchos sistemas. Para maximizar la reutilización, las clases se construyen de manera que se puedan adaptar a los otros sistemas. Un objetivo fundamental de las técnicas orientadas a objetos es lograr la reutilización masiva al construir el software.

Estabilidad. Las clases diseñadas para una reutilización repetida se vuelven estables, de la misma manera que los microprocesadores y otros chips se hacen estables.

El encapsulamiento oculta los detalles y hace que las clases complejas sean fáciles de utilizar.

Se construyen clases a partir de otras clases, las cuales a su vez se integran mediante clases. Esto permite construir componentes de software complejos, que a su vez se convierten en bloques de construcción de software más complejo.

Calidad. Los diseños suelen tener mayor calidad, puesto que se integran a partir de componentes probados, que han sido verificados y pulidos varias veces.

Un diseño más rápido. Las aplicaciones se crean a partir de componentes ya existentes. Muchos de los componentes están contruidos de modo que se pueden adaptar para un diseño particular.

Integridad. Las estructuras de datos (los objetos) sólo se pueden utilizar con métodos específicos.

Mantenimiento más sencillo. Cada clase efectúa sus funciones independientemente de las demás.

Independencia del diseño. Las clases están diseñadas para ser independientes del ambiente de plataformas, hardware y software. Utilizan solicitudes y respuestas con formato estándar. Esto les permite ser utilizadas en múltiples sistemas operativos, controladores de bases de datos, controladores de red, interfaces de usuario gráficas, etc. El creador del software no tiene que preocuparse por el ambiente o esperar a que éste se especifique.

Interacción. El software de varios proveedores puede funcionar como conjunto. Un proveedor utiliza clases de otros. Existe una forma estándar de localizar clases e interactuar con ellas. El software desarrollado de manera independiente en lugares ajenos debe poder funcionar en forma conjunta y aparecer como una sola unidad ante el usuario.

Mejores herramientas CASE. Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora) utilizarán las técnicas gráficas para el diseño de las clases y de la interacción entre ellas, para el uso de los objetos existentes adaptados a nuevas aplicaciones.

A lo mencionado anteriormente cabe mencionar que una metodología es un conjunto de procedimientos, técnicas, herramientas y soporte documental para la realización de nuevo software. Con lo último dicho UML es un potente lenguaje que ayuda a la documentación del proceso de desarrollo con lo que le da otro número de beneficios:

Mejores tiempos totales de desarrollo (de 50 % o más).

Modelar sistemas utilizando conceptos orientados a objetos.

Establecer conceptos y artefactos ejecutables.

Encaminar el desarrollo del escalamiento en sistemas complejos de misión crítica.

Crear un lenguaje de modelado utilizado tanto por humanos como por máquinas.

Mejor soporte a la planeación y al control de proyectos.

Alta reutilización y minimización de costos.

Con todo lo nombrado anteriormente la elección de metodología para este proyecto es  
Orientación a Objetos

## Capítulo 7. Herramientas y Tecnologías de Desarrollo

Para la realización de este proyecto, son necesarias varias herramientas que ayudan a la documentación y desarrollo del mismo.

Las herramientas empleadas para la documentación son:

Microsoft Word 2003

Microsoft Power Point 2003

Microsoft Visio 2003

Microsoft Project 2003

Para el desarrollo e implementación del proyecto se necesita Microsoft Visual Studio .NET 2003 con el lenguaje de programación C# de esta misma plataforma. Otra herramienta necesaria es LP\_SOLVE, el que ayudara a resolver el modelo de programación lineal, y se detalla a continuación.

La elección de la plataforma .NET se debe a que proporciona gran facilidad al momento de crear aplicación con formularios y manejo de archivos, también se debe a su gran similitud con C++ y el soporte que proporciona para trabajar con productos de Microsoft (manuales en línea, tutoriales, etc.) y a que se puede integrar fácilmente la librería LP\_SOLVE a una aplicación.

### 7.1 Herramientas Solve

En cuanto a la implementación de un modelo de programación lineal, existen ciertas herramientas que ayudan a su programación y resolución.

Estas herramientas se conocen como *Solvers*, existen gran variedad y de distintas características. Se pueden encontrar 2 tipos generales de herramientas *Solvers*, aquellos que ayudan de manera algorítmica, donde uno ingresa restricciones y se obtiene un valor como

resultado; y los *Solvers* de modelado que ayudan a modelar un problema de tal forma que sea entendible para el usuario y entregan un resultado, también, sencillo de entender.

El de mayor interés para el proyecto es algorítmico, ya que la mayoría de las veces son desarrolladas como librerías para ciertos lenguajes de programación.

## 7.2 Herramientas Solve Algorítmicos

La resolución del problema se basará en herramientas de código abierto, en esta categoría existen 3 proyectos bastante documentados, con sus respectivos sitios de apoyo y que ayudan bastante a la hora de resolver este tipo de modelos.

- CLP. Sitio: <http://www.coin-or.org/Clp/index.html>, en esta pagina se puede encontrar una guía de usuarios bastante completa, con ejemplos escritos en C++ .
- GLPK. Sitio: <http://directory.fsf.org/libs/glpk.html>, es un completo paquete con librerías y ejemplos escritos en C.
- LP\_SOLVE. Sitio: <http://lpsolve.sourceforge.net/5.5/>, al igual que los sitios anteriores posee manual de usuario, ejemplos de implementación y librerías escritas para casi todos los lenguajes de programación existentes .

La herramienta elegida para resolver el problema fue LP\_SOLVE, ya que soporta una variedad de lenguajes, como toda la gama de lenguajes de la plataforma .NET, además posee ejemplos para muchos de estos lenguajes.

La documentación sobre esta herramienta es bastante sencilla de comprender y en cuanto a su implementación, todas las herramientas analizadas funcionan de la misma manera, es decir poseen funciones bastantes similares.

### 7.3 Funcionamiento e Implementación de Herramientas Solve

Básicamente, estas herramientas poseen 2 maneras de ingresar un problema de programación lineal, mediante la lectura de un archivo, generalmente con extensión .MPS (soportado por la mayoría de herramientas Solve), un archivo que define una matriz bastante compleja de entender y definir, otro tipo de archivo soportado es .LP (soportado por LP\_SOLVE), que define de manera clásica un problema de programación lineal, es decir:

$$\begin{aligned} \max( & 143 x + 60 y) \\ 120 x + 210 y & \leq 15000 \\ 110 x + 30 y & \leq 4000 \\ x + y & \leq 75 \\ x & \geq 0 \\ y & \geq 0 \end{aligned} \qquad \text{Ecuación 16}$$

Lo que indica que la función objetivo es maximizar  $143x + 60y$  sujeto a las condiciones anteriores.

La otra manera de ingresar un problema es mediante funciones propias de sus librerías, la mayoría poseen funciones que agregan restricciones al modelo, de manera de un vector con el largo del número total de variables, dentro de este vector se encuentran los coeficientes de las variables para dicha restricción, tomando el ejemplo anterior, la primera restricción sería:

$$120 \quad 210$$

Las funciones permiten definir si es una restricción del tipo EQ (igual), LE (menor e igual) o GE (mayor e igual). Si existiera una tercera variable, que no posee valor para una restricción, se debe poner como coeficiente el 0.

“LP\_SOLVE” permite también modelar el problema mediante programas de lenguaje matemáticos tal y como “Mathlab”, Octave y Ampl., pero para esto es necesario saber utilizar estos programas.

Para comenzar a crear un problema de programación lineal primero se debe tener claro el número de variables que se necesita, la razón, cuando ingresemos una restricción debemos

ingresar también los coeficientes de todas las variables, incluso aquellas de valor 0, por lo tanto en el caso de una malla curricular se debe multiplicar el número de cursos por la cantidad de semestres que se está calculando, obteniendo así el mínimo número de variables.

# Capítulo 8. Análisis del Sistema

Ya investigado y analizado el modelo de programación lineal entera binaria que ayuda a resolver el problema de la reprogramación de la carga académica, llega el momento en que este modelo será aplicado de forma informática por lo que a continuación se realizara el análisis correspondiente a la aplicación.

## 8.1 Casos de uso

Como primera etapa de análisis se identifican actores y las principales funcionalidades del sistema indicando sus interacciones. Para este sistema en particular, se han reconocido las siguientes entidades:

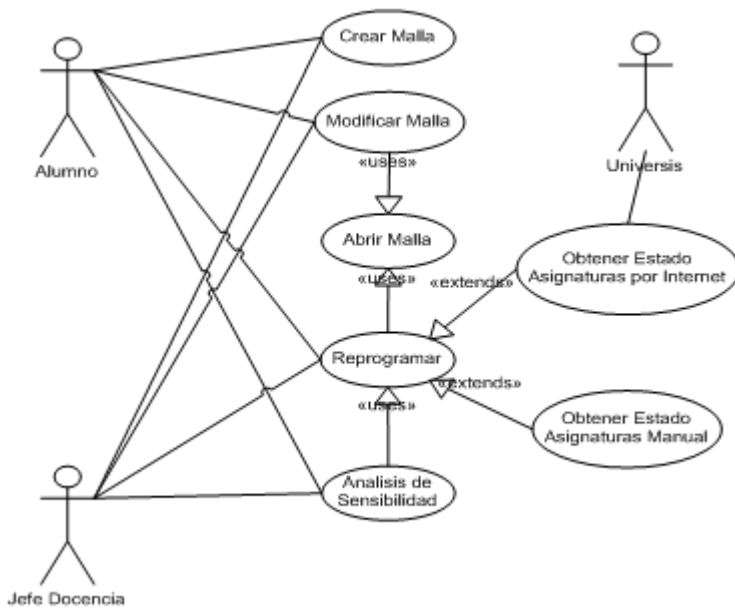


Ilustración 4: Diagrama General de Casos de Uso

Crear Malla: el usuario deberá definir las asignaturas de una malla curricular y sus características (número de créditos, nombre y sigla), las relaciones de prerrequisitos y la cantidad de créditos y asignaturas permitidas por semestre.



**Abrir Malla:** el usuario, a través de este caso de uso, indica con que malla se desea trabajar al momento de realizar una reprogramación, el sistema por lo tanto mostrará la malla seleccionada.

**Modificar Malla:** el usuario, puede realizar cambios a una malla creada, si este lo cree necesario.

**Obtener Estado Asignaturas por Internet:** En este caso de uso se capturarán las asignaturas, ya sean generales, optativas u obligatorias que el alumno ha aprobado y/o reprobado hasta el actual período. Estas asignaturas se obtendrán a través del portal Navegador Académico, por lo que se debe indicar el RUT y contraseña que se usa en dicho portal.

**Indicar Estado Asignaturas Manualmente:** En este caso de uso se indicara el estado de las asignaturas del alumno por el usuario manualmente.

**Reprogramar:** habiendo seleccionado una malla, además obtenido las asignaturas aprobadas y reprobadas del alumno, el usuario, se encuentra en condiciones de obtener una solución.

**Análisis de Sensibilidad:** luego de haberse realizado una reprogramación, la solución puede o no ser satisfactoria para el usuario, por lo que se podrían realizar cambios a la solución, por ejemplo: en el caso de que el usuario tenga problemas en dar cierta asignatura en un semestre determinado o si se planea pedir levantamiento de prerrequisitos, caso donde la aplicación no tiene información necesaria para resolver el problema, en estos casos se informará que tanto afecta que una asignatura se tome en un período o en otro, indicando las restricciones se dejan de cumplir o cuanto cambia la carga académica en ese semestre para el alumno.

## **8.2 Especificación de Casos de Uso**

### **8.2.1 Caso de Uso Crear Malla**

---

<b>Caso de Uso</b>	<b>Crear Malla</b>
Actor Principal	Usuario (Jefe Docencia, Alumno)
Actores	
Secundarios	
Precondición	
Post condición	El sistema deja registrada la malla creada para su posterior uso.
Participantes e intereses	Jefe Docencia, Alumno: Tener una malla base para su reprogramación de la carga académica del alumno
Escenario Principal	<ol style="list-style-type: none"> <li>1) El Usuario selecciona la opción “Nueva Malla”.</li> <li>2) El Sistema despliega un formulario en donde irán el nombre de la carrera, el numero de períodos máximo y mínimo, numero de asignaturas obligatorias que contempla el plan de estudio, generales y optativos, la sigla de la carrera (tres caracteres), numero de créditos máximo y mínimo por semestre y el numero máximo y mínimo por semestre de asignaturas.</li> <li>3) El Usuario ingresa todos los campos solicitados y acepta.</li> <li>4) El Sistema muestra un nuevo formulario para el ingreso de las asignaturas, en donde los datos solicitados son: nombre, clave, número de créditos y el semestre que se dicta.</li> <li>5) El Usuario ingresa los datos solicitados de todas las</li> </ol>

---

---

asignaturas obligatorios.

6) El sistema mostrara el listado de las asignaturas ingresadas dando la opción, a cada uno, del ingreso de prerrequisitos.

7) El usuario ingresa los prerrequisitos de cada uno si así es necesario y finaliza.

8) El Sistema despliega la malla creada.

---

Escenario

Alternativo

---

Observaciones El formato de la clave de las asignaturas tendrá que ser la misma que aparece en el Navegador Académico esto es para las asignaturas obligatorias

---

Asociaciones

---

## 8.2.2 Caso de Uso Abrir Malla

---

**Caso de Uso**      **Abrir Malla**

---

Actor Principal      Usuario (Jefe Docencia, Alumno)

---

Actores

Secundarios

---

Precondición	
Post condición	El sistema muestra la malla seleccionada con la que se trabajara.
Participantes e intereses	Jefe Docencia, Alumno: Tener una malla visual para su posterior trabajo en ella
Escenario Principal	<ol style="list-style-type: none"> <li>1) El usuario selecciona la opción “Abrir Malla”.</li> <li>2) El sistema despliega una ventana de dialogo, en la cual se selecciona el archivo de la malla.</li> <li>3) El Usuario selecciona una de éstas y selecciona abrir.</li> <li>4) El sistema muestra la malla seleccionada.</li> </ol>
Escenario Alternativo	3a) El archivo seleccionado no corresponde a un archivo “malla” o este se encuentra defectuoso, el sistema vuelve al inicio.
Observaciones	El archivo de la malla, es de texto plano y es creado por el caso de uso “Crear Malla”
Asociaciones	

### 8.2.3 Caso de Uso Modificar Malla

<b>Caso de Uso</b>	<b>Modificar Malla</b>
--------------------	------------------------

Actor Principal	Usuario (Jefe Docencia, Alumno)
Actores Secundarios	
Precondición	Haber realizados el caso de uso Abrir Malla
Post condición	El sistema muestra la malla modificada. Esta debe ser guardada.
Participantes e intereses	Jefe Docencia, Alumno: Cambiar datos erróneos de una malla.
Escenario Principal	<ol style="list-style-type: none"> <li>1) El sistema debe tener una malla abierta.</li> <li>2) El usuario selecciona la opción “Modificar Malla”.</li> <li>3) El sistema muestra los datos de las asignaturas obligatorias.</li> <li>4) El usuario cambia los datos de las asignaturas obligatorias y selecciona “Aceptar”, también se pueden agregar o quitar asignaturas.</li> <li>5) El sistema muestra las asignaturas y sus prerequisites.</li> <li>6) El usuario modifica los prerequisites y selecciona “Aceptar”.</li> <li>7) El sistema muestra el diálogo “Guardar Archivo”.</li> <li>8) El usuario indica el nombre del archivo donde se guardará la malla.</li> </ol>
Escenario	

Alternativo	
Observaciones	
Asociaciones	Incluye a Abrir Malla.

### 8.2.4 Caso de Uso Obtener Estado Asignaturas por Internet

<b>Caso de Uso</b>	<b>Obtener Estado Asignaturas por Internet</b>
Actor Principal	Usuario (Jefe Docencia, Alumno)
Actores	
Secundarios	
Precondición	Tener una malla cargada
Post condición	El sistema registra el estado de las asignaturas del alumno
Participantes e intereses	Jefe Docencia, Alumno: conocer las asignaturas que el alumno ya ha aprobados.
Escenario Principal	<p>1) El usuario selecciona la opción “obtener ramos aprobados”.</p> <p>2) El Sistema muestra una interfaz en donde solicita RUT y clave correspondientes al Navegador Académico.</p>

---

	3) El usuario ingresa los datos solicitados y acepta.
	4) El sistema muestra la interfaz de ingreso al Navegador Académico. Si el usuario es un alumno se dirige a “Compasión Malla” que aparece en el navegador. Si el usuario es el Jefe de Docencia se dirige a “Alumnos” luego busca al alumno a tratar, e ingresa a su “Comparación Malla”.
	5) El sistema da la opción de “obtener ramos”.
	6) El usuario selecciona la opción “obtener ramos”.
	7) El sistema muestra la malla con las asignaturas aprobadas.

---

Escenario Alternativo	4a) La clave o RUT proporcionados son inválidos. La aplicación vuelve al paso 2.
	4b) El computador no posee conexión a Internet, en este caso no se pueden obtener las asignaturas aprobadas y la aplicación muestra un error y termina su ejecución.
	4c) Las asignaturas obtenidas desde el navegador no corresponden a los de la malla actual (la malla fue mal creada), en este se avisara del error, la aplicación mostrara la opción de modificar la malla.

---

Observaciones	Se debe poseer acceso a Internet.
	La clave proporcionada debe coincidir con la del navegador.

---

Asociaciones	Incluye Modificar Malla.
--------------	--------------------------

---

## 8.2.5 Caso de Uso Indicar Estado Asignaturas Manualmente

---

<b>Caso de Uso</b>	<b>Indicar Estado Asignaturas Manualmente</b>
Actor Principal	Usuario (Jefe Docencia, Alumno)
Actores	
Secundarios	
Precondición	Tener una malla cargada.
Post condición	El sistema registra el estado de las asignaturas del alumno
Participantes e intereses	Jefe Docencia, Alumno: conocer las asignaturas que el alumno ya ha aprobado.
Escenario Principal	1) El usuario indica el estado de las asignaturas seleccionando un estado, Reprobado, Aprobado y dejando en blanco en el caso que no este cursada la asignatura.
Escenario Alternativo	
Observaciones	
Asociaciones	Incluye Modificar Malla.

---



## 8.2.6 Caso de Uso Resolver Modelo

---

<b>Caso de Uso</b>	<b>Resolver Modelo</b>
Actor Principal	Usuario (Jefe Docencia, Alumno)
Actores Secundarios	
Precondición	Tener una malla base con las asignaturas aprobadas del alumno para efectuar la operación.
Post condición	El sistema da la posibilidad de hacer un análisis de sensibilidad.
Participantes e intereses	Jefe Docencia, Alumno: conocer cual seria una carga académica ideal para el alumno en los siguientes períodos
Escenario Principal	<ol style="list-style-type: none"><li>1) El Usuario selecciona la opción "Reprogramar".</li><li>2) El Sistema muestra un formulario donde se muestran los actuales parámetros de la reprogramación.</li><li>3) El usuario realiza los cambios necesarios y presiona "Resolver".</li><li>4) El sistema muestra la malla reprogramada.</li></ol>
Escenario Alternativo	
Observaciones	Cuando el sistema de la opción de "reprogramar" se incluirá

---

---

	la opción de “análisis de sensibilidad”
Asociaciones	Incluye Abrir Malla, Obtener Estado Asignaturas por Internet o Indicar Estado Asignaturas.

---

### 8.2.7 Caso de Uso Análisis de Sensibilidad

---

<b>Caso de Uso</b>	<b>Análisis de Sensibilidad</b>
Actor Principal	Usuario (Jefe Docencia, Alumno)
Actores Secundarios	
Precondición	Haber efectuado una reprogramación de una malla.
Post condición	
Participantes e intereses	Jefe Docencia, Alumno: poder realizar cambios en la reprogramación para que se ajuste a lo que al usuario le acomoda.
Escenario Principal	<p>1) El Usuario selecciona la asignatura que desea mover al período que mas le acomode.</p> <p>2) El sistema muestra los resultados del cambio realizado, indicando los nuevos valores, como la cantidad de asignaturas por semestre, la cantidad de créditos por semestre, etc.</p>

---

---

3) El usuario repite el paso N° 1 hasta que esté conforme.

---

Escenario

Alternativo

---

Observaciones

---

Asociaciones      Incluye Resolver Modelo

---

### **8.3 Diagramas de Secuencia**

El objetivo principal por el cual se describirán los Diagramas de Secuencia es para especificar en un nivel mas bajo el Sistema, de tal manera que ayude a conocer con más detalle, como interactúan cada una de las entidades dentro éste, y así conocer los mensajes que utilizan estas entidades para comunicarse entre si para cumplir una funcionalidad.

### 8.3.1 Diagrama de Secuencia Crear Malla

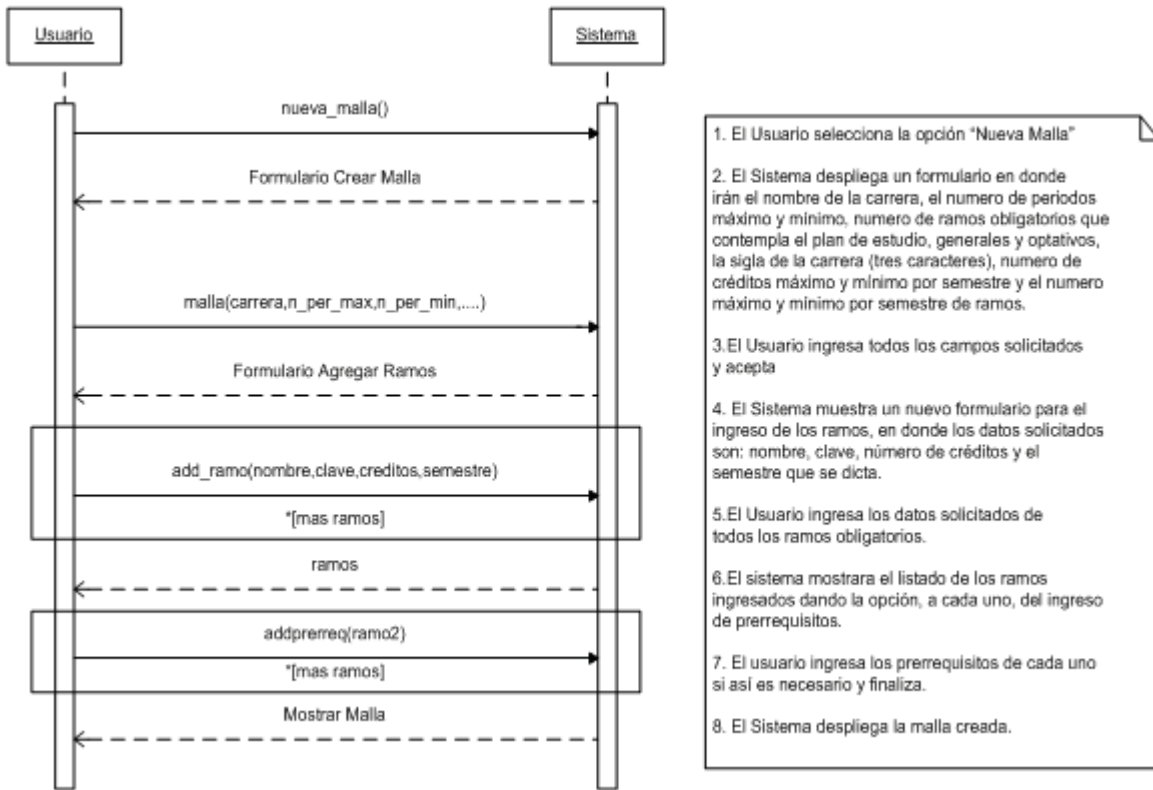


Ilustración 5: Diagrama de Secuencia Crear Malla

### 8.3.2 Diagrama de Secuencia Abrir Malla

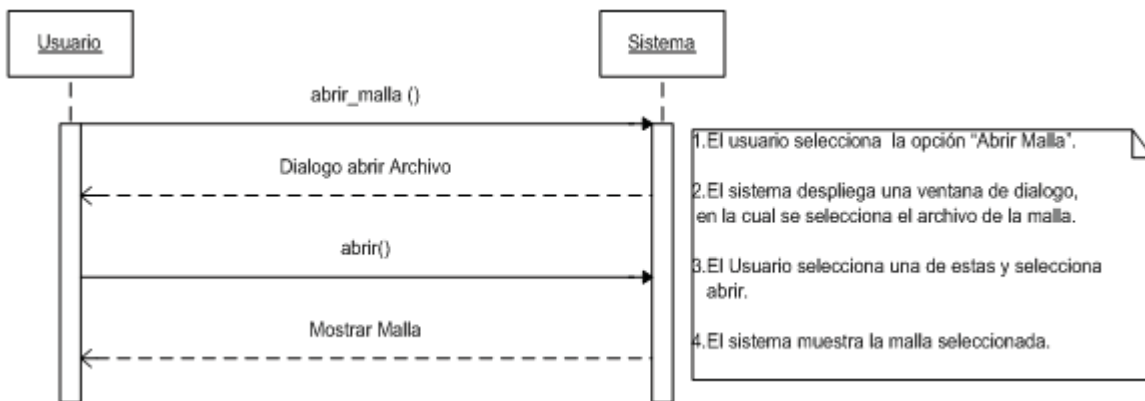


Ilustración 6: Diagrama de Secuencia Abrir Malla

### 8.3.3 Diagrama de Secuencia Modificar Malla

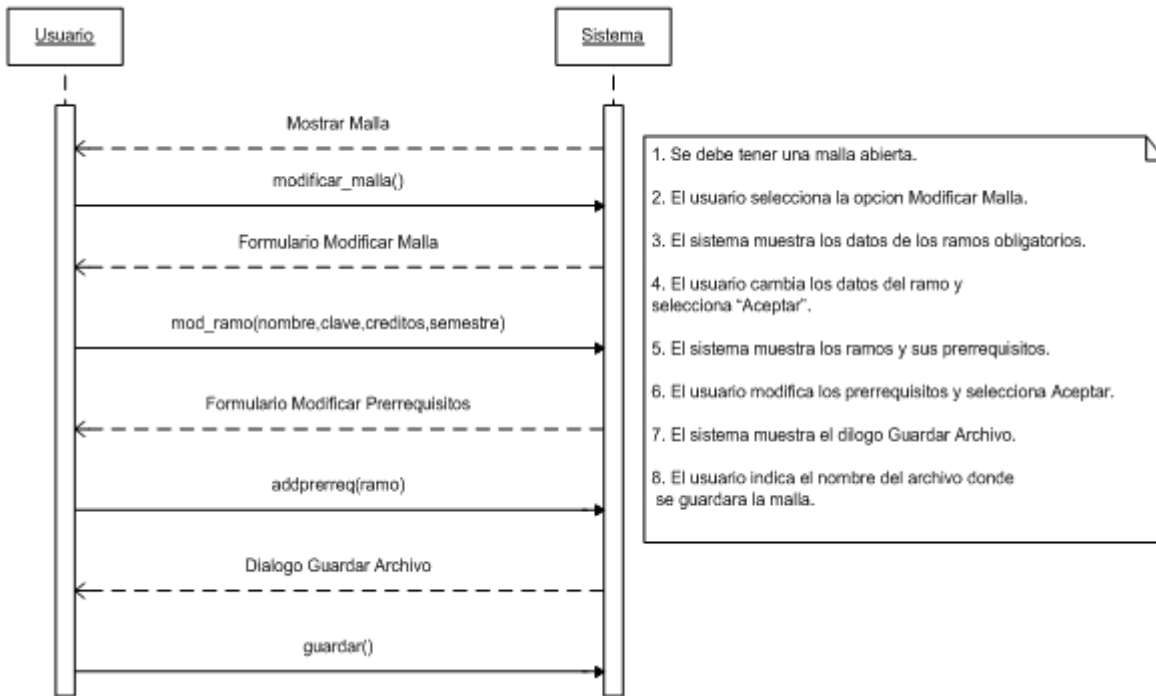


Ilustración 7: Diagrama de Secuencia Modificar Malla

### 8.3.4 Diagrama de Secuencia Obtener Estado Asignaturas por Internet

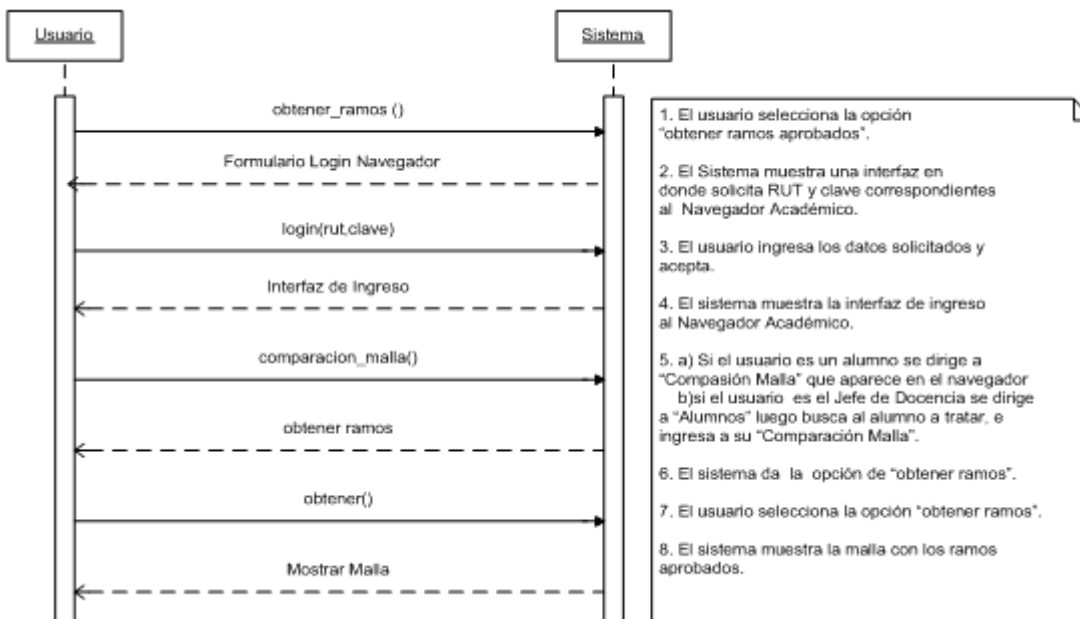


Ilustración 8: Diagrama de Secuencia Obtener Estado Asignaturas por Internet

### 8.3.5 Diagrama de Secuencia de Estado Asignaturas Manualmente

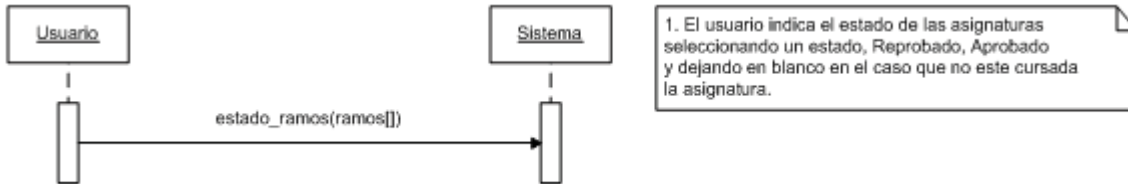


Ilustración 9: Diagrama de Secuencia de Estado Asignaturas Manualmente

### 8.3.6 Diagrama de Secuencia Resolver Modelo

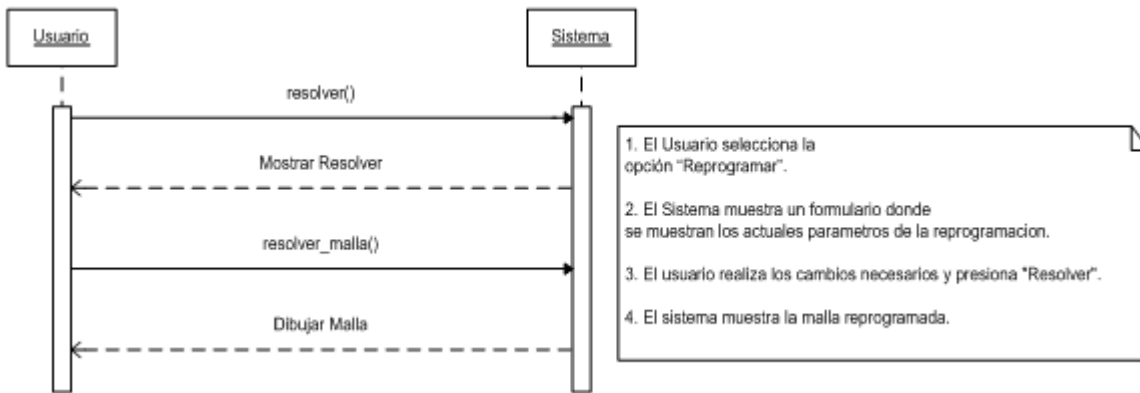


Ilustración 10: Diagrama de Secuencia Resolver Modelo

### 8.3.7 Diagrama de Secuencia Análisis de Sensibilidad

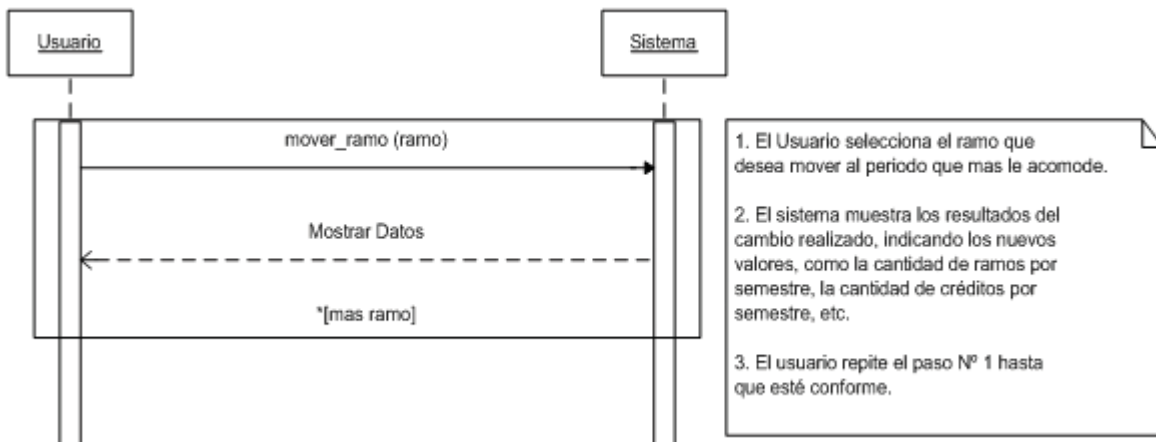


Ilustración 11: Diagrama de Secuencia Análisis de Sensibilidad

# Capítulo 9. Diseño

## 9.1 Arquitectura

La arquitectura del sistema consta básicamente de una sola aplicación “Malla”, la que realiza todas las operaciones necesarias para resolver el problema, es decir, la arquitectura del sistema caería dentro de la clasificación de “Monolítica”, consta básicamente de un solo producto que ofrece toda la funcionalidad necesaria, aunque dentro de este producto se pueden distinguir distintas unidades funcionales.

Las funcionalidades que este sistema posee son:

- La obtención de asignaturas aprobadas.
- Conocer la situación del alumno.
- Reprogramar la carga académica del alumno

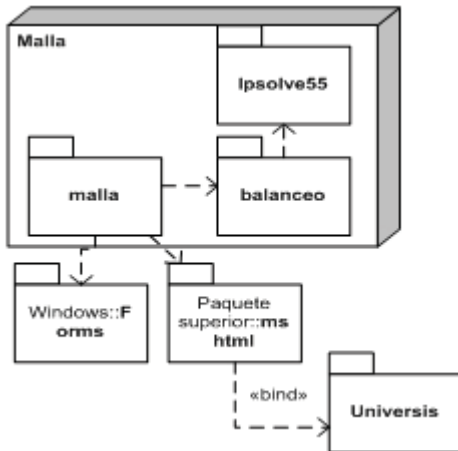


Ilustración 12: Diagrama de Implementación

El paquete balanceo, junto con Ipsolve55, contienen las funcionalidades necesarias para resolver el modelo y analizar el resultado, se encargan de implementar el modelo de la reformulación de la malla académica. El paquete malla, se encarga de la interacción con el



usuario y la parte grafica de la aplicación, las funcionalidades de mostrar el resultado y obtener los datos.

## **9.2 Rasgos**

Como se mencionó anteriormente el Desarrollo Guiado por Rasgos (FDD) se debe utilizar rasgos que corresponden a características del Sistema.

La aplicación de Reprogramación de la carga académicas, se identificaron 4 rasgos los cuales se detalla a continuación.

### **9.2.1 Rasgo Reprogramar**

Este rasgo es la base de la aplicación ya que a este será el encargado de resolver el modelo de programación lineal entera según las variables que a este le sean ingresadas y que sean necesarias para que entregue un resultado. Este resultado será desarrollado mediante librerías de “LP\_SOLVE” que entrega como un soporte al lenguaje que se esta utilizando para realizar la aplicación.

### **9.2.2 Rasgo Mostrar Malla.**

Después de que el rasgo Reprogramar entregue un resultado, éste tiene que ser desplegado al usuario de manera que sea de una manera mas intuitiva y fácil de comprender para este.

### **9.2.3 Rasgo Obtención de Datos.**

Para que el modelo tenga sus variables es necesario que sean ingresados de alguna forma, por lo que este rasgo se encarga de eso, principalmente consiste en obtener las asignaturas reprobadas, aprobadas y desde ahí, concluir cuales no han sido cursadas. La obtención puede ser tanto manualmente, es decir ingresado para cada asignatura su estado, como a través del navegador académico.

Un punto importante referente a lo mencionado sobre la manera en cual se obtienen los datos del Navegador Académico, es que no es directamente sobre la base de datos con la que éste

cuenta. Esto es porque al ser una base de datos con información privada de la Universidad el ingreso a ella es bastante restringida, por lo que se optó de obtener estos datos de una forma mas independiente de esta base de datos, esto se hizo a través de la captura del código fuente de página del Navegador Académico al momento en que el alumno o Jefe de Docencia se encuentre en la ficha “Comparación Malla” en donde se entregan las asignaturas que el alumno ha cursado y estado de éstas.

### 9.2.4 Rasgo Análisis de Sensibilidad.

Este rasgo permite que una vez entregado el resultado de una posible solución, el usuario tenga la capacidad de poder mover una asignatura hacia un semestre que desee. Esto con la finalidad de que la malla sea acomodada a las necesidades del alumno considerando también el levantamiento de prerrequisitos en algunos casos.

## 9.3 Diagrama de clases

Este diagrama de clases representa el dominio central de nuestra aplicación, es decir el rasgo de Resolución del Modelo. Las clases han sido simplificadas, por razones de tamaño.

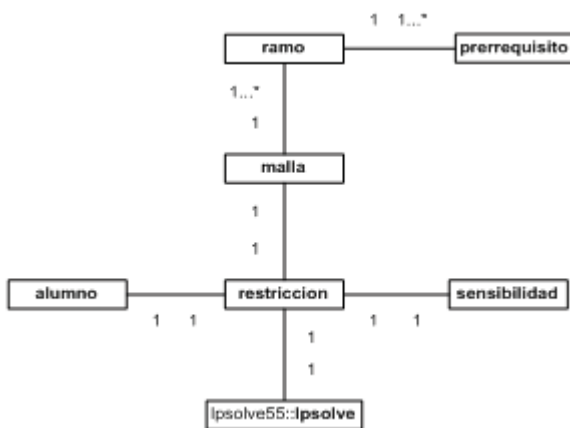


Ilustración 13: Diagrama de Clases Balanceo

Clase “prerrequisito”: encargada de mantener la relación entre una asignatura y sus prerrequisitos.

Clase “ramo”: contiene la información necesaria de cada asignatura (créditos, nombre de la asignatura, período en el que se dicta, etc.).

Clase “malla”: es una agrupación de asignaturas de un mismo currículum académico, maneja datos importantes para la reprogramación de un currículum, como por ejemplo: la cantidad de créditos mínimos y máximos, la cantidad de asignaturas, el número de períodos, etc.

Clase “alumno”: posee información sobre la situación actual de un alumno, relevante para realizar la reprogramación de un currículum (semestre actual, asignaturas aprobadas, reprobadas, no cursadas, etc.).

Clase “restriccion”: clase encargada de generar las restricciones del modelo, para esto es necesario tener definida una malla y los datos del alumno.

Clase “sensibilidad”: posee los métodos necesarios para poder realizar pequeños análisis de sensibilidad al resultado final.

Clase “Ipsolve”: contiene los métodos necesarios para resolver un modelo de programación lineal.

Dentro del paquete “malla”, se encuentran las siguientes clases:

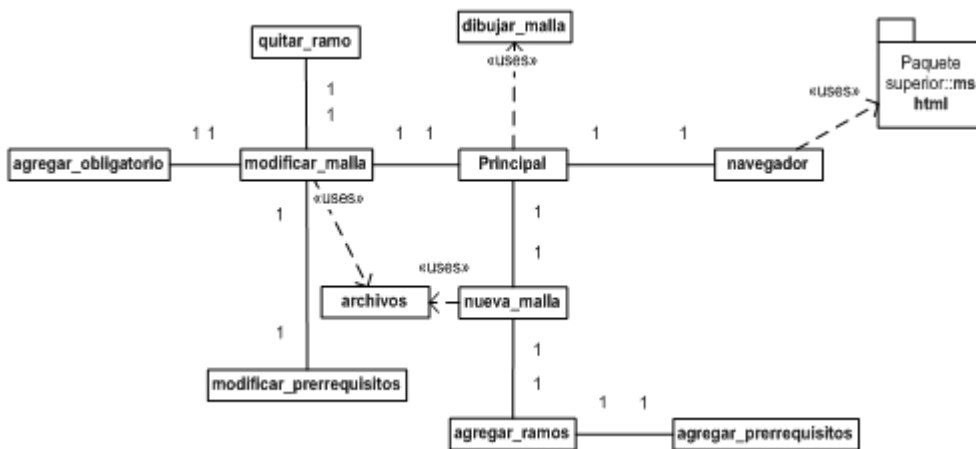


Ilustración 14: Diagrama de Clases Malla

Estas clases son las encargadas de visualizar la aplicación, son parte de la interfaz de usuario del programa.

## Capítulo 10. Plan de Prueba

Existen varios tipos de pruebas envueltos en dos grandes bloques que son: Pruebas de Caja Blanca y de Caja Negra. El primero involucra un tipo de prueba exhaustivo que da, con mayor y mejor detalle, los posibles fallos del Sistema a costo de una gran cantidad de tiempo y recursos que se requiere para realizarlo. Por otro lado, se encuentran las pruebas con Caja Negra las que se centran en lo que se espera de un módulo, es decir, intentan encontrar casos en que el módulo no se atiene a su especificación. Por ello, se denominan pruebas funcionales o casos de prueba, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro.

Con lo dicho anteriormente para efectos de este proyecto el tipo de prueba utilizado es de Caja Negra a través de los Casos de Pruebas

### 10.1 Casos de prueba

Caso de Prueba	Condición	Entrada	Salida
Validar datos ingresados a malla	Estar creando una malla o modificando una	Nombre de la carrera, Numero de períodos, Numero de asignaturas, Numero de créditos, Nombre de la asignatura, clave de la asignatura, créditos de la asignatura,	Datos correctos: se abre una ventana para ingresar el nombre del archivo malla que contendrá la información de la carrera.  En Caso de no ingresar el tipo de datos pedidos se muestre mensaje con el

		semestre de la asignatura y los prerrequisitos.	error.  Faltan datos: envía mensaje que faltan datos.
Obtener datos	Tener una cuenta en el Navegador Academizo de la universidad	Rut y password del usuario.	Datos correctos: se abre una ventana con la ficha del alumno.  En caso contrario el navegador va a solicitar los datos correctos.
Validar ingreso de datos en resolver	Tener abierta una malla con las asignaturas reprobadas y aprobadas del alumno	Máximo y mínimo de créditos por semestre, máximo y mínimo de asignaturas por semestres, período actual, numero de semestres a reprogramar	Datos correctos: se procede a realizar una reprogramación de la malla.  En Caso de no ingresar el tipo de datos pedidos se muestre mensaje con el error.  Faltan datos: envía mensaje que faltan datos.
Validar el ingreso de una malla existente	Tener un archivo malla	Archivo con extensión .malla	Si el archivo es del formato solicitado, muestra las asignaturas de la malla de la carrera.  Si el archivo no corresponde al tipo de archivo solicitado

			muestra mensaje de error.
Validar cambio de semestre de asignaturas en análisis de sensibilidad	Haber resuelto una carga académico	Numero de semestre en que se desea mover la asignatura	Si el dato es correcto la asignatura es movida al semestre indicado.  El dato no corresponde muestra el mensaje de que el semestre no corresponde.

## 10.2 Detalles de Casos de Pruebas

Con el motivo que al momento de hacer las pruebas al Sistema no existan duda de cuales serial las principales pruebas a realizar es que se detallan a continuación cada caso de prueba.

### 10.2.1 Validar datos ingresados a malla

Propósito:	Ver que los datos ingresados a la malla sean los correctos para su lectura correcta
Prerrequisitos:	Estar creando una malla o modificando alguna.
Datos de Prueba:	Nombre de la carrera={inf. ejecución}  Numero de períodos={4,2,6}  Numero de asignaturas ={20,15,10}  Numero de créditos={40,25,20}

	<p>Nombre de la asignatura = {inf111,inf112,inf112, ... , inf130}</p> <p>clave de la asignatura = {inf111,inf112,inf112, ... , inf130}</p> <p>créditos de la asignatura = {4,3,5}</p>
Pasos:	<p>1.- hacer clic en el icono de modificar o crear nueva malla.</p> <p>2.- ingresar los datos nombrados en la nueva ventana que se desplegó.</p> <p>3.- aceptar</p> <p>4.- ingresar los prerrequisitos a cada una de las asignaturas</p> <p>5.- darle un nuevo nombre o reemplazar uno existente al archivo .malla</p>
Notas y Preguntas:	

### 10.2.2 Obtener datos

Propósito:	Probar que los datos se pueden obtener del navegador y que estos sean entregados correctamente.
Prerrequisitos:	<p>Tener una cuenta valida en el Navegador Académico de la universidad.</p> <p>Tener abierta una malla curricular.</p>
Datos de Prueba:	Rut = {15973521-4}

	Password= {password correspondiente al navegador}.
Pasos:	<p>1.- hacer clic en icono correspondiente a la opción de Obtener datos.</p> <p>2.- ingresar Rut y password del navegador.</p> <p>3.- aceptar.</p> <p>4.- ingresar a la sección Comparación Malla del Navegador Académico.</p> <p>5.- hacer clic en el botón Capturar.</p> <p>6.- cerrar la ventana.</p>
Notas y Preguntas:	

### 10.2.3 Validar ingreso de datos en resolver

Propósito:	Probar la valides de los asignaturas aprobadas y reprobadas al momento de resolver el modelo.
Prerrequisitos:	Tener abierta una malla curricular.
Datos de Prueba:	<p>asignaturas = {aprobada, reprobada, no cursada}</p> <p>Máximo de créditos por semestre= {18,22,19}</p> <p>mínimo de créditos por semestre = {3}</p> <p>máximo de asignaturas por semestres = {10,6,8}</p>



	<p>mínimo de asignaturas por semestres = {1}</p> <p>período actual={4,5}</p> <p>numero de semestres a reprogramar = {11,10,9}</p>
Pasos:	<p>1.- seleccionar la opción aprobada o reprobada a cada una de las asignaturas</p> <p>2.- hacer clic en resolver.</p> <p>3.- ingresar los datos en los campos correspondientes.</p> <p>4.-hacer clic en resolver.</p>
Notas y Preguntas:	

#### 10.2.4 Validar el ingreso de una malla existente

Propósito:	Probar que la carga de un archivo .malla sea exitoso
Prerrequisitos:	Contar con un archivo .malla
Datos de Prueba:	Archivo .malla= {ejecución completa .malla}
Pasos:	<p>1.- hacer clic el icono correspondiente a abrir malla.</p> <p>2.- seleccionar el archivo .malla</p> <p>3.- hacer clic en aceptar</p>
Notas y Preguntas:	

### 10.2.5 Validar cambio de semestre de asignaturas en análisis de sensibilidad

Propósito:	Probar el cambio de semestre en el análisis sea correcto
Prerrequisitos:	Haber resuelto una reprogramación de una carga académica
Datos de Prueba:	Numero de semestre = { 9, 10, 11 }
Pasos:	1.- hacer clic sobre la asignatura que se desea mover. 2.- hacer clic en aceptar.
Notas y Preguntas:	

# Capítulo 11. Descripción de la Aplicación

## **11.1 Requerimientos de Software para la Ejecución**

El software fue desarrollado con C#, un lenguaje perteneciente a la plataforma .Net de Microsoft, por este motivo necesita del .Net Framework ver. 1.1, un S.O Windows que soporte este framework (Windows 98 en adelante), y para la obtención de datos de alumno Internet Explorer 5.5 o superior aunque se puede ejecutar sin esta característica. En cuanto a requerimientos de hardware, solo se necesita que soporte los requerimientos anteriores, pero fue probado en un procesador AMD-K6 450 Mhz, 128 MB RAM, con Windows 2000 Profesional y no presentó problemas.

## **11.2 Manual de Usuario**

Lo primero que se muestra al ejecutar el programa es esta pantalla, con 2 opciones: “Abrir Malla” y “Nueva Malla”.

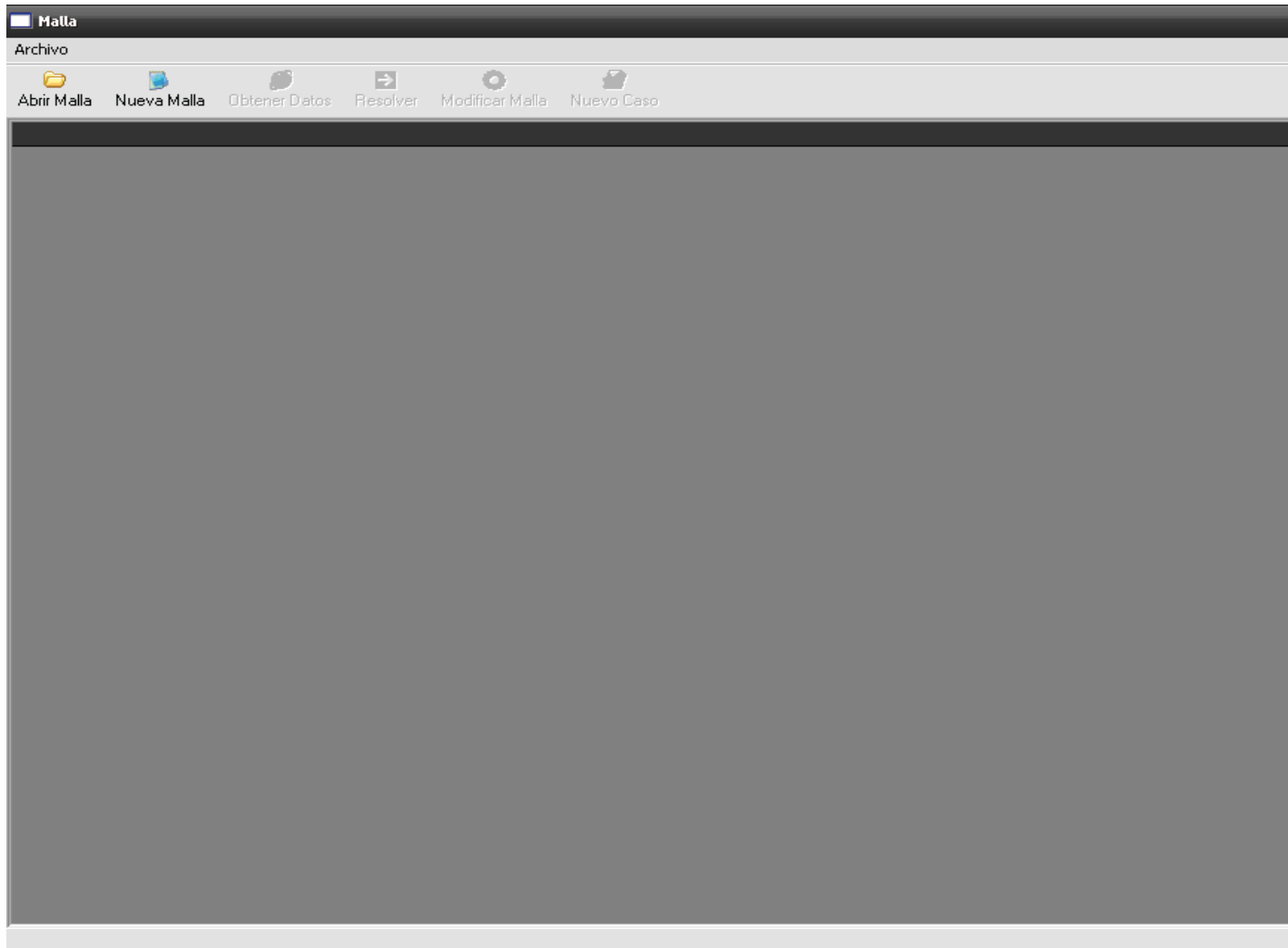


Ilustración 15: Interfaz Principal

Dependiendo de lo que se quiera hacer, se debe elegir una de las opciones anteriores.

### 11.2.1 Abrir Malla

En la ventana principal seleccionamos “Abrir Malla” y nos aparecerá el siguiente diálogo.

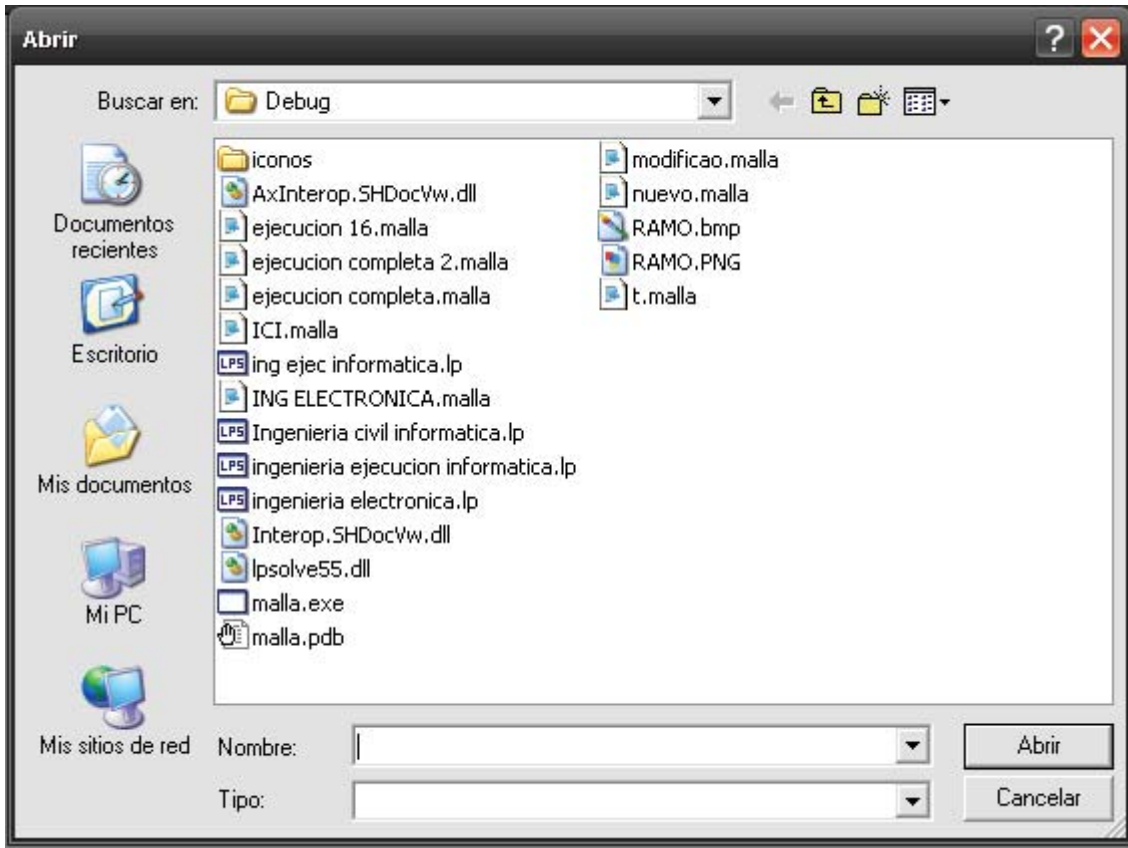


Ilustración 16: Interfaz Abrir Malla

Aquí se busca el archivo malla necesitado. Al abrir un archivo, mostrará un diálogo indicando que el archivo era válido, o por el contrario, que no lo era. Si todo salió correctamente, mostrará una pantalla similar a ésta.

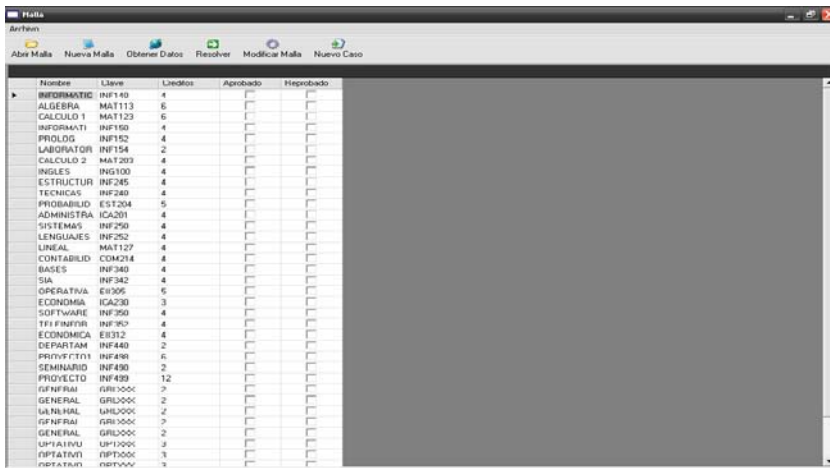
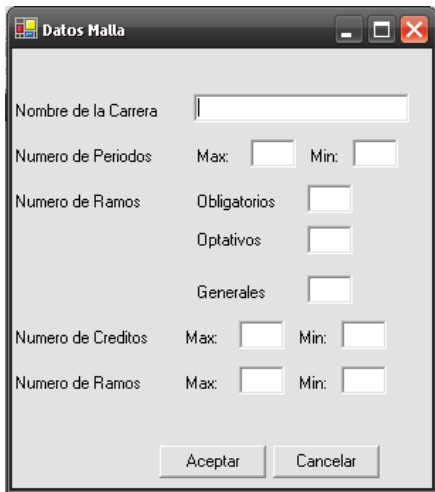


Ilustración 17: Interfaz Mostrar Malla

Aquí se muestra una lista con las asignaturas de la malla seleccionada.

## 11.2.2 Crear Malla

Al seleccionar la opción “Crear Malla” aparecerá el siguiente diálogo.



The screenshot shows a dialog box titled "Datos Malla". It contains several input fields and checkboxes. The fields are: "Nombre de la Carrera" (text input), "Numero de Periodos" (Max: [input], Min: [input]), "Numero de Ramos" (Obligatorios: [checkbox], Optativos: [checkbox], Generales: [checkbox]), "Numero de Creditos" (Max: [input], Min: [input]), and "Numero de Ramos" (Max: [input], Min: [input]). At the bottom, there are two buttons: "Aceptar" and "Cancelar".

Ilustración 18: Interfaz Datos Malla

Donde los datos que se deben ingresar son el nombre de la carrera, el número máximo y mínimo de períodos de la carrera, el número de asignaturas obligatorias, optativas y generales que posee el currículo, donde dentro de generales, se debe incluir las asignaturas de Cultura Religiosa, luego se indican el máximo y mínimo de créditos por semestre así como también de asignaturas por semestre.

Al aceptar el siguiente dialogo es mostrado:



The screenshot shows a dialog box titled "agregar\_ramos". It contains a message: "Ingrese los datos de los ramos 'OBLIGATORIOS'". Below this, there are two sets of input fields. Each set includes: "Nombre del Ramo" (text input), "Clave del Ramo" (text input), "Creditos del Ramo" (text input), "Semestre del Ramo" (dropdown menu with "Semestral" selected), and "Semestre del Ramo" (dropdown menu with "Semestral" selected). A vertical scrollbar is visible on the right side of the dialog box.

Ilustración 19: Interfaz Agregar Ramos

Donde, por cada asignatura se deben ingresar los datos solicitados, es importante destacar que la clave, de la asignatura, debe ser la misma que aparece en el navegador en la sección “Comparación Malla”, esto es, en mayúscula y sin identificador de paralelo. Al terminar se debe presionar “Aceptar”, al final del formulario.

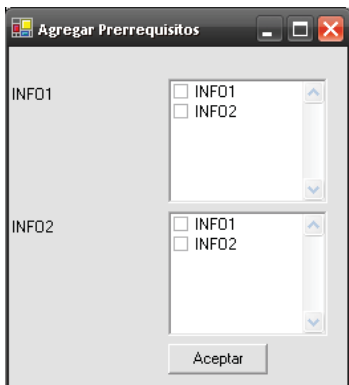


Ilustración 20: Interfaz Agregar Prerrequisitos

En el siguiente diálogo, se muestra cada asignatura y junto a él, una lista de todas las asignaturas de la malla, las que deben ser marcadas como prerrequisitos, si así corresponde, en la imagen superior INFO2 debería tener marcado INFO1 como prerrequisito. Al hacer clic en aceptar en este formulario preguntará donde se desea guardar el archivo y con que nombre. Si la operación fue exitosa, debería salir un mensaje indicándolo.

### 11.2.3 Obtener datos

Con una malla abierta, la opción “Obtener datos”, se mostrará habilitada, al hacer clic aparecerá una ventana, que carga la página Web del Navegador Académico de la universidad, donde se debe ingresar como comúnmente se hace y seleccionar la opción “Comparación Malla”.



Ilustración 21: Interfaz Obtener Datos

En esta sección, el botón “Capturar”, de la esquina superior izquierda se habilitará, al hacer clic la página cambiará a “Avance Curricular”, cuando la página se encuentre completamente cargada, esta ventana puede ser cerrada. Y en la ventana principal, se mostrará la situación actual del alumno

### 11.2.4 Modificar Malla

Teniendo una malla abierta, se puede seleccionar la opción “Modificar Malla”, que en cuanto a diálogos es bastante similar a los de Crear Malla, la diferencia, es que al final del primer formulario se encuentran las opciones “Agregar Obligatorio” y “Quitar Obligatorio”, además de poder indicar el número de asignaturas optativas y generales de la malla.

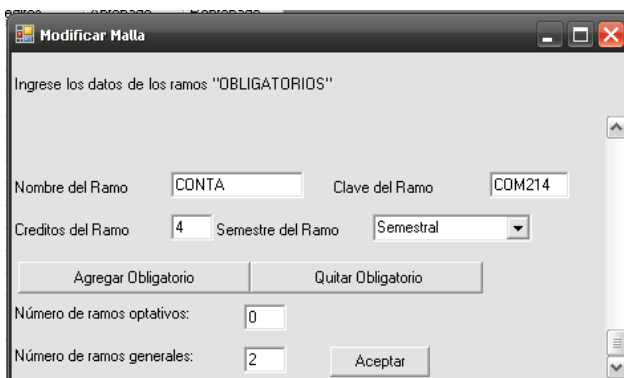


Ilustración 22: Interfaz Modificar Ramos

Al seleccionar la primera opción, “Agregar Obligatorio”, aparece la siguiente ventana:



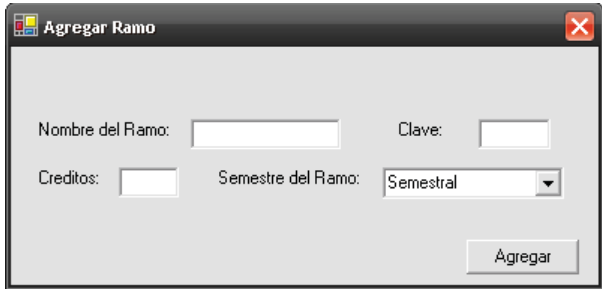


Ilustración 23: Interfaz Agregar Ramo

Donde los datos deben cumplir las mismas restricciones que en “Crear Malla”, al aceptar la nueva asignatura aparecerá dentro de las asignaturas listadas en el dialogo anterior.

Al seleccionar “Quitar Obligatorio”, se muestra:

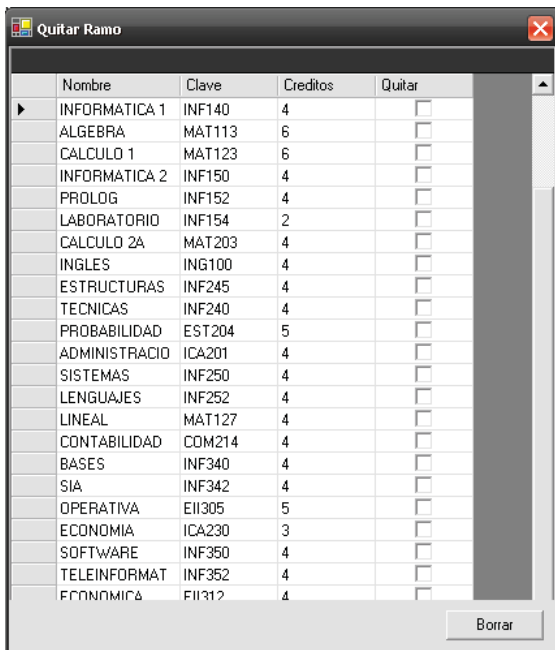


Ilustración 24: Interfaz Quitar Ramo

En esta pantalla se seleccionan las asignaturas que se desean eliminar y luego se presiona el botón “Borrar”.

Al finalizar el proceso de “Modificar Malla”, se mostrará el mismo diálogo para indicar los prerrequisitos que en “Crear Malla”, donde se puede modificar la relación de prerrequisitos, y luego se debe indicar, el archivo, dónde se desea guardar los cambios.

## 11.2.5 Resolver

Al presionar la opción “Resolver” desde el menú principal, aparecerá un dialogo donde se debe indicar los parámetros para la reprogramación de la malla y de la situación actual.

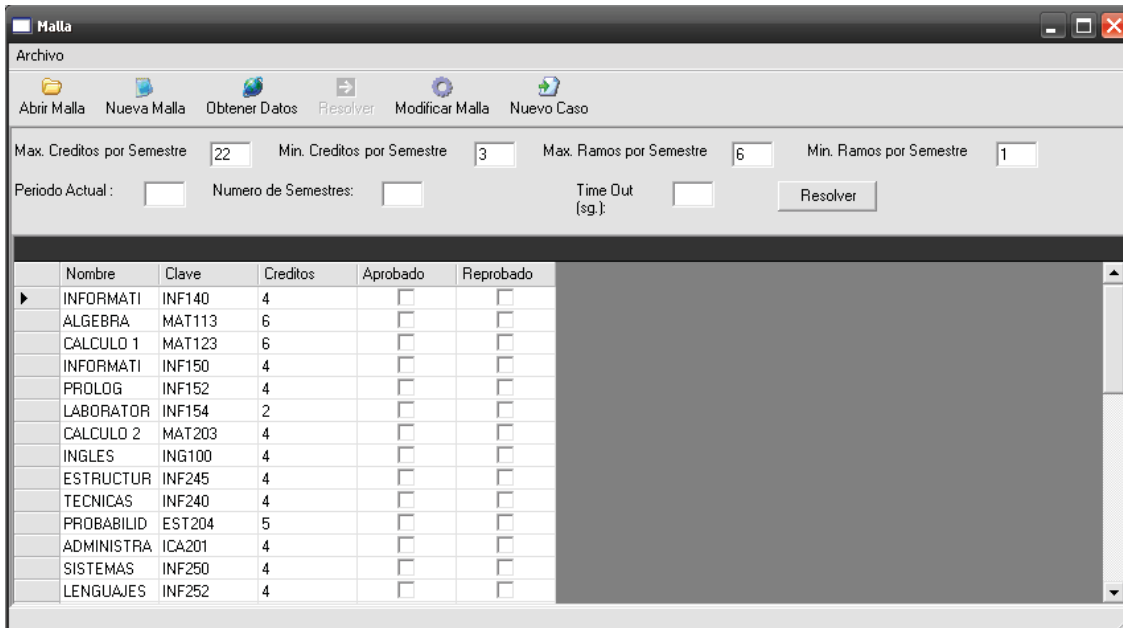


Ilustración 25: Interfaz Resolver Malla

En esta parte se debe indicar el período actual del alumno, es decir, el último período terminado, el número de semestres de duración total de la carrera, incluyendo los ya terminados, “Time Out” corresponde al tiempo en el que se detendrá la aplicación para devolver la solución factible, se recomienda que este no pase de los 30 sg y no sea menor a 10 sg, dado que en ciertos casos, muy poco tiempo puede devolver una solución no óptima. Al presionar “Resolver” se muestra la solución de la reprogramación.

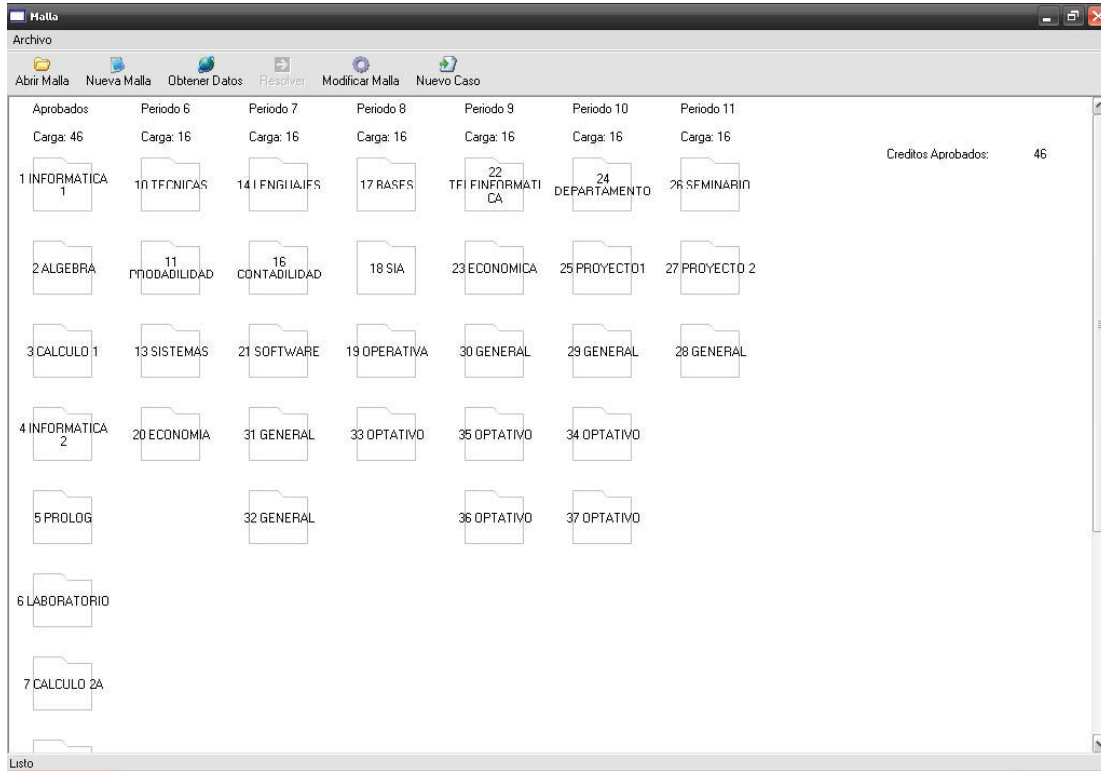


Ilustración 26: Interfaz Mostrar Resultados

Al seleccionar una asignatura se muestran los detalles de ésta, al costado derecho de la aplicación:

Creditos Aprobados: 46

21 SOFTWARE

INF350

4

Semestre:

Ilustración 27: Interfaz Análisis de Sensibilidad

Al cambiar el valor de “Semestre” la asignatura será reubicada en el semestre indicado, actualizando los valores “Carga” de cada columna.

## Conclusiones

Una malla balanceada permite un desarrollo de las capacidades de estudio del alumno e incrementa sus posibilidades de terminar prontamente su currículum. Lo que suele suceder es que por diversos motivos estas mallas se van modificando a lo largo de toda la carrera dado que no siempre los alumnos aprueban las asignaturas en las primeras oportunidades y a veces necesitan una pequeña ayuda o guía para replantear o reformular su currículum, el acercamiento matemático que se ha presentado y la futura aplicación o solución deberán prestar esta ayuda mejorando así el desempeño de los alumnos en la Universidad.

Sin lugar a duda la programación lineal es de gran ayuda y lo son también los avances que se han hecho en este ámbito por los diseñadores de software o desarrolladores, que han sido capaces de desarrollar herramientas de gran utilidad, sin las cuales sería bastante complicado desarrollar una solución de estas características. Herramientas como “LP\_SOLVE” son de gran difusión en la red por la gente interesada en el área de la programación lineal, constantemente surgen grupos de discusión y de desarrollo que tratan de mejorar o buscar nuevas formas para implementar estas herramientas, como por ejemplo, las adaptaciones de librerías para otros lenguajes o plataformas, permitiendo a los desarrolladores centrar su atención en el modelo y no en aprender un nuevo lenguaje de programación que muchas veces puede ser incompleto, carecer de funcionalidades o ser difícil de comprender.

La malla completa de la carrera de Ingeniería de Ejecución y una situación similar a la del ejemplo de resolución anterior puede llegar a tener más de 500 restricciones y cerca de 450 variables, basta imaginar una malla con más períodos y el problema se complica mucho más, he ahí el interés por desarrollar una herramienta que ayude a encontrar una solución de manera confiable y aliviane el trabajo que se genera en estas circunstancias.

## **Verificación del Cumplimiento de los Requerimientos**

Con respecto al ingreso de los datos, la aplicación necesita un archivo de texto plano con los datos de la malla, este archivo puede ser creado con la misma aplicación seleccionando del menú la opción “Crear Malla”. Este archivo guarda datos como, el número de asignaturas generales, optativas y obligatorias de la malla y el número máximo y mínimo de asignaturas, tanto como de créditos por semestre que se deben considerar al momento de reprogramar la carga académica; posee cada asignatura con su respectivo nombre y clave, créditos y un identificador de la asignatura. Al final de cada asignatura, se lista el identificador de los prerrequisitos de la asignatura. Antes de realizar una reprogramación, los datos son presentados al usuario de manera que este pueda modificar estos parámetros, estos cambios solo afectarán la reprogramación actual y no se guardarán para futuras reprogramaciones; es importante destacar que para realizar la reprogramación los principales datos son aquellos ingresados por el usuario: asignaturas aprobadas, reprobadas, semestre actual y número de semestres de reprogramación.

Sobre el balanceo de la malla, el principal parámetro para la realización de la reprogramación es el número de créditos por semestre, el modelo se encarga de equiparar semestre por semestre la cantidad de créditos, haciendo que ésta sea la mínima y más pareja posible, dentro de los límites establecidos al momento de reprogramar la carga, esto sin desmerecer las restricciones del número de asignaturas por semestre; un punto importante a considerar son los valores que indican el período actual y el número de períodos en los que se desea realizar la reprogramación, ya que de éstos dependen también la exigencia y exactitud que posea el resultado final.

Sobre la presentación del resultado final, éste se muestra en columnas que identifican los semestres y dentro de éstas, las asignaturas programadas para cada semestre, se intentó de presentar el resultado de una manera similar en la que son representadas en la mayoría de los casos y específicamente en la escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso.

Con estas características la aplicación cumple con los requerimientos establecidos.

## **Verificación del Cumplimiento de los Objetivos**

Respecto al modelo de programación lineal. Es capaz de resolver el problema siempre que los datos de entrada así lo permitan, cubre las principales características de una malla curricular (prerrequisitos, carga académica, número de asignaturas por semestre, etc.).

Se implemento una manera de obtener los datos del alumno a través del navegador académico, esto se realizo para minimizar la intervención del usuario.

Sobre la reducción del trabajo de los Jefes de Docencia, un caso promedio demora de 15 a 30 minutos en reprogramar, con la aplicación, la reprogramación puede ser realizada en 2 o 3 minutos si se cuenta con la malla definida previamente (definir una malla demora de 5 a 10 minutos, dependiendo de la complejidad de los prerrequisitos, número de asignaturas que ésta tenga y el conocimiento que tenga la persona sobre los datos de la malla), la reprogramación también depende de la conexión a Internet que se posea, aunque si es realizada por un alumno que conozca muy bien su situación, la aplicación puede prescindir de la conexión y realizar la reprogramación sin obtener los datos de navegador académico.

Con las características anteriores, los objetivos específicos se ven cumplidos, por lo tanto también el objetivo general.

## **Limitaciones de la Aplicación**

Obtención de los datos: uno de los principales problemas o limitantes de esta aplicación, se refiere a la manera en que se obtiene la situación actual del alumno, la cual se realizó mediante el análisis de los datos del navegador académico, por lo que existe una dependencia directa a este sistema externo y de la manera en que se accede a tal, cabe mencionar que cualquier cambio en la interfaz del navegador, puede interferir en la correcta obtención de los datos.

Definición de un currículo académico: otro de los problemas que surgen con el uso de la aplicación, es que se debe tener una definición de la malla curricular con la cual se desea trabajar, definiendo asignaturas y relaciones entre estos, en ciertos casos las carreras constan de 12 semestres con un promedio de 5 asignaturas por cada uno, teniendo que definir cerca de

60 cursos para una sola carrera, lo que puede ser un trabajo agotador y que generalmente lleva a cometer errores.

Restricciones: la aplicación considera, solo dos tipos de restricciones para la reprogramación de asignatura, prerrequisitos y semestre en el que se dicta el curso (anual o semestral), en muchas carreras esto no es suficiente para definir las relaciones de una malla ya que poseen restricciones que no pueden ser definidas en términos de estas relaciones, por ejemplo, si una asignatura debe ser tomado solo con la aprobación del Jefe de Docencia, tampoco considera situaciones en las que un alumno solicita el levantamiento de prerrequisitos.

## Para Considerar en Futuras Mejoras

Finalmente, para tener en consideración para una futura versión de la aplicación, se debe considerar la implementación de una solución a los problemas anteriormente nombrados y la rectificación de los reglamentos de la Universidad, los cuales pueden ser modificados dejando obsoletos más de algún requerimiento de la aplicación.

## Referencias

- [1] C. Castro & S. Manzano, *The design of academic curricula: A quantitative approach*. Jornadas Chilenas de Computación, Punta Arenas, Chile, 2001.
- [2] C. Vergara, Desarrollo de un modelo de programación lineal para generación de mallas curriculares balanceadas. Memorias de IEI, UTFSM, Valparaíso, Chile, 1994.
- [3] Reglamento General de Estudios, Vicerrectoría de Asuntos Docentes y Estudiantiles, Pontificia Universidad Católica de Valparaíso, <http://dpd.ucv.cl/>.
- [4] Brahim Hnich, Zeynep Kiziltan & Toby Walsh, Modelling a balanced academic curriculum problem. Proc. of CP-AI-OR02, 2002.
- [5] *Introduction to lp\_solve*, <http://lpsolve.sourceforge.net/5.5/>.
- [6] Robert Fourer (4er@iems.nwu.edu), "*Linear Programming Frequently Asked Questions*," Optimization Technology Center of Northwestern University and Argonne National Laboratory,
- [7] <http://www-unix.mcs.anl.gov/otc/Guide/faq/linear-programming-faq.html> (2000).
- [8] <http://directory.fsf.org/libs/glpk.html> - *GLPK - GNU Linear Programming Kit*

[9] <http://www.coin-or.org/Clp/userguide/index.html> -John Forrest, David de la Nuez, Robin Lougee-Heimer, CLP User Guide.

[10] Alberto Molpeceres, Procesos de Desarrollo: RUP, XP y FDD, <http://www.javahispano.org>.

## Anexos

### Anexo A: Artículo 22 Reglamento General de Estudios

**22°** Cada currículum de grado o título de acuerdo con los objetivos que lo conforman, deberá señalar:

Las asignaturas obligatorias que lo integran, sus claves, pre-requisitos y créditos que otorgan.

Las áreas de asignaturas optativas, si las hubiere, con indicación del número de créditos que el alumno deberá aprobar por este concepto para optar al grado o título respectivo.

El número de períodos académicos ordinarios que lo componen.

El número máximo de períodos académicos ordinarios en que debe ser completado por el alumno.

El período académico ordinario a contar del cual se aplicará lo dispuesto en el artículo 28°, que no podrá exceder de la mitad más uno de los semestres que componen el currículum.

### Anexo B: Artículo 28 Reglamento General de Estudios

**28°** El promedio acumulativo mínimo de créditos aprobados por semestre, exigible a contar del período académico lectivo indicado en la letra e. del artículo 22°, corresponderá al cociente entre el total de créditos del Plan de Estudios y el número máximo de períodos académicos a que se refiere la letra d. del artículo 22°. El alumno será eliminado del currículum de grado o título al finalizar el período académico en que el promedio acumulado mínimo de créditos obtenido sea inferior al establecido en el inciso precedente. No obstante lo dispuesto en el inciso anterior Vice-Rectoría de Asuntos Docentes y Estudiantiles a través del Director de Procesos Docentes podrá, excepcionalmente y ante solicitud fundada del alumno que haya



sido eliminado por tal causa, autorizarle la continuación de sus estudios, mediante resolución escrita, previa solicitud de los antecedentes del alumno a la Unidad Académica y opinión fundamentada de la Jefatura de Docencia. Dicha resolución deberá ser aprobada previamente por el Vicerrector de Asuntos Docentes y Estudiantiles. La Dirección de la Unidad Académica determinará para los alumnos ingresados vía Caso Especial y Reincorporados las condiciones especiales de adscripción a su respectivo currículum de grado o título, de acuerdo a su grado de avance curricular. Asimismo, dichos alumnos dispondrán de un plazo máximo para completar el respectivo currículum de grado o título, de acuerdo a la cantidad de créditos que restaren, tras descontar los créditos convalidados, homologados o reconocidos. El plazo se fijará a contar de la fecha de incorporación o reincorporación.

## Anexo C: Artículo 37 Reglamento General de Estudios

**37°** El alumno que reprueba una asignatura obligatoria de su currículum de grado o título deberá cursarla nuevamente en la primera oportunidad que se dicte. Si no lo hiciera, la Jefatura de Docencia de la Unidad respectiva podrá requerir la inscripción correspondiente.

## Anexo D: Ejemplo artículo 28

Ejemplo en el que se detalla la situación de un alumno que incurre en el artículo 28.

Tomemos como ejemplo la siguiente malla curricular:

Supongamos que un alumno se encuentra finalizando su 4 período, y ha aprobado las asignaturas:

INF 140, MAT 113, MAT 123, INF 150, INF 152, MAT203, ING 100, INF 245, ICR 100 y un Estudio general.



Ilustración 28: Ejemplo de Resolución

Para finales del cuarto semestre o a contar del quinto semestre, el alumno posee 42 créditos aprobados, entre asignaturas aprobadas por primera, segunda o tercera oportunidad, debiendo tener 44, es decir 11 por semestre, que es lo que indica el artículo 28, para el quinto semestre el alumno tiene la posibilidad de tomar 32 créditos, o 9 asignaturas en total respetando los prerrequisitos (oscuro suave).

El problema es saber replantear la malla y equilibrar estas asignaturas, de tal manera que la nueva malla no sea tan pesada pero a la vez que el alumno pueda avanzar rápidamente contar de no perder más tiempo para terminar la carrera.