

**CRONOS: SISTEMA DE CARGA ACADÉMICA Y PLANEACIÓN DE HORARIOS  
PARA LA UNIVERSIDAD DE MANIZALES**

**DAVID MARCELO AGUDELO RAMÍREZ  
JUAN CARLOS CORTÉS CASTRO**

**UNIVERSIDAD DE MANIZALES  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES  
MANIZALES  
2006**

**CRONOS: SISTEMA DE CARGA ACADÉMICA Y PLANEACIÓN DE HORARIOS  
PARA LA UNIVERSIDAD DE MANIZALES**

**DAVID MARCELO AGUDELO RAMÍREZ  
JUAN CARLOS CORTÉS CASTRO**

**Proyecto de Grado para optar al título de  
Ingeniero de Sistemas y Telecomunicaciones**

**Presidente  
CARLOS BETANCOURT CORREA  
Ingeniero de Sistemas  
Especialista en redes y Telecomunicaciones**

**UNIVERSIDAD DE MANIZALES  
FACULTAD DE INGENIERÍA  
INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES  
MANIZALES  
2006**

**Nota de aceptación:**

---

---

---

---

---

---

---

---

---

**Firma del Jurado**

---

**Firma del Jurado**

---

**Firma del Jurado**

## **AGRADECIMIENTOS**

Este trabajo de grado presenta muchos sentimientos y vivencias no solamente durante su desarrollo sino durante todos los periodos académicos hasta su final, es decir, este documento.

Agradezco a Dios, del cual durante algunos momentos dude o cuestione tal vez por tantos avances científicos, problemas comunes o ignorancia pero que cuando me despierto y logro ver, respirar y palpar y tantas otras emociones me doy cuenta que no hay ciencia, no hay descubrimiento, no hay razón tal que logre acercarse a El y entenderlo.

Agradezco a mi familia compañía incansable, fuente inagotable de amor de quien sin duda alguna no estuviera escribiendo ahora. A todos ellos mi amor, mi profundo respeto y admiración (Isabel, Mónica, Julio, Kike y la mas bella Belén) y a mis sobrinos que apenas inician sus vidas, todo mi amor para ustedes.

A mi querida, amable y paciente compañera Viviana, que ojalá y este momento en mi vida te sirva de aliciente y ejemplo para tu más cercano objetivo.

A Gregorio Posada (Goyo) mi primo-hermano, quien le dio un sentido más amplio y crítico a mi profesión.

Al Decano Carlos Cortes(incondicional apoyo y consejos), a José(por su exagerada paciencia e incondicional ayuda), y todos los compañeros de trabajo, de todos ustedes aprendí y que de alguna manera reflejan lo positivo de mi vida profesional. Y Carlos Betancurt, presidente de tesis y profesor.

Agradezco la paciencia y amistad de los padres de Juan Carlos compañero de tesis, también por su paciencia y constante preocupación por vernos profesionales.

Y a mi compañero de tesis de quien he aprendido mucho, mucha suerte en su vida profesional llena de metas que sin duda logrará alcanzar.

Y a usted querido lector que ha tomado este documento como referencia para sus trabajos.

David Marcelo Agudelo

Es sin duda un momento de felicidad y complacencia por sentir que he alcanzado otra meta en mi vida profesional y que el esfuerzo de muchos años se ha visto gratificado.

Gracias a todos los compañeros que influyeron en mi crecimiento personal, fueron muchos los momentos de convivencia que hicieron del estudio una agradable forma de vida.

A mis padres doy gracias por el apoyo incondicional y la orientación que día a día inculcaron en mí para ser una persona de bien y un gran profesional con ansia de nuevas metas.

A David Marcelo por su compañerismo, su liderazgo su sentido de pertenencia y comprensión, gracias por ver en nuestro trabajo de grado algo más que estudio y lograr un fuerte lazo de amistad y una grata convivencia. Muchos fueron los momentos de angustia al ver complejos problemas, pero el trabajo de equipo dio dar sus frutos y es hoy recompensado.

A Alejandra por su paciencia y comprensión, siempre he pensado que nunca se deja de crecer, el tiempo está en sentido inverso a la vida, pero no se pueden perder los sueños ni la esperanza y siempre habrá una luz que ilumine el camino.

A la familia de David Marcelo por su apoyo y comprensión en toda la historia CRONOS.

A todas las personas que quieren tomar como opción de grado Tesis, es algo más que conocimiento, es una convivencia, requiere tiempo, constancia, paciencia y dedicación.

Juan Carlos Cortés

## **PALABRAS PARA EL LECTOR**

### **Ingenieros, profesionales:**

Vivimos en una sociedad exquisitamente dependiente de la ciencia y la tecnología, en la cual difícilmente cualquiera sabe algo acerca de ciencia y tecnología.

Carl Sagan

### **Los que creen que han terminado de estudiar:**

La verdadera ignorancia no es la ausencia de conocimientos, sino el hecho de rehusarse a adquirirlos.

Karl Popper

### **Para el estudiante:**

Antiguos y contemporáneos genios apenas han podido entregarnos la tecnología en forma de dedos, para que nosotros formemos la mano que en un futuro armara el eterno rompecabezas del conocimiento.

David M. Agudelo

## CONTENIDO

	pág.
INTRODUCCIÓN	1
1. DESCRIPCIÓN DEL ÁREA PROBLEMÁTICA	3
2. OBJETIVOS	5
3. ANTECEDENTES	6
4. METODOLOGÍA	10
5. JUSTIFICACIÓN	11
6. MARCO TEÓRICO	12
6.1 FACTORES DE COMPLEJIDAD Y DE VARIEDAD DE SOLUCIÓN	12
6.2 EL SEGUNDO FACTOR EL ALGORITMO	13
6.3 ACCESIBILIDAD Y MOVILIDAD	13
6.4 LOS SISTEMAS DE INFORMACIÓN	14
6.5 LOS ALGORITMOS	14
6.6 LOS ALGORITMOS GENÉTICOS	15
6.7 DESCRIPCIÓN DE METODOLOGÍAS DE ANÁLISIS Y DESARROLLO	15
6.7.1 OMT	15
6.7.2 Ciclo de vida	15
Análisis de requerimientos	17

Diseño detallado	17
Diseño del sistema	17
6.7.3 UML	17
7 RESULTADOS	20
8 CONCLUSIONES	22
9 RECOMENDACIONES	23
BIBLIOGRAFÍA	24
ANEXOS	27

## LISTA DE ANEXOS

Anexo A. COMPARACIONES MYSQL Y POSTGRES	28
Anexo B. COMPARACIÓN DE SERVIDORES WEB	29
Anexo C. ALGORITMOS GENÉTICOS	30
Anexo D. REQUERIMIENTOS	40
Anexo E. MODELO DE OBJETOS	43
Anexo F. DIAGRAMA DE CLASES	43
Anexo G. DICCIONARIO DE CLASES Y ATRIBUTOS	44
Anexo H. MODELO DINÁMICO	53
Anexo I. MÁQUINA DE ESTADOS	53
Anexo J. DIAGRAMA DE SECUENCIA	57
Anexo K. MODELO FUNCIONAL	100
Anexo L. CASOS DE USO	100
Anexo M. DIAGRAMA DE PAQUETES	158
Anexo N. DIAGRAMA DE DESPLIEGUE	162
Anexo O. MODELO ENTIDAD RELACIÓN	164
Anexo P. MÉTODOS DE HORARIO: TÉCNICAS DE BLOQUEO	165
Anexo Q. MANUAL DE USUARIO Y DEL PROGRAMADOR	168
Anexo R. MODELO DE ENTREVISTAS	208
Anexo S. ENTREVISTAS Y PRUEBAS DEL SISTEMA	219
Anexo T. CATÁLOGO DE FUENTES	226



## RESUMEN

CRONOS: sistema de carga académica planeación de horarios y aulas es una herramienta 100% web que permite administrar y gestionar todo lo relacionado con la asignación de cargas académicas: clases u ofrecimientos, proyectos, horarios, disponibilidad en docentes y empleados administrativos, reservas de aulas, clasificación de aulas, administración de recursos académicos e inmobiliarios, creación de planes académicos, programas, facultades y dependencias. Un modelo web que permite crear reportes para toma de descisiones por parte del personal administrativo de la Universidad.

Cronos se convierte en una herramienta desde el punto de vista de desarrollo como un modelo portable y escalable, usando herramientas que le permiten este cometido como modelos MVC\*, AJAX\*\*, validaciones multiplataforma y construcción dinámica de Webs desde la base de datos.

Desde el punto de vista de la seguridad CRONOS presenta modelos de seguridad por capas, perfiles de usuario, validaciones MD5 y conexiones seguras a bases de datos

Se presenta entonces, una herramienta que puede ser el mejor acompañante en procesos académicos como lo es la asignación de carga académica la cual acorta tiempos de diseño y se vuelve mas cómoda y fácil de usar para tareas tan complejas como la elaboración de horarios, asignación de cargas y proyectos entre otros.

---

\* MVC: Modelo Vista Controlador, que permite conectividad a diferentes motores de bases de datos evitando procesos de re-ingeniería

\*\* AJAX: Javascript asíncrono y XML, que permite que las paginas web parezcan par el usuario final como si fuera una herramienta instalada en su computador y omitiera las posibles desventajas de aplicaciones web como es el refresco en ellas

## **RESUME**

The present work must like objective characterize, standart and modernize the systems or models of allocation of academic load of the different faculties from the University of Manizales improving its strategic managements and traditional models generating an only system for all 100% Web and with free software, that satisfies necessities with schedules, modules of creation in educational availability, allocation of boosting charge from types (extension, preparation of class, investigation between as.) labor entailment. Cronos appears then a system 100% Web, with models of security at level of servant, data bases and profiles of user who guarantee the integrity of the data, agreed process of construction of dynamics for webs, dynamic menus to each profile, and mainly to be a useful tool for the organizational component of the University, is a model of consultation as much for the educational one as for the student, allowing the impression of schedules, academic grades, matters from the home or the office of work.

## INTRODUCCIÓN

El presente trabajo tiene como objetivo caracterizar, Homogenizar y modernizar los sistemas o modelos de asignación de carga académica de las diferentes facultades de la Universidad de Manizales mejorando sus gestiones estratégicas y modelos tradicionales generando un único sistema modelo para todos.

Desde un principio la Universidad de Manizales y muchas otras organizaciones de diferente carácter han planteado la necesidad de regular sus acciones y recursos y que a su vez permita administrarlos, este proyecto CRONOS busco durante todo su desarrollo y como se ve en su producto final orientarse hacia una idea o necesidad determinada, desarrollarla, mejorarla y postular cambios de integración que le sean mas útil y ágil a la hora de interactuar con todo su entorno. Es por eso que también orientándonos hacia esa necesidad logramos dar a conocer exactamente, quién o quiénes eran responsables de una u otra labor y en que momento de su horario laboral(carga académica adicional)lo harían. Este proceso que inicialmente era una actividad intuitiva, fue perfeccionándose gradualmente y con el desarrollo de este proyecto evolucionó a modelos que reforzarían su carácter racional y por lo tanto han ido profundizando y refinando sus mecanismos de funcionamiento y formas de ejecución, hasta convertirse en un subsistema que, adaptados a características concretas y particulares, han pasado a formar parte elemental y punto de atención en su desarrollo final.

Con el desarrollo de sistemas de planeación de tiempo y agenda la forma de enfrentar situaciones objetivas ha exigido una mayor profundidad de análisis y conceptos para asumir funciones o desempeñar papeles determinados y mantener al menos un nivel de competencia que permita sobrevivir. Derivados de este proceso surgen ideas y términos como la investigación de operaciones y todo lo que ella representa, enfrentándola con algoritmos un tanto mas interesantes como los AG's, Algoritmos genéticos. Al final de este documento usted encontrara una lectura que nos explica que es un algoritmo genético, su anatomía y como puede apoyarnos para una versión futura de este desarrollo, sobre todo en el desarrollo de horarios de clase con aulas, disponibilidad de los docentes, materias y recursos audiovisuales.

El componente de gestión del proyecto Cronos está caracterizado por una visión más amplia de las posibilidades reales de una organización para resolver determinada situación o arribar a un fin determinado. Y aquí puede asumirse, como la "disposición y organización de los recursos de aulas y materias, disponibilidad de docentes, reserva de aulas, horarios de clases, creación de

ofrecimientos, perfiles de docentes-áreas, perfiles de usuarios escalables, distinción y agrupación de cargas adicionales entre otros".

En todo este desarrollo, la gestión y el control de las cargas académicas se ha ido reforzando bajo una serie de etapas que lo caracterizan hasta llegar a un resultado muy similar al manual, pero con componentes que le permiten una mayor agilidad como: modelos manuales optimizados, control en proyectos, consultas directas y organizadas en todo momento, restricciones automáticas en horarios de clase y proyectos de carga adicional, estadísticas graficas pero sobre todo un modelo de desarrollo que le permite al administrador y al programador una fácil manera de actualización y mejoramiento de módulos, inclusive permitiéndole crear nuevos, además de manejo de perfiles a la medida de la institución en base a su contrato laboral y su oficio diario(Coordinadores, jefes de planeación, decanos, secretarios académicos, estudiantes).

Se presenta entonces un sistema 100% Web, con modelos de seguridad a nivel de servidor, bases de datos y perfiles de usuario que garantizan la integridad de los datos, con menús dinámicos acordes a cada perfil, y sobre todo ser una herramienta útil para el componente organizacional de la Universidad, es un modelo de consulta tanto para el docente como para el estudiante, permitiendo la impresión de horarios, planes de estudio, materias, desde el hogar o la oficina de trabajo.

## 1. DESCRIPCIÓN DEL ÁREA PROBLEMÁTICA

En todas las instituciones de educación superior existe la necesidad de asignar y coordinar los recursos económicos, materiales y humanos en beneficio de los estudiantes y la misma academia; la Universidad de Manizales no es ajena a ello y aún más cuando en los últimos años ha incrementado el número de facultades, programas y estudiantes, generando una mayor cantidad de docentes y grupos por asignatura, haciendo más dispendioso el tiempo para una adecuada asignación de horarios y distribución de carga académica.

Para realizar una adecuada asignación de la carga académica en la Universidad de Manizales y en general en entidades de educación superior, se realiza un modelo general donde se establecen los factores que influyen, así como la relación entre ellos; por ejemplo, es importante determinar el número de grupos que se pretende ofrecer por materia de acuerdo al archivo general de los alumnos y el plan académico correspondiente, el horario disponible de la planta docente ya que servirá para hacer una correcta asignación de este personal

Además es recomendable mantener información actualizada del perfil profesional y las preferencias académicas de los profesores así como el horario disponible para orientar las asignaturas. También es necesario saber si un profesor es de tiempo completo (tc) o medio tiempo (mt) o catedrático y es necesario considerar algunos otros factores que afectan la asignación de carga académica. Algunos factores:

Los factores que influyen en la asignación de grupos por materia en las facultades Universidad de Manizales son:

a) Carga académica por especialidad: Investigación, Ciencias Aplicadas, Ciencias Básicas, Profundizaciones y otras áreas específicas de cada una de las facultades.

b) Número de programas.- Cantidad de especialidades que se imparten en la Universidad.

c) Horario del personal docente.- Las horas disponibles en que el profesor puede estar frente a grupo.

d) Perfil académico del profesor.- Preferencia de la asignación académica según la especialidad del docente.

También se puede observar que en la Universidad de Manizales se manejan diferentes modelos y subsistemas de Asignación de Carga. Cada facultad y dependencia presenta un orden diferente cuando se entra en comparaciones de carga. Es necesario que exista un solo sistema regulatorio y estándar que unifique la Universidad y no muestre a cada entidad, facultad, dependencia o subsistema como islas de información que no logran el objetivo de organización dentro de la Universidad de Manizales, que acompañe necesidades y las resuelva como disponibilidad del docente, administración y organización de aulas y ayudas pedagógicas. Existe un sistema de planeación que es usado en un 35% ya que el objetivo por el cual fue concebido no cumple las labores como asignación de horarios, cómoda asignación y reserva de aulas. Las facultades por su lado no llevan un control centralizado y clasificado de las cargas académicas.

Es necesario un sistema que sea capaz de asignar la carga académica y presente subsistemas que se vinculen a este y además ofrezcan beneficios que cubran estas necesidades y promuevan nuevas formas de organización.

## **2. OBJETIVO GENERAL**

### **2.1 OBJETIVO GENERAL**

Diseñar e implementar un Sistema de Información para la distribución de carga académica y horarios para la Universidad de Manizales.

### **2.2 OBJETIVOS ESPECÍFICOS**

- Diseñar e implementar un módulo capaz de gestionar la asignación de horarios de clase mediante algoritmos de validación y control.
- Diseñar e implementar un módulo para la creación de tipos de carga y la asignación de esta a los docentes y administrativos.
- Diseñar e implementar un módulo para la asignación y reserva de aulas de clase, laboratorios u otros tipos según necesidades específicas de la materia y el usuario.
- Diseñar e implementar un módulo de creación de horarios que brinde la facilidad de evitar cruces entre docentes y clases, validación de horas en la materia y disponibilidad horario del docente a partir de su tipo de contrato y carga asignada
- Diseñar e implementar un módulo de disponibilidad en docentes y administrativos que permita alimentar la creación de horarios en las clases y cargas adicionales.
- Diseñar e implementar un sistema Web centralizado que unifique cada módulo en un solo entorno de interacción y gestión 100% Web.

### 3. ANTECEDENTES

En la actualidad existen algunos acercamientos al problema de asignación de carga académica y horarios en otras universidades, pero no se ha llegado a desarrollar un sistema Web de calidad que trascienda el actual análisis, diseño, desarrollo e implantación de sistemas de información, y que sea utilizado en diversas universidades.

**Prototipo de un sistema de información de los procesos realizado en la admisión y registro académico bajo un modelo intranet – Internet y un motor de base de datos Informix<sup>1</sup>.** El sistema busca diseñar un sistema de registro académico, el cual permita ser utilizado por cualquier número de personas conectadas a una red de información, posee información de docentes, horarios, planes de estudios, entre otros.

**Elementos computacionales para un manejo sistemático de los horarios académico en instituciones de enseñanza superior<sup>2</sup>.** El sistema permite asignar mediante algoritmos heurísticos asignar los horarios académicos, teniendo en cuenta restricciones de docentes, estudiantes y aulas.

**Modelo de asignación de carga académica usando algoritmos genéticos<sup>3</sup>.** Este trabajo presenta un modelo computacional capaz de encontrar la asignación óptima de clases, maestros y horarios utilizando un algoritmo genético.

---

<sup>1</sup> BETANCOURT ESCOBAR, Andrés. Prototipo de un sistema de información de los procesos realizado en la admisión y registro académico bajo un modelo intranet – Internet y un motor de base de datos Informix. Manizales, 1986. 215 p. Tesis de grado (Ingeniero de Sistemas y Telecomunicaciones). Universidad Autónoma de Manizales, Facultad de Ingenierías.

<sup>2</sup> GÓMEZ CORREA, Alejandro. Elementos computacionales para un manejo sistemático de los horarios académico en instituciones de enseñanza superior. Manizales, 1992. 198 p. Tesis de grado. (Ingeniero de Sistemas). Universidad de los Andes, Facultad de Ingeniería.

<sup>3</sup> LÓPEZ TAKEYAS, Bruno. Aplicación de Algoritmos Genéticos para la Asignación de Carga Académica en Instituciones de Educación Superior.[en línea] Nuevo Laredo, 2001. Tesis de grado. Maestría en Computación. (Magíster en computación). Instituto Tecnológico de Nuevo Laredo. Disponible en Internet: URL: [http://www.itnuevolaredo.edu.mx/maestros/sis\\_com/takeyas/](http://www.itnuevolaredo.edu.mx/maestros/sis_com/takeyas/). Tesis de maestría

**HORW<sup>4</sup>**. Sistema de asignación de horarios que funciona bajo la plataforma Windows 32bits; las características más destacadas son:

- Puede elegir, si se desea, los Profesores de cada grupo o sección de alumnos, teniendo en cuenta sus preferencias, con lo que se consigue que éstas se respeten al máximo.
- Minimiza los huecos de profesores, mediante rutinas interactivas de ejecución indefinida, consiguiéndose en la práctica horarios sin huecos.
- Facilidad completa de edición de los resultados, pudiendo modificarse a voluntad los horarios realizados por el ordenador.

**KRONOWIN<sup>5</sup>**. Software generador de Horarios para Windows 98/ME/XP y NT. En la nueva versión Milenio-3, se puede realizar horarios prescindiendo de los Profesores de los cuales desconocemos su nueva incorporación generando un resultado obteniendo así el horario correcto para cada grupo pero haciendo falta de la asignación del Profesor correspondiente en su momento.

**PATHOS**. Programación automática de temas, horarios y salones. La solución implanta el sistema completo de asignación automática de aulas y horarios para la totalidad de la Facultad de Ingeniería de la Universidad Javeriana - Cali y también para la Escuela de Ingeniería Eléctrica de la Universidad del Valle.

Se ideó e implementó asimismo un nuevo esquema de salvaguarda de espacios computacionales para el lenguaje Mozart, que bajó el tiempo de encontrar una solución para toda la facultad, de tres horas a nueve minutos.

**SIU-Guaraní**. El sistema permite mejorar el tratamiento de la información y agilizar los mecanismos de gestión académica. Por otro lado la instalación en las unidades académicas permite homogenizar los procedimientos de las mismas. Los aspectos relevantes de la implementación del sistema son:

Aligerar los trámites académicos evitando la carga burocrática, sin perder el control de los mismos.

---

<sup>4</sup> Horw.exe. Sistema de asignación de horarios que funciona bajo la plataforma Windows 32bits. Sevilla. Fecha de actualización 03.05.06. Disponible en Internet: URL: <http://clientes.vianetworks.es/personal/horarios/>

<sup>5</sup> ADOSSIS SA, Sistemas Informáticos. SIU-Guaraní. Valladolid, 2006. Disponible en Internet: <http://www.adossis.es/index.htm> / enlace KRONOWIN / URL: <http://www.adossis.es/kronowin.htm>

Brindar a las autoridades encargadas de la conducción, herramientas de control sobre el manejo de la gestión académica. También administrará las ofertas de Postgrado de la Universidad.

### **Servicios de la aplicación:**

- Asignación de docentes a cursos y carreras.
- Asignación de aulas.
- Conexión remota vía Internet.
- Terminales de autogestión.
- Registro de inscripción en carreras.
- Registro de equivalencias y exámenes rendidos.
- Manipulación de planes de estudio.

**Sistema de Gestión de Personal (SIU-PAMPA)**<sup>7</sup>. El sistema SIU-Pampa tiene como objetivo principal solucionar la administración de información sobre el personal y facilitar la liquidación a través de un legajo electrónico único por Agente.

El sistema se basa en la idea de mantener un legajo electrónico del empleado que permita conocer sus datos personales, familiares, niveles de estudio, cargos y la historia de los cargos mantenidos por la persona en su relación con la Universidad.

Implementado en la Dirección de Personal de la Administración Central e instalado con recepción de información parcializada en Facultad de Economía y Administración, Ciencias de la Educación e Ingeniería.

**CONTESCO.** Sistema de Control Escolar. CONTESCO es un sistema computacional que ofrece una solución inmediata, completa y de alta eficiencia para colegios y centros educativos que requieran de información al momento sobre el alumnado y la planta docente, asignación de aulas y horarios, control de cobranzas y facturación a alumnos, seguimiento académico de alumnos, retrasos y faltas, generación de reportes. El sistema servidor funciona bajo SO Windows y Linux, el sistema cliente solamente bajo Windows.

**Sistema Sipefco.** Tiene por objeto asegurar el control de manejo de fondos que deben realizar las autoridades de conducción. Sus aplicaciones brindan al sector contable, presupuestario y financiero una herramienta apta para el procesamiento del flujo de fondos, su incidencia presupuestaria y su expresión contable por centros de costos responsables, lo que permitirá en un futuro relacionar el valor de

---

<sup>7</sup> SIU-PAMPA. Sistema de Gestión de Personal. Buenos Aires, 1990. Universidad Nacional del Comahue. Disponible en Internet: URL: <http://www.uncoma.edu.ar/dti/presentacion.html>

los recursos que se utilizan a los resultados educativos, aligerar los trámites administrativos, evitando la carga burocrática.

**Modelo de asignación de carga académica mediante la aplicación de técnicas meta-heurísticas<sup>8</sup>.** El proceso de asignación de carga académica en una institución educativa consiste en la asignación de cursos, salones, estudiantes y profesores en unos espacios de tiempo preestablecidos, de forma tal que no se presenten inconsistencias como dos cursos en un mismo salón a la misma hora o un profesor con dos clases al mismo tiempo. Este proceso implica una alta complejidad debido a los diferentes aspectos o factores que se deben tener en cuenta al momento de generar la programación de una unidad académica. De lo anterior se ha concluido que este problema es combinatorial y es conocido en la literatura especializada como un problema del tipo “NP-HARD COMPLETE”.

**Software de planeación Universidad de Manizales<sup>9</sup>.** Software que funciona bajo sistemas operativos Windows, fue desarrollado con el manejador de bases de datos FoxPro, este software permite asignaciones manual, semiautomática, automática de aulas y horarios de clase, además permite el manejo de docentes, edificios y facultades.

---

<sup>8</sup> RAMÍREZ OSORIO, Carlos; GAVILÁN, León. Modelo de asignación de carga académica mediante la aplicación de técnicas meta-heurísticas. Bucaramanga, 1999 56 p. Tesis de Grado (Ingeniero de Sistemas). Universidad Industrial de Santander. Facultad de Ingeniería.

<sup>9</sup> GONZÁLEZ, Albeiro. Software de Planeación de Aulas. Manizales, 1994. Universidad de Manizales, Oficina de planeación.

## 4. METODOLOGÍA

- Análisis del proceso actual de asignación de carga académica y horarios en la Facultad de Ingeniería.
- Análisis de los procesos en las otras facultades de la Universidad de Manizales. Se realizó una entrevista (Anexo A).
- Diseño de un modelo mediante metodología OMT y la herramienta de modelado UML que represente los procesos realizados para la asignación de carga académica y horarios.
- Selección de las herramientas de software libre a utilizar para la construcción del sistema.
- Implementación del sistema en un servidor asignado por la Universidad de Manizales.
- Ajustes de servidores Web y bases de datos
- Ingreso de información válida de las facultades como prueba del sistema
- Evaluación y validación de la asignación de carga académica y horarios.
- Corrección de errores durante el proceso de ingreso de información por medio Web.
- Análisis de documentación en algoritmos genéticos para resolución de problemas tipo N-P Complejos par la solución de problemas de planeación de tiempo (timetabling) para versiones futuras de Cronos
- Informe Final

## 5. JUSTIFICACIÓN

La expansión de los programas académicos de la Universidad de Manizales, ha generado un mayor flujo y volumen de datos en cada una de sus facultades, que a su vez involucra procesos de gestión y administración más complejos.

Uno de estos procesos de gestión es la asignación de la carga académica y horarios de clase en el inicio de cada semestre, los cuales se convierten en procesos dispendiosos en tiempo y recurso humano (Anexo R.), gastando en muchas facultades varios días, algunas de ellas meses para su adecuada preparación. Citando a la facultad de Ingeniería que siendo una de las más eficientes, se demora un mes y medio en esta preparación, y facultades como la de Psicología dos meses siendo este trabajo realizado por profesionales que deberían ocupar su tiempo en labores adecuadas para las cuales fueron contratados por la Universidad de Manizales, pudiendo éste ser utilizado en la atención de problemas de los estudiantes o docentes en cada una de las secretarías académicas y decanatos.

Debido a éstos inconvenientes de tiempo y recurso humano, se pretende dar una solución informática y de comunicaciones que cambiaría los procesos manuales a otros donde se utilizaría información centralizada en una base de datos y redes de información compartida mediante Internet y la red.

## **6. MARCO TEÓRICO**

El objeto de estudio en esta tesis es el manejo de la información para la asignación de carga académica y horarios, que se debe enmarcar en una constante evolución con desarrollos informáticos que le permitan asimilar e imitar los procesos físicos cotidianos en procesos automatizados, capaces de imitar las formas manuales, pero con unas grandes diferencias muy bien enmarcadas entre lo automático y lo manual, ellas son: velocidad, ahorro de tiempo, comodidad, seguridad, escalabilidad y portabilidad.

Durante varios años los sistemas de ASIGNACION DE CARAGA ACADEMICA Y HORARIOS (ACA), son operados de forma manual. Están llenos de variables y restricciones que dificultan su desarrollo formal, donde se ven invertidos grandes espacios de tiempo y control, pero muchas veces por no definir inicialmente un buen planteamiento de distribución de variables en el caso de horarios, aulas y docentes, generan grandes vacíos y mala distribución de espacio y tiempo, causando inconvenientes en el momento de la planeación y asignación de recursos.

Las observaciones que se pueden entregar para la situación actual se han tomado de los antecedentes mostrados en este documento y algunas soluciones vanguardistas de buenos y malos resultados.

### **6.1 FACTORES DE COMPLEJIDAD Y DE VARIEDAD DE SOLUCIÓN**

A nivel informático hay momentos donde es tanta la información y tecnología que se presenta al dar una solución, que se debe tomar gran parte del tiempo para investigar que opciones podrían ser óptimas, al final no resulta tan bueno, pues cuando se ha tomado la decisión, puede haber una nueva solución en el mercado.

El anterior antecedente podría ilustrarse con el siguiente ejemplo: ¿desarrollar una aplicación únicamente cliente?, o ¿desarrollar una aplicación cliente-servidor?, ¿software libre, o software propietario?, que pasara si se elige Software libre?, ¿se ganara en recursos económicos o se ganara tiempo si se invierte en Software propietario, así que se entraría en un debate de gran interés y de actualidad y que es necesario abordarlo por variables de tiempo, costo-beneficio, y soporte y otros cuantos que se pueden escapar.

En la actualidad hay varias aplicaciones cliente en su mayoría con software propietario, con una muy buena estructura de desarrollo y manejo, que da una buena solución en el tiempo en que fue desarrollado(mas de 8 años), pero servirá

hoy donde la exigencia es conectividad? Acceso global?, o la que nosotros desarrollemos servirá a futuro?

Es necesario crear aplicaciones que no se centren en una solución actual si no en una duradera y que se atreva a indagar en el futuro, soluciones que sea cual sea el desarrollo se prospecten

## **6.2 EL SEGUNDO FACTOR EL ALGORITMO**

Una de la partes mas interesantes en la solución de este proyecto son los algoritmos que deben ser tratados, que dan solución a los problemas de planeación de tiempo y espacio, Para la asignación de horarios a los ofrecimientos, se tienen 3 procesos generales para el desarrollo del algoritmo que intervienen para consulta, inserción y borrado, pero que son analizados independientemente para su clara interpretación. Es importante tener en cuenta para el desarrollo Web, se realiza una consulta inicial de información de la base de datos, los datos son llevados a matrices y al finalizar el proceso son almacenados en la base de datos, teniendo en cuenta que en este proceso de asignación se llevan a cabo unos bloqueos que impiden la alteración de los datos fuentes mientras la información se encuentra en memoria volátil. De igual forma se debe tener en cuenta que al realizar modificación en los datos, se realiza un refresco en la página Web y en las matrices de información, lo cual implica modificaciones instantáneas.

## **6.3 ACCESIBILIDAD Y MOVILIDAD**

En nuestro contexto no hay una manera definitiva de lograr un desarrollo que demuestre que es el final y la manera de hacerlo, pues constantemente recibimos cambios, leyendo esta misma tesis y las anteriores se muestran muchas renovaciones en cuanto a otras tesis, y también se encuentran otras mas novedosas, al terminar esta tesis, en otro lugar y otros estudiantes estarán renovando las mas actualizadas, y tomando como base esta y otras cuantas mas, lo cierto es que en la actualidad las aplicaciones están siendo migradas hacia la red, el Internet incluso los celulares, por eso la necesidad de desarrollar aplicaciones no solo clientes si no aplicaciones donde no necesites un programa de instalación y llevar el PC, el Portátil o la PALM a todo lugar, sino un simple acceso a Internet, esta es talvez una de las grandes ventajas que presenta esta tesis en cuanto a sus predecesoras a nivel internacional, pues la mayoría son aplicaciones que aun teniendo agradables modelos heurísticos fueron programados para ser instalados.

En el problema del ACA y horarios es también indispensable tener controles de gestión que involucren los horarios disponibles de los diferentes tipos de docentes y su tipo de contratación, pues no todos contarán con la misma carga de horas, algunos 20 horas semanales, otros más, otros menos, materias que igualmente no serán impartidas por el mismo docente varias veces al día o consecutivamente en la semana.

Se deben plantear modelos informáticos que involucren la disponibilidad del docente por su tipo de vinculación laboral, y que los horarios de clase no sean los únicos que usen esta disponibilidad, si no también las otras cargas académicas diferentes a las clases de manera que se tenga un control general en horarios y en docentes, que permitan desarrollar una gestión más completa en agendas, disposición de tiempo y determinar en que lugar y en que momento estarán los empleados de la institución, se devela así una oportunidad de control y gestión a la hora de mejorar procesos académicos y empresariales.

#### **6.4 LOS SISTEMAS DE INFORMACIÓN**

Un Sistema de Información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa, es decir, “un conjunto formal de procesos que, operando sobre una colección de datos estructurada según las necesidades de la empresa, recopilan, elaboran y distribuyen la información o parte de ella necesaria para las operaciones de dicha empresa y para las actividades de dirección y control correspondientes para desempeñar sus actividades de acuerdo a su estrategia de negocio”.<sup>10</sup>

#### **6.5 LOS ALGORITMOS**

Un algoritmo, nombre que proviene del matemático persa del siglo IX al-Khowârizmî, “es sencillamente un conjunto de reglas para efectuar algún cálculo bien sea a mano o más frecuentemente en una máquina”<sup>11</sup>. De esta manera pues y ampliando un poco esta definición los algoritmos son una serie de pasos y reglas que nos deben llevar a una solución correcta de un determinado problema

---

<sup>10</sup> ANDREW, et. al. Concepto de sistemas de información. [en línea]. Costa Rica, fecha de actualización 1999. Preguntas más frecuentes (<http://www.sipromicro.com/modules.php>). / sistemas de información sobre la microempresa en América Central / preguntas frecuentes / conceptos generales / Concepto de sistemas de información. Disponible en : <http://www.sipromicro.com/modules.php?op=modload&name=Search&file=index>

<sup>11</sup> BRASARD, G. Fundamentos de algoritmia. Madrid : Prentice Hall, 1999. p. 50

## 6.6 LOS ALGORITMOS GENÉTICOS

Los algoritmos genéticos son métodos sistemáticos para la resolución de problemas de búsqueda y optimización que aplican a estos los mismos métodos de la evolución biológica: selección basada en la población, reproducción sexual y mutación.

## 6.7 DESCRIPCION DE METODOLOGIAS DE ANALISIS Y DESARROLLO

**6.7.1 Técnica de Modelado de Objetos (Object Modeling Technique OMT).** El Modelado y Diseño Orientado a Objetos es una técnica que se fundamenta en pensar acerca de problemas a resolver, empleando modelos que se han organizado tomando como base conceptos del mundo real.

**OMT** es una metodología (y una notación gráfica) para el desarrollo orientado a objetos que consiste en “construir un modelo de un dominio de aplicación añadiéndole detalles durante el diseño del sistemas”<sup>12</sup>.

**6.7.2 Ciclo de vida.** Tomando como referencia lo descrito por Rumbaugh<sup>13</sup>, el ciclo de vida de un desarrollo de software se visualiza de la siguiente forma:

**Análisis:** El analista describe el problema y plantea un modelo de la situación del mundo real para mostrar sus propiedades importantes. El analista debe trabajar con los usuarios finales que son quienes comprenden y evitar presentar soluciones incompletas. Los modelos generados en la fase del análisis son una abstracción resumida y precisa de lo que debe hacer el sistema deseado y no de la forma en que se hará.

**Diseño del sistema:** Los diseñadores de sistemas toman decisiones acerca de la arquitectura a utilizar. Durante el diseño, el sistema a desarrollar se organiza en subsistemas basados tanto en la estructura del análisis como en la arquitectura planteada.

Esta metodología se extiende desde el análisis hasta la implementación, pasando por el diseño. En primer lugar, se construye un modelo de análisis para abstraer los aspectos esenciales del dominio de la aplicación sin tener en cuenta la implementación eventual. En este modelo se toman decisiones importantes que después se completan para optimizar la implementación, en segundo lugar.

---

<sup>12</sup> RUMBAUGH, James; BLAHA, Michael; PREMERLANI, William y FREDERICK, Eddy. Modelado y Diseño Orientado a Objetos. Madrid : Prentice Hall, 1991. p. 24.

<sup>13</sup> *Ibíd.* p. 100.

Los objetos de la aplicación constituyen el marco de trabajo del modelo de diseño, pero se implementan en términos de objetos del dominio de los recursos tecnológicos. Por último, el modelo de diseño se implementa en algún lenguaje de programación, base de datos o hardware.

La metodología OMT emplea tres clases de modelos para describir el sistema, los cuales “no son igualmente importantes en todos los problemas”.<sup>14</sup>

- El Modelo de Objetos: Describe los objetos del sistema y sus relaciones.
- El Modelo Dinámico: Describe las interacciones existentes entre objetos del sistema.
- El Modelo Funcional: Describe las transformaciones de datos del sistema.

OMT es una de las metodologías de análisis y diseño orientadas a objetos, más maduras y eficientes que existen en la actualidad. La gran virtud que aporta esta metodología es su carácter de abierta (no propietaria), que le permite ser de dominio público y, en consecuencia, sobrevivir con enorme vitalidad. Esto facilita su evolución para acoplarse a todas las necesidades actuales y futuras de la ingeniería de software.

Una metodología de desarrollo de software OO consta de los siguientes elementos:

- Conceptos y diagramas ( Modelo)
- Etapas y definición de entrega en cada una de ellas.
- Actividades y recomendaciones.

#### **Etapas:**

- Análisis de requerimientos.
- Diseño del sistema.
- Diseño detallado.
- Implementación y pruebas.

---

<sup>14</sup> Ibid., p. 204.

**Análisis de requerimientos.** En esta etapa se busca las necesidades del usuario y la forma que se va a presentar la solución.

Las actividades contempladas son: - Identificar los casos de uso del Sistema, - Construir el Diagrama de casos de usos, - Desarrollar el modelo del mundo, - Identificar clases (Atributos, Operaciones, Relaciones, Cardinalidad) y - Validar los modelos (con el cliente : clases, atributos, operaciones y crear diagrama de secuencia o de colaboración)

**Diseño del sistema.** En esta etapa se define una subdivisión en aplicaciones del sistema y la forma de comunicación con los sistemas existentes con los que debe interactuar.

Las actividades contempladas son: -Identificar la arquitectura del sistema, - Definir componentes del sistema, -Refinar los casos de uso aplicados al software.

**Diseño detallado.** En esta etapa se adecua el análisis a las características específicas de ambiente de implementación.

Las actividades contempladas son: Agregar los detalles de implementación del modelo del mundo. - Completar los detalles de la clase ( diagrama de clases ) , - Subdividir en paquetes ( diagrama de paquetes ). -Desarrollar el modelo de interfaz, - Conocer el ambiente de base. - Enlazar las clases de interfaz con el modelo del mundo, - Crear diagrama de interacción.

**6.7.3 UML (Unified Modeling Language).** A lo largo de los años el software se ha vuelto cada vez más complejo. El incremento en la potencia de los ordenadores, la aparición de modernos dispositivos y tecnologías ha generado la necesidad de nuevos sistemas operativos y aplicaciones de mayor complejidad. Simultáneamente, esta complejidad ha traído consigo la demanda de nuevas técnicas y herramientas que faciliten la construcción de este software.

En este contexto, UML surge como respuesta al problema reseñado para contar con un lenguaje estándar para crear diagramas de diseño de software.

UML es un lenguaje de modelado que se usa para especificar, visualizar, construir y documentar los elementos que forman un sistema software orientado a objetos. Es utilizado para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.

Igualmente, permite captar la información sobre la estructura estática y el comportamiento dinámico de un sistema, además ofrece una abstracción del sistema y sus componentes. Pretende unificar experiencias pasadas sobre

técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar. Este lenguaje ha sido concebido por los autores de las tres metodologías más usadas para la orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh

La notación ha sido ampliamente aceptada debido al prestigio de sus creadores y debido a que incorpora las principales ventajas de cada uno de las metodologías particulares en los que se basa: Booch, OMT y OOSE. Con UML se fusiona la notación de estas técnicas para formar una herramienta compartida entre todos los ingenieros de software que trabajan en el desarrollo orientado a objetos.

UML no es un lenguaje de programación, es un lenguaje de propósito general para el modelado orientado a objetos.

**Diagrama de casos de uso.** Los casos de uso es una técnica para capturar información de cómo un sistema o negocio trabaja, o de cómo se desea que trabaje, describen bajo la forma de acciones y reacciones el comportamiento de un sistema desde el punto de vista del usuario.

**Diagrama de clases.** El mismo autor, presenta los elementos mas estables del sistema, estos son los dispositivos de control de datos e información, correspondientes a las clases del sistema, con sus relaciones estructurales y de herencia. El modelo de casos de uso aporta información para establecer las clases, objetos, atributos y operaciones.

**Diagrama de estados.** Es usado para describir el comportamiento de las instancias y elementos de un modelo. Específicamente describen las posibles secuencias de estados y acciones a través de las cuales las instancias puedes proceder durante su ciclo de vida, como resultado de eventos discretos, tales como señales externas y, mensajes.

**Diagrama de actividad.** Es un caso especial del diagrama de estados, este puede especificar, el comportamiento de los objetos de una clase, la lógica de una operación (método), y parte o toda la descripción de un caso de uso.

**Diagrama de secuencia.** Presenta una interacción, la cual es un conjunto de mensajes entre un conjunto de instancias interactuando, estas son un grupo de estímulos entre instancias con el efecto de determinar el funcionamiento deseado de un proceso o un resultado.

**Diagrama de colaboración.** Modela la interacción entre los objetos de un Caso de Uso, estos están conectados por enlaces en los cuales se representan los mensajes enviados acompañados de una flecha que indica su dirección, el

diagrama ofrece una mejor visión del escenario cuando el analista está intentando comprender la participación de un objeto en el sistema.

**Diagrama de paquetes.** Ofrecen un mecanismo general para la organización de los modelos/subsistemas agrupando elementos de modelado, esta agrupación se realiza por operaciones comunes o por divisiones de la organización.

**Diagrama de componentes.** Muestra las dependencias entre los diferentes componentes de software, incluyendo las clasificaciones que se puedan realizar, estas implementaciones son de diferentes tipos entre ellas se encuentra archivos de código fuente, archivos binarios, archivos ejecutables, scripts, entre otros.

**Diagrama de despliegue.** modela la distribución en tiempo de ejecución de los elementos de procesamiento y componentes de software, junto a los procesos y objetos asociados. “Muestra las relaciones físicas entre los componentes software y hardware asociados, en el desempeño del sistema”<sup>15</sup>.

---

<sup>15</sup> FOWLER, citado por GIRALDO R., Juan Pablo. Análisis y Diseño Orientado a Objetos : Aplicando OMT (Object Modelling Technique) con UML(Unified Modelling Lenguaje). Inédito. Universidad de Manizales, Manizales, Colombia. p. 8-9.

## 7. RESULTADOS

El sistema de carga académica Cronos ha logrado garantizar la coherencia, la continuidad y la homogeneidad de las políticas académicas en las que se involucra todos los procesos de gestión de asignación de carga académica.

El sistema de carga académica Cronos fortalece y desarrolla una estructura de gestión académico-administrativa congruente con la cobertura y las dimensiones del nuevo momento de la Universidad como Universidad que busca ser acreditada y garantiza el desempeño eficiente de tareas estructurales tales como el seguimiento personalizado a docentes; la articulación de las cargas adicionales con la disponibilidad de los académicos; la gestión de los apoyos académicos para asistencia a eventos y prácticas de los estudiantes;

Se ha logrado desarrollar un sistema Web que asista la creación de horarios por parte de los encargados, con operaciones de bloqueos y validaciones en docentes, materias, horas, áreas de docentes y materias a la hora de crear o modificar horarios, integrando también datos como disponibilidad del docente.

El sistema permite crear las disponibilidades de los docentes para una mayor efectividad en el momento de planear una carga académica y hacer seguimiento a los empleados.

Cronos provee también algunas estadísticas para el apoyo a toma de decisiones y auditoría en proyectos tales como: Estadísticas gráficas por facultades y generales en:

- Tipos de contratación docente por Contrato
- Desarrollo Académico en Docentes por escalafón

Además provee una tabla de gestión y control general de carga donde muestra la situación actual del docente integrando número de horas asignadas por cada uno de los tipos de proyectos de facultades, bonificaciones, tipo de contratación y cuadros de error cuando el tope de horas respecto al contrato es inferior.

Control de Proyectos:

- Clasificación de proyectos y cargas adicionales
- Horas por proyecto
- Numero de docentes por proyecto

- Horas asignadas a docentes
- Horario de docentes por proyecto y ofrecimientos.

Con respecto a aulas, Cronos provee un subsistema de reserva de aulas, y asignación de aulas a horarios, que brinda además un conocimiento de ubicación, capacidad y recursos audiovisuales de los que dispone.

## 8. CONCLUSIONES

- Las formas tradicionales del ACA\* en las diferentes facultades de la Universidad han sido fundamentales para la operación general de planeación y gestión académica, sin embargo, ello ha requerido su reorganización por su diversidad de planeación a fin de atender con eficacia la validez de los procesos que se generan a raíz de la misma tales como: La oferta educativa y su vinculación con el medio laboral, la docencia y su relación con la investigación y la extensión.
- Es necesario fortalecer el espacio de gestión académica-administrativa para garantizar de un modo eficiente las tareas de enlace entre las entidades participantes (facultades-oficina de plantación-registro académico y otros entes administrativos). Esta necesidad se ha suplido en momentos en que se desarrolla la planeación de carga académica con el nuevo sistema (Cronos), pero el análisis entrega que otros procesos diferentes necesitan una mayor homogeneidad e interacción de procesos.
- Los principales cambios derivados de la asignación de carga académica que se llevaba tradicionalmente con respecto a la entregada Cronos han pretendido asegurar el logro de los objetivos de las diferentes facultades y la Universidad de Manizales y garantizar el aprovechamiento óptimo de los apoyos diferenciales proporcionados en ésta como disponibilidad de empleados, reserva de aulas, creación y consulta de recursos en aulas, recursos en materias, creación y clasificación de proyectos, horarios docentes por proyecto etc.

---

\* ACA: Asignación de carga académica

## 9. RECOMENDACIONES

- Proyectos de grado que hagan parte de procesos de desarrollo académico o administrativo para la universidad de Manizales como es el caso de Cronos y otros tantos, no deben quedar se en la versión en la cual fueron entregados, sino ser parte de procesos permanentes de actualización, siendo abordados por otros estudiantes o ingenieros que apoyen estos procesos de cambio a nivel de Software, y de esta manera lograr que el producto sea competitivo y que realmente cumpla el objetivo y servicio por el cual fue creado.
- Dentro de los documentos anexos, se encuentra “**ALGORITMOS GENÉTICOS EN PROBLEMAS DE COMPLEJIDAD N-P**”, que es un artículo de autoría propia que da un punto de partida sobre algoritmos genéticos, esto con el fin que para futuras versiones del sistema CRONOS puedan involucrarse algoritmos de este tipo en sus procesos ya que son los ideales o por lo menos más tratados al momento de asignar horarios o crear agendas en la actualidad.
- El sistema Cronos tiene un modelo de reserva de aulas, el cual podría ser tomado para nuevas versiones como modelo para un nuevo módulo encargado de las reservas de equipos de la dependencia AUDIOVISUALES, además que el sistema cronos posee un sistema de creación de recursos académicos que podría acompañar esta labor.
- Es necesario uno o dos administradores del sistema que le brinden mantenimiento y apoyo técnico a los usuarios.

## BIBLIOGRAFÍA

ADOSSIS SA, Sistemas Informáticos. SIU-Guaraní. Valladolid, 2006. Disponible en Internet: <http://www.adossis.es/index.htm> / enlace KRONOWIN / URL: <http://www.adossis.es/kronowin.htm>

ANDREW, et. al. Concepto de sistemas de información. [en línea]. Costa Rica, fecha de actualización 1999. Preguntas más frecuentes (<http://www.sipromicro.com/modules.php>). / sistemas de información sobre la microempresa en America Central / preguntas frecuentes / conceptos generales / Concepto de sistemas de información.

BETANCOURT ESCOBAR, Andrés. Prototipo de un sistema de información de los procesos realizado en la admisión y registro académico bajo un modelo intranet – Internet y un motor de base de datos Informix. Manizales, 1986. 215 p. Tesis de grado (Ingeniero de Sistemas y Telecomunicaciones). Universidad Autónoma de Manizales, Facultad de Ingenierías.

BRASARD, G. Fundamentos de algoritmia. Madrid : Prentice Hall, 1999. p.212

BRASSARD, G. Fundamentos de Algoritmia. Montreal : Prentice Hall, 1997. 460 p.

CRAIG, Larman. UML y Patrones. Introducción al análisis y diseño orientado a objetos. California: Prentice Hall. 1999. 230 p.

Disponible en :

<http://www.sipromicro.com/modules.php?op=modload&name=Search&file=index>

FANGO, Hsiao-Lang Genetic Algorithms in timetabling and scheduling, department of artificial intelligence. Edinburgh: University of Edinburgh 1994. 224 p.

FOWLER, citado por GIRALDO R., Juan Pablo. Análisis y Diseño Orientado a Objetos : Aplicando OMT (Object Modelling Technique) con UML(Unified Modelling Lenguaje). Inédito. Universidad de Manizales, Manizales, Colombia. p. 36.

GIRALDO, Juan Pablo. Modelo Orientado a objetos OMT + UML, 31 de marzo 2005. Facultad de Ingeniería, Universidad de Manizales (material de trabajo)

GÓMEZ CORREA, Alejandro. Elementos computacionales para un manejo sistemático de los horarios académico en instituciones de enseñanza superior. Manizales, 1992. 198 p. Tesis de grado. (Ingeniero de Sistemas). Universidad de los Andes, Facultad de Ingeniería.

GONZÁLEZ, Albeiro. Software de Planeación de Aulas. Manizales, 1994. Universidad de Manizales, Oficina de planeación.

Horw.exe. Sistema de asignación de horarios que funciona bajo la plataforma Windows 32bits. Sevilla. Fecha de actualización 03.05.06. Disponible en Internet: URL: <http://clientes.vianetworks.es/personal/horarios/>

JACOBSON, Iván; BOOCH, Grady y RUMBAUGH James. El proceso unificado del desarrollo del software. California: Pearson Education, 2000, 257p.

LÓPEZ TAKEYAS, Bruno. Aplicación de Algoritmos Genéticos para la Asignación de Carga Académica en Instituciones de Educación Superior.[en línea] Nuevo Laredo, 2001. Tesis de grado. Maestría en Computación. (Magíster en computación). Instituto Tecnológico de Nuevo Laredo. Disponible en Internet: URL: [http://www.itnuevolaredo.edu.mx/maestros/sis\\_com/takeyas/](http://www.itnuevolaredo.edu.mx/maestros/sis_com/takeyas/). Tesis de maestría.

PRESSMAN, Roger. Ingeniería del Software un Enfoque Práctico. 5 ed. Madrid : McGraw-Hill, 2002. 601 p.

QUINTERO, Jorge Andrés; VARGAS, Carlos Alberto. Línea de investigación y desarrollo en Inteligencia Artificial. En: Revista Ventana Informática. No. 9 (oct., 2001). p. 45 – 57. Manizales: Universidad de Manizales.

RAMÍREZ OSORIO, Carlos; GAVILÁN, León. Modelo de asignación de carga académica mediante la aplicación de técnicas meta-heurísticas. Bucaramanga, 1999 56 p. Tesis de Grado (Ingeniero de Sistemas). Universidad Industrial de Santander. Facultad de Ingeniería.

RUMBAUGH, James; BLAHA, Michael; PREMERLANI, William y FREDERICK, Eddy. Modelado y Diseño Orientado a Objetos. Madrid : Prentice Hall, 1991. p. 440.

SENN, James. Análisis y Diseño de Sistemas de Información. 2 ed. México, McGraw-Hill, 1992. 942 p.

SIU-PAMPA. Sistema de Gestión de Personal. Buenos Aires, 1990. Universidad Nacional del Comahue. Disponible en Internet: <http://www.uncoma.edu.ar/dti/presentacion.html>

Referencia Web que presenta manuales para el acercamiento al modelado de objetos

[http://es.wikipedia.org/wiki/Lenguaje\\_Unificado\\_de\\_Modelado](http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado)

Referencia web que presenta como hacer backups de bases de datos en el motor de bases de datos PostgreSQL

<http://es.tldp.org/Postgresql-es/web/navegable/user/app-pgdump.html>

Referencia web que presenta ayudas en desarrollos de Webs y código HTML

<http://www.desarrolloweb.com/articulos/1469.php?manual=6>

Referencia web que nos da un vistazo general sobre los Algoritmos Genéticos

<http://ai.bpa.arizona.edu/~mramsey/ga.html>

## **ANEXOS**

## ANEXO A. COMPARACIÓN MYSQL Y POSTGRES

### POSTGRESQL VS. MYSQL<sup>5</sup>

#### POSTGRESQL

- Postgres es un sistema de bases de datos de mayor nivel, al nivel Oracle, Sybase o Interbase.
- Consume bastantes recursos y carga más el sistema.
- Soporta transacciones y desde la versión 7.0, llaves foráneas (integridad referencial).
- Soporta un subconjunto de SQL92 MAYOR que el que soporta MySQL.

#### MYSQL

- Su principal objetivo de diseño fue la VELOCIDAD.
- Consume muy pocos recursos, tanto de CPU como de memoria.
- No considera las llaves foráneas. Ignora la integridad referencial, dejándola en manos del programador de la aplicación.
- Se comporta mejor que Postgres a la hora de modificar o añadir campos a una tabla "en caliente".

---

<sup>5</sup> Postgres Vs Mysql. Disponible en Internet: <URL:  
[http://glud.udistrital.edu.co/areas/doc/charlas/7\\_introduccion\\_pgsql/pgsql-intro/foil03.html](http://glud.udistrital.edu.co/areas/doc/charlas/7_introduccion_pgsql/pgsql-intro/foil03.html)>

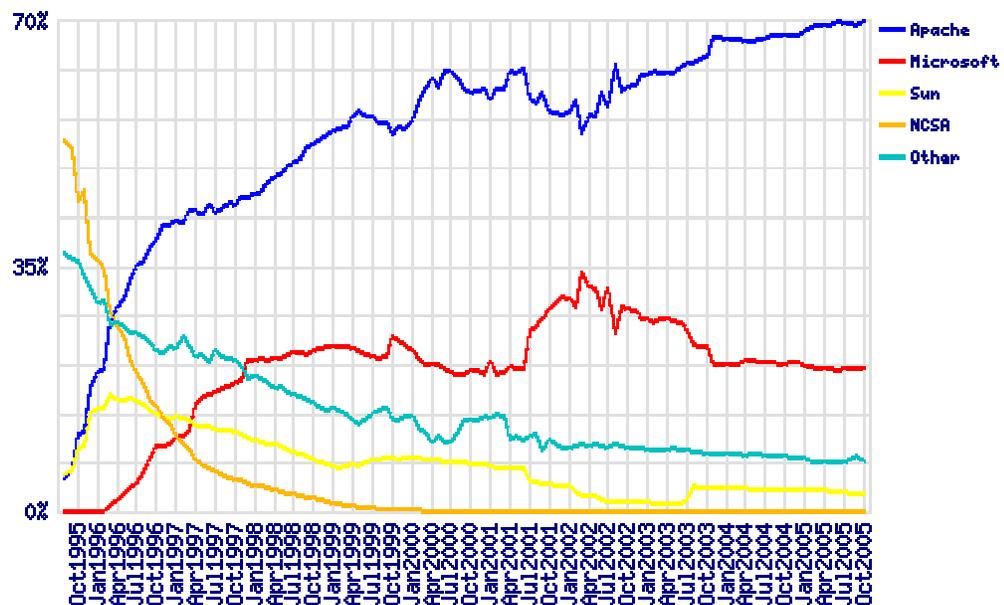
## ANEXO B. COMPARACIÓN DE SERVIDORES WEB

### SERVIDORES

La grafica 1 muestra la cuota de mercado para los servidores superiores a través de todos los dominios agosto de 1995 - octubre de 2005

#### Top Desarrolladores

Desarrollador	Septiembre 2005	Porcentaje	Octubre 2005	Porcentaje	Cambio
Apache	49598424	69.15	52005811	69.89	0.74
Microsoft	14601553	20.36	15293030	20.55	0.19
Sun	1868891	2.61	1889989	2.54	-0.07
Zeus	584598	0.82	585972	0.79	-0.03



## ANEXO C. ALGORITMOS GENÉTICOS

### ALGORITMOS GENÉTICOS EN PROBLEMAS DE COMPLEJIDAD N-P

#### Resumen:

Este anexo pretende mostrar y definir algunos problemas a los que se pueden aplicar los algoritmos evolutivos específicamente los algoritmos genéticos. Por eso parece conveniente, antes que nada, describir el tipo de problemas a los que nos podremos enfrentar, la anatomía de los algoritmos genéticos, un ejemplo aplicado a la resolución de este tipo de problemas y algunas conclusiones de este y otros proyectos que involucran desarrollos con AG.

David Agudelo & Juan Carlos Cortes

#### INTRODUCCIÓN

En realidad, los algoritmos de búsqueda abarcan prácticamente todo algoritmo para resolver problemas automáticamente. habitualmente, en informática se habla de búsqueda cuando hay que hallar información, siguiendo un determinado criterio, dentro de un conjunto de datos almacenados; sin embargo, aquí nos referiremos a otro tipo de algoritmos de búsqueda, a saber, aquellos que, dado el espacio de todas las posibles soluciones a un problema, y partiendo de una solución inicial, son capaces de encontrar la solución mejor o la única, mas específicamente estos problemas son muy complejos y comúnmente llamados np-completos (aquellos cuya complejidad crece con el tamaño del problema de forma exponencial (problema del viajante, horarios, agendas etc.) que no pueden ser abordados de forma tradicional(algoritmos recursivos o de tipo voraz (*greedy*)).

#### ¿Donde aplicar algoritmos genéticos?

Generalmente, los problemas de optimización son tratados por la rama de las matemáticas denominada investigación operacional, aunque prácticamente todas las ramas de la ciencia y la ingeniería necesitan tratar con problemas de optimización en algún momento. Por ejemplo, en *teoría de juegos* se trata de maximizar la probabilidad de ganar, y en *reconocimiento de patrones* de minimizar el error de clasificación de un patrón desconocido (como una imagen de satélite digitalizada, o un canal procesado de una señal de un electroencefalograma).

En control industrial se plantean también problemas de optimización: como mantener el funcionamiento de una máquina dentro de su régimen óptimo, por ejemplo. Cada máquina suele tener una serie de parámetros variables, y lo que se desea optimizar es habitualmente la calidad del producto final o la rapidez a la hora de producirlo.

Para los ejemplos anteriores en algunos casos, la función de evaluación para dar una buena solución ni siquiera existe (cuando hablamos de función (f) hablamos de ecuaciones matemáticas propias, que dan soluciones en el momento de modelar un algoritmo de optimización). En algunos otros casos se trata de optimizar  $f(c)$ , donde  $c$  es una combinación de diferentes elementos variables que pueden tomar un número finito de valores; pueden ser combinaciones con o sin repetición, o incluso permutaciones, como en el caso del problema del viajante; (en este caso se denominan problemas de optimización combinatoria).

No siempre, el espacio de búsqueda completo contiene soluciones válidas; en algunos casos, los valores de las variables se sitúan dentro de un rango, más allá del cual la solución es inválida. Se trata entonces de un problema de optimización con restricciones. En este caso, el problema consiste en maximizar  $f(x_i)$  dentro de un subespacio. Un ejemplo de este problema es el de optimización de los horarios de clase de una institución de enseñanza; hay que disponerla de forma que un profesor no deba estar en dos sitios a la vez (un alumno, puede), que el número de horas libres entre clases sea mínimo, y que se cumplan las preferencias de todos los implicados (catedráticos, tiempo completo, áreas, materias, días, horas, etc.). En este caso, la optimización se reduce a cumplir todas las restricciones (este será una parte como ejemplo abstrayendo algunas partes de la tesis de grado CRONOS para la universidad de Manizales).

Otros ejemplos son muy comunes y necesitan de AG como lo es la programación un router en telecomunicaciones

## **ALGORITMOS GENÉTICOS**

Anatomía de un algoritmo genético

### **¿Que es un AG?**

Los algoritmos genéticos son métodos sistemáticos para la resolución de problemas de búsqueda y optimización que aplican a estos los mismos métodos de la evolución biológica: selección basada en la población, reproducción sexual y mutación.

Los algoritmos genéticos son métodos de optimización, que tratan de resolver el mismo conjunto de problemas que se ha contemplado anteriormente, es decir,

hallar  $(x_1, \dots, x_n)$  tales que  $f(x_1, \dots, x_n)$  sea máximo (mejor solución posible). En un algoritmo genético, tras parametrizar el problema en una serie de variables,  $(x_1, \dots, x_n)$  se codifican en un cromosoma. Todos los operadores utilizados por un algoritmo genético se aplicarán sobre estos cromosomas, o sobre poblaciones de ellos. En el algoritmo genético va implícito el método para resolver el problema; son solo parámetros de tal método los que están codificados. Hay que tener en cuenta que un algoritmo genético es independiente del problema, lo cual lo hace un algoritmo robusto, por ser útil para cualquier problema, pero a la vez débil, pues no está especializado en ninguno.

### **Anatomía:**

Un algoritmo genético consiste en lo siguiente: hallar de qué parámetros depende el problema, codificarlos en un cromosoma, a estos se les aplican los métodos de la evolución (para nosotros operadores): selección y reproducción sexual con intercambio de información y alteraciones que generan diversidad.

### **Cromosoma:**

Un cromosoma es una posible solución al problema planteado y que internamente está formado por varios genes.

### **Gen:**

Grupo de parámetros únicos y diferentes dentro del cromosoma, que pueden ir tomando diferentes valores (alelos) y pueden ser modificados (mutar) por los diferentes operadores que se aplican al cromosoma.

### **Alelos:**

Éstos son los valores que puede tomar cada gen.

**Población:** es una cantidad  $n$ , de cromosomas dentro de una solución

**Generación:** son poblaciones que van sobreviviendo a medida que avanza el proceso de evolución y selección de cromosomas aptos para el desarrollo del problema (estas generaciones son creadas a partir de procesos selectivos aplicando los diferentes operadores de evolución como son el fitness)

## Operadores:

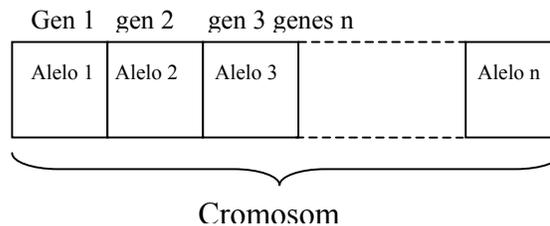
Fitnes: el fitness determina siempre los cromosomas que se van a reproducir, y aquellos que se van a eliminar, pero hay varias formas de considerarlo para seleccionar la población de la siguiente generación (evaluaciones propias de acuerdo a las necesidades)

## Crossover o cruzamiento:

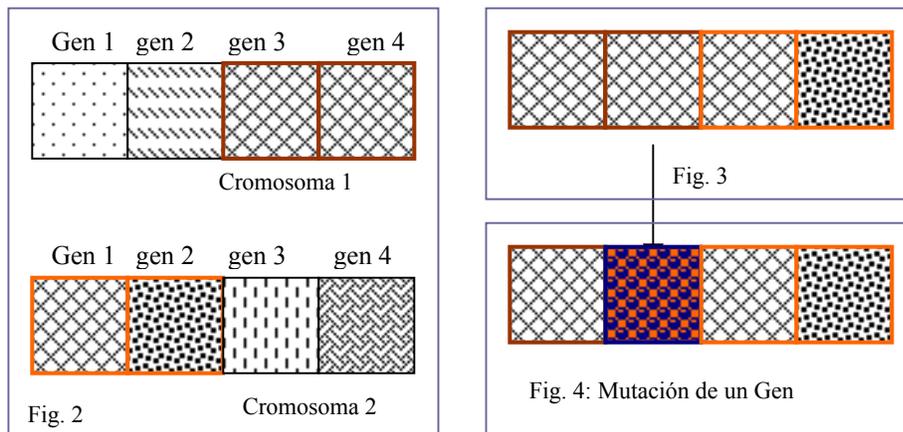
Consiste en el intercambio de material genético entre dos o más cromosomas

Mutación: contribuyen a la diversidad genética de la especie. En un algoritmo genético tendrán el mismo papel, y la misma frecuencia (es decir, muy baja).

## Modelo anatómico



- Operadores comunes
- Cruzamiento y mutación



En la figura dos podemos ver que tanto en el cromosoma uno como en el dos, se han seleccionado dos puntos de cruce, que normalmente son generados aleatoriamente, al unir estos dos puntos se generan nuevos cromosomas un tanto evolucionados (el cruzamiento puede ser negativo o positivo en la figura 3 vemos que para este caso fue un tanto negativo pues 3 de los 4 genes son idénticos).

En la figura 4 vemos como se ha seleccionado aleatoriamente un gen para ser mutado y generar diversidad para evitar poblaciones similares y evitar un mínimo local.

### **Ejemplo práctico**

Generar horarios con un gran numero de restricciones se vuelve una tarea muy compleja y la manera tradicional en que se lleva actualmente es mediante programación a manual con un tanto de “malicia indígena”, que puede generar soluciones pero no totalmente satisfactorias, por sus niveles de restricciones, pero como desarrollar un programa que simule esa “malicia indígena” cuando se empieza a convertir en funciones de crecimiento exponencial, pues hasta ahora, año 2005 y décadas a tras las formas mas acertadas de abarcarlos son los algoritmos evolutivos, heurísticas, algoritmos genéticos, colonias de hormigas y otros tantos.

basados en el hecho de que los algoritmos genéticos han sido exitosamente aplicados a otros problemas de horarios y cronogramas, hemos decidido hacer la implementación de un horario de clase usando este método, este ejemplo que se presenta continuación puede ser aplicado en una pequeña parte a la tesis CRONOS de la universidad de Manizales para generar horarios con aulas de clases que puedan usar algoritmos genéticos los cuales se describirán en las siguientes secciones(se extraen algunas partes no en su totalidad para dar un entendimiento no tan complejo para este documento).

### **Ejemplo**

#### **Datos**

Afortunadamente podemos utilizar datos del mundo real para tareas de horarios.

Para hacer que la comparación sea mas fácil los datos han sido estandarizados como:

- Todos los grupos de datos describen trabajos alrededor de las horas laborales docentes(cátedra directa) divididos en turnos.
- En cada turno hay diferentes posiciones que deben ser ocupadas, cada posición requiere uno o mas funciones especificas para que tengan buena presentación y solamente pueden ser llenados por un docente(secretario académico o decano) calificado para estas funciones.
- cada turno

Para cada posición de cada turno hay requerimientos que requieren un número de miembros de cada facultad, que estén disponibles a llenar esta posición. Estos requerimientos son notados de acuerdo al siguiente esquema:

Requerimientos mínimos, requerimientos objetivos, y máximo de requerimientos

- Hay una base de datos que entra a cada miembro del trabajo indicando si este cumple con las calificaciones requeridas (disponibilidad horario) para la calificación que va a llenar.
- Los docentes tienen que tener diferentes horas de trabajo a la semana, algunos son contratados tiempo completo, medio tiempo o catedráticos
- Las materias se inscriben dentro de áreas.
- No pueden quedar materias de una misma área en una misma jornada

## **Relaciones**

Cuando una lista está a punto de ser programada en una diferente relación tiene que ser considerada. Comúnmente las relaciones son divididas en dos grupos:

Relaciones rígidas las cuales tienen que ser llenadas sobre todas las circunstancias y las relaciones suaves que deben ser llenadas si es posible.

### **Ejemplos de relaciones rígidas son:**

Por ley el número mensual de horas trabajadas (con una tolerancia) debe tener un límite que no puede ser excedido, el número de horas asignados para cada uno de los docentes debe tener un mínimo y máximo

### **Las relaciones suaves son:**

Los días festivos bloquean los trabajadores esto debería ser lo más compacto posible los días de trabajo aparte o fiestas deben ser evitados (24 y 31 de dic) los turnos similares son escogidos de acuerdo a como los prefieran los docentes y decanos en común acuerdo, ejemplo, no siempre se alternan turnos de la noche y turnos del día.

Los turnos especialmente los de la noche y fines de semana deben ser compartidos la mayoría de los casos con una distribución uniforme a través de los docentes o áreas del pensum

## **El algoritmo**

El centro de nuestro algoritmo es un AG estándar con una selección y 4 puntos cruzados con 6 individuales. Sin embargo se utiliza la estrategia elitista para sustituir únicamente como el peor caso individual de la población al mejor caso

## **La codificación**

La codificación directa a sido escogida para representar semanalmente un grupo de docentes en un cromosoma, ejemplo:

Los genes representan las secuencias de las materias o las funciones de cada turno(ofrecimiento). Entonces tenemos el gen para cada posible asignación a un docente con una función la consecuencia de este método de codificación es una larga cadena de genes

Las tablas representan los ofrecimientos respectivamente y los ofrecimientos vacíos. La asignación de horas esta permitida únicamente si los docentes están disponibles y cumplen con todos los requisitos de las funciones de esta posición.

Usando el esquema codificado los campos son asignados a docentes con estas funciones de estos turnos los cuales están directamente relacionados y no tienen tiempo consumido y no se necesitan algoritmos decodificadores. La desventaja es que varias relaciones duras no son forzadas ej: una representación genotípica inválida es una solución inválida, como múltiples asignaciones a un docente en un solo turno. Sin embargo una gran parte significativa de las búsquedas de espacios es buscar por medio de un método que haga asignaciones certeras o de lo contrario será excluido de la búsqueda por un método codificado. Por esta razón la representación directa ha sido dada de preferencia a una representación implícita.

Por su puesto que los resultados tiene una violación a un largo estado del algoritmo

## **Las Mutaciones**

Escoger un esquema codificado induce a una mutación estándar de asignación a docentes. La mutación inicialmente definida es reducir linealmente hacia abajo a un limite bajo de dos cambios. Esto es para asegurar que el algoritmo converge y es óptimo

## **Función de desempeño**

El desempeño individual es computado por una asignación penal a la violación de las relaciones. Como se menciona anteriormente la violación debe tener

relaciones duras y es permitido escoger un esquema de codificación para abreviar las cosas los resultados de las violaciones presentan un puntaje de bajo desempeño y sin embargo se incrementa la evolución de este a modo individual.

Los costos penales son asignados a cada evento de una violación de la relación. La penalización es calculada dependiendo de la rigidez de la violación.

Ejemplo para la señalización de la violación

### **Los operadores de reparo**

El algoritmo genético describe una producción relativa de buenas listas, pero todavía presenta un número alto inaceptable de relaciones duras violadas. Para aprovechar la cualidad y velocidad del algoritmo era necesario introducir problemas específicos para acelerar la convergencia. La idea es aplicar los operadores de reparo para que la presentación sea una simple señalización de una relación cerrada. Estos nuevos operadores son aplicados de manera individual después de la selección, recombinación y mutación pero antes de calcular el rendimiento. La desventaja es que la búsqueda del comportamiento se vuelve más local. Entonces los parámetros del algoritmo tiene que ser escogidos cuidadosamente para asegurarse que una parte del rechazo no sea reducido también: ejemplo: que la búsqueda se vuelva local y rápida, entonces los operadores de reparo desarrollados son aplicados de mejor manera a un impacto y gradualmente se incrementan. Este impacto es para ser bien utilizado durante un experimento.

El operador de reparo lleva afuera las siguientes modificaciones:

#### **Cancelando una tarea**

Cuando un ofrecimiento se presenta más de una vez en un día, todas las asignaciones se le deben ser canceladas menos una

#### **Selección de asignaciones**

A un docente se le pueden asignar días sin previo aviso, ósea que si un numero de horas trabajadas al mes, son menores al numero de horas objetivo.

Seleccionar cancelado: de una asignación y de asignársela a otro docente. A veces es más favorable respetar los niveles del personal en los horarios que cancelar una asignación a un docente y reasignársela a otro que tenga un horario en el mismo día

Cambiando de asignación: de dos docentes que producen más secuencias homogéneas de horarios para los docentes.

## Resultados

El AG presentado aquí a sido probad diferentes bases del dato del mundo real y a producido soluciones decentes en casi todos los casos ejemplo

El algoritmo calcula un horario válido con una aproximación. 15 docentes en menos de 45 minutos

Conclusiones generales extraídas de experiencias propias y de especialistas en el área de los AG

Este es el título de un artículo que se publicó en la conferencia sobre algoritmos genéticos celebrada en el año 89 (ICGA 89), en donde da una serie de consejos para que se apliquen los algoritmos genéticos debidamente, y avisa a aquellos que se quieren apartar de la ortodoxia. Estos consejos son los siguientes

-Deja que la naturaleza sea tu guía: dado que la mayoría de los problemas a los que se van a aplicar los algoritmos genéticos son de naturaleza no lineal, es mejor actuar como lo hace la naturaleza, aunque intuitivamente pueda parecer la forma menos acertada. Si queremos desarrollar sistemas no lineales que busquen y aprendan, mejor que comencemos (como mínimo) imitando a sistemas que funcionan (goldberg). Y estos sistemas se hallan en la naturaleza.

-Cuidado con el asalto frontal: a veces se plantea el problema de pérdida de diversidad genética en una población de cromosomas. Hay dos formas de resolver este problema: aumentar el ritmo de mutación, lo cual equivale a convertir un algoritmo genético en un algoritmo de búsqueda aleatoria, o bien introducir mecanismos como el sharing, por el cual el fitness de un individuo se divide por el número de individuos similares a él. Este segundo método, más parecido al funcionamiento de la naturaleza, en la cual cada individuo, por bueno que sea, tiene que compartir recursos con aquellos que hayan resuelto el problema de la misma forma, funciona mucho mejor. Otro caso que surge a menudo en los grupos de discusión de grupos especializados es el tratar de optimizar AG mediante AG; es mucho mejor tratar de entender el problema que acercarse a él de esta manera.

-Respetar la criba de esquemas: para ello, lo ideal es utilizar alfabetos con baja cardinalidad (es decir, con pocas letras) como el binario.

-No te fíes de la autoridad central: la naturaleza actúa de forma distribuida, por tanto, se debe de minimizar la necesidad de operadores que vean a cada la población. Ello permite, además, una fácil paralelización del algoritmo genético. Por ejemplo, en vez de comparar el fitness de un individuo con todos los demás,

se puede comparar sólo con los vecinos, es decir, aquellos que estén, de alguna forma, situados cerca de él.

## ANEXO D ANÁLISIS Y DISEÑO DEL SISTEMA

### REQUERIMIENTOS

#### Requerimientos obligatorios

**Desarrollo y funcionamiento bajo software libre:** En la actualidad los sistemas de información que quieran ser implementados para dar solución a algún problema de la Universidad de Manizales, deben cumplir con la política y perfil adoptados por ella. Las políticas de desarrollo de software en la actualidad y en el caso preciso de esta tesis se enmarcan en plataformas de software libre o sistemas no propietarios tales como: Sistemas operativos Linux, BSD, Bases de datos libres como Postgres o Mysql, servidores Web como Apache para mencionar algunos.

**Sistema Web centralizado:** El sistema de información que se pretenda desarrollar así vea involucradas diferentes áreas, secciones administrativas o sistemas no debe ser un sistema aislado como lo podría entregar en su análisis. Debe ser un sistema único y centralizado que involucre estas diferentes áreas o secciones en un solo sistema o base de datos, que además a su vez sea capaz de integrar perfiles diferentes que discriminen cada área o sección en el momento de ser accedido.

#### Requerimientos del sistema de información

**Análisis y Diseño:** Se debe construir un modelo de análisis para abstraer los aspectos esenciales del dominio de la aplicación sin tener en cuenta la implementación eventual. En este modelo se deben tomar decisiones importantes que después se completan para optimizar la implementación. Después de esto el modelo de diseño se deberá implementar en algún lenguaje de programación libre, base de datos y hardware específico.

**Control de acceso y especificidad de usuarios:** A través del proceso de Análisis y diseño del sistema de Información CRONOS se devela la necesidad de caracterizar y discriminar por perfiles de acceso cada una de las funciones del sistema. Cada perfil debe acceder solo a lugares e interfaces específicas de sus funciones académicas, por ejemplo las actividades normales y administrativas de

un decano no deben ser accedidas por un docente, o un decano no tiene por que acceder a las funciones de la oficina de planeación de la Universidad.

**Manipulación de horarios para ofrecimientos y proyectos:** Se debe mostrar un horario completo y dinámico que presente la información tanto de horario de clases como, proyectos y otras cargas adicionales en las que se vea involucrado el personal académico de la Universidad. Igualmente integrado y alimentado por su disponibilidad horaria y tipo de vinculación laboral.

**Clasificación de cargas académicas adicionales:** Se deben Caracterizar o clasificar las cargas académicas diferentes a la cátedra, dando también una información general de esa carga como tipo, máximo de docentes y horas asignadas a la carga adicional entre otros.

**Reportes de distribución de carga académica:** Se debe presentar informes que permitan observar todos los elementos enmarcados en la carga académica final como nombre del empleado, tipo de carga, horas por cada tipo de proyecto, horas por cátedra y un acumulado final que identifique si cantidad de horas asignadas cumple con su contratación. Además de permitir al decano, talento humano entre otros hacer análisis de carga académica y mostrar también a donde se están dirigiendo sus esfuerzos como lo son la investigación, la extensión o la cátedra.

**Disponibilidad:** El sistema debe permanecer en línea bajo la base de 7X24x365, siete días a la semana, las 24 horas, durante todo el año ya que es un sistema de consulta en línea, donde tanto estudiantes como docentes se benefician de el. La métrica a utilizar es:  $\% \text{ de disponibilidad} = (\text{tiempo transcurrido} - \text{suma de horas off-line}) / \text{tiempo transcurrido}$ . Para lograr un porcentaje de disponibilidad de un 99% Lo cual es igual a 3, 65 días off-line de 365 días, Esta métrica de disponibilidad es adoptada para contemplar posibles causas de inactividad.

**Escalabilidad:** El sistema debe permitir adaptarse a un posible crecimiento de sus servicios y usuarios para satisfacer las demandas de rendimiento cada vez mayores.

**Portabilidad:** el servicio web y base de datos puede ser implementado en cualquier sistema operativo y su visualización debe ser transparente del navegador usado por los usuarios.

## **SOFTWARE:**

**Servicio Web:** El sistema debe ser implementado bajo el servicio web HTTPD APACHE 2.0 usado en plataformas GNU y/o propietarias.

### **Servicio de Bases de Datos:** Motor Postgres Vs 7.3 Min.

El sistema fue implementado bajo el gestor de base de datos Postgres. La aplicación es probada en versiones 7.3 y 8.0. sin embargo la aplicación fue desarrollada bajo un modelo de portabilidad que le permite instalarse en cualquier ODBC(Mysql, Oracle, etc)

**Servicio de transferencia de archivos y actualización de contenidos:** Las actualizaciones a la base de datos y/o web deben hacerse a través de clientes SSH por el nivel de encriptación de datos que maneja. Acceder a el vía FTP negaría el acceso a este y violaría las políticas de seguridad en las cuales fue implementado.

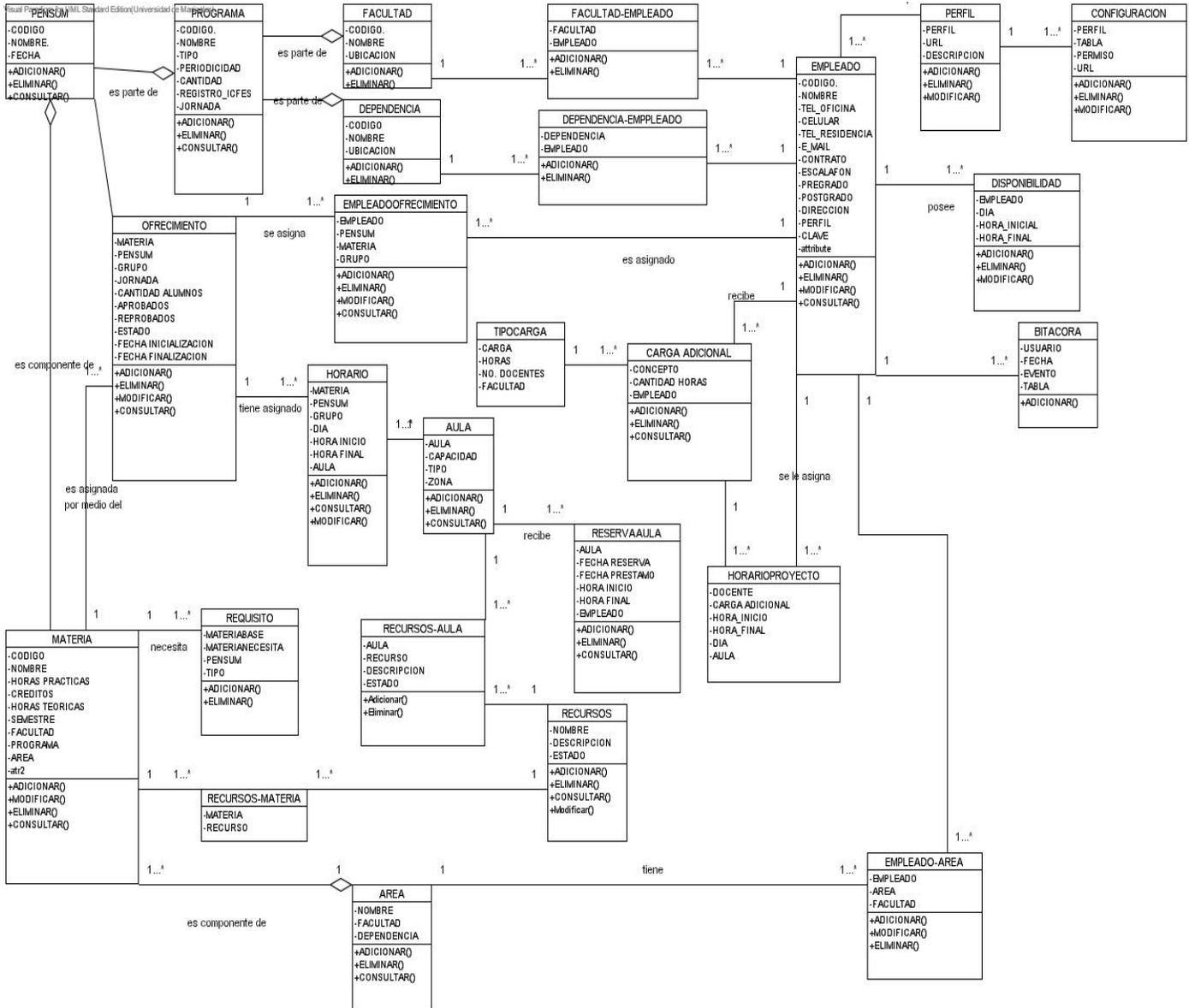
### **Requisitos opcionales**

**Administración de Recursos** (Audiovisuales, sillas, cortinas): Administración de recursos audiovisuales y demás recursos inmobiliarios, además de su asignación a clases, aulas y materias.

**Reserva de Aulas:** Permitir la reserva de aulas a diferentes empleados de la Universidad, bloqueando situaciones no ideales de reserva y permitiendo el acceso a consulta de aulas para reserva.

**Estadísticas de contratación y escalafón:** Un sistema que permita observar la situación actual de contratación de la Universidad clasificado por facultades, También mostrar una estadística del nivel del empleado en la Universidad a partir de su escalafón, lo cual podría llevar a diferentes análisis económico-administrativo a las diferentes partes administrativas de la Universidad.

## ANEXO E. DIAGRAMA DE CLASES



## ANEXO F. DICCIONARIO DE CLASES Y ATRIBUTOS

<b>Clase</b>	<b>Pensum</b>	
<b>Descripción</b>	Plan estudiantil creado para los programas de cada facultad y está compuesto por un conjunto de asignaturas adecuadamente sistematizadas que el estudiante debe cursar y aprobar en el transcurso de su carrera.	
<b>Atributos</b>	codigo varchar(20) nombre varchar(80) fecha date	Código del pensum Nombre del pensum Fecha de creación
<b>Métodos</b>	Adicionar() Modificar() Consultar()	

<b>Clase</b>	<b>programa</b>	
<b>Descripción</b>	Programa Académico es el conjunto de actividades orientadas a la formación profesional en una determinada área del saber y a la obtención de un título profesional.	
<b>Atributos</b>	tipo varchar(2) nombre varchar(60) codigo varchar(2) periodicidad varchar(10) cantidad varchar(2) registro_icfes varchar(15) jornada varchar(6) hora_inicio_clases time	Tipo de programa académico Nombre del programa Código del programa Periodicidad del programa No. de niveles del programa Registro Icfes para el programa Jornada del programa Hora de inicio de clases para el programa
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>facultad</b>	
<b>Descripción</b>	Sección de una universidad que engloba estudios de una misma rama.	
<b>Atributos</b>	codigo varchar(2) nombre varchar(40)	Código de la facultad Nombre de la facultad

	ubicacion varchar (2)	Ubicación de la facultad
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>dependencia</b>	
<b>Descripción</b>	Sección de una universidad que engloba estudios de una misma rama.	
<b>Atributos</b>	codigo varchar(2) nombre varchar(40) ubicacion varchar (2)	Código de la Dependencia Nombre de la Dependencia Ubicación de la Dependencia
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadofacultad</b>	
<b>Descripción</b>	Docentes y empleados asociados a una facultad de la Universidad.	
<b>Atributos</b>	facultad varchar(2) empleado varchar(12)	Código facultad Cédula del empleado
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadodependencia</b>	
<b>Descripción</b>	Docentes y empleados asociados a una dependencia de la Universidad.	
<b>Atributos</b>	dependencia varchar(2) empleado varchar(12)	Código Dependencia Cédula del empleado
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleado</b>	
<b>Descripción</b>	Personas que hacen parte del sistema, con su respectiva información personal.	
<b>Atributos</b>	codigo varchar(12) nommbre varchar(40) tel_oficina varchar(10) celular varchar(10) tel_residencia varchar(10) email varchar(30) contrato varchar(2) escalafon integer pregrado varchar(40) postgrado varchar(40) direccion varchar(40) perfil varchar(1)	Cédula del empleado Nombre Teléfono de la oficina Celular Teléfono de la residencia Correo Electrónico Tipo de contrato laboral Tipo de escalafón docente Pregrado Postgrado o Especialización Postgrado o Especialización Tipo de perfil asociado
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>bitacora</b>	
<b>Descripción</b>	Información histórica de control de acciones ejecutadas sobre las tablas del sistema por cualquier usuario.	
<b>Atributos</b>	usuario varchar(12) fecha timestamp evento varchar(10) tabla varchar(20)	Cédula del empleado Fecha de la acción Acción realizada Tabla del sistema afectada
<b>Métodos</b>	Adicionar() Consultar()	

<b>Clase</b>	<b>contrato</b>	
<b>Descripción</b>	Tipos de contrato que se pueden presentar para un docente según el no. de horas.	
<b>Atributos</b>	contrato varchar(3) nombre varchar(70) horas numeric(3)	Tipo del contrato Nombre del tipo de contrato No. de horas mínimas para el tipo de contrato
<b>Métodos</b>	Adicionar() Modificar() Eliminar()	

	Consultar()
--	-------------

<b>Clase</b>	<b>cargaadicional</b>	
<b>Descripción</b>	Asignación de carga académica adicional a un docente, en una cantidad de horas determinada.	
<b>Atributos</b>	concepto numeric(2) empleado varchar(12) cantidad horas numeric(3)	Código tipo de carga adicional Cédula del empleado Cantidad de horas para el proyecto
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>tipocarga</b>	
<b>Descripción</b>	Corresponde a las cargas académicas adicionales que no hacen parte de la orientación de clases como son los proyectos, procesos administrativos, investigación, planeación, entre otros.	
<b>Atributos</b>	carga varchar(40) no docentes numeric(2) facultad varchar(2)	Carga adicional No de docente para la carga adicional Facultad donde se asignará la carga adicional
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>perfil</b>	
<b>Descripción</b>	Corresponde a los grupos identificados de usuarios que acceden al sistema con unos permisos comunes dentro del perfil.	
<b>Atributos</b>	perfil varchar(12) Descripción varchar(40) url varchar(40)	Código del perfil Descripción de perfil Url home asociada al perfil
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadoarea</b>	
<b>Descripción</b>	Docentes que poseen habilidades para un área del conocimiento que agrupa varias materias.	
<b>Atributos</b>	empleado varchar(12) area numeric(2) facultad varchar(2)	Cédula del empleado Área a la que se vincula el empleado Facultad donde se realiza la vinculación
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	
<b>Clase</b>	Area	
<b>Descripción</b>	Agrupación de materias según una determinada área del conocimiento.	
<b>Atributos</b>	nombre varchar(40) facultad varchar(2) dependencia varchar(2)	Nombre el área Facultad del área (si es facultad) Dependencia del área (si es dependencia)
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>materia</b>	
<b>Descripción</b>	Asignatura que hace parte de un pensum dentro de un programa y asociado a la facultad.	
<b>Atributos</b>	cod_materia varchar(12) nombre varchar(20) horas practicas numeric(2) creditos varchar(1)  horas teoricas numeric(2) semestre varchar(2) facultad varchar(2), programa varchar(2) references programa(cod_prog), area numeric(2)	Código de la materia Nombre de la materia No. de horas prácticas No. mínimo de créditos para aprobar materia No. de horas teóricas Semestre en que se dicta la materia Facultad asociada a la materia Programa asociada a la materia  Área asociada a la materia
<b>Métodos</b>	Adicionar() Modificar() Eliminar()	

	Consultar()
--	-------------

Clase	ofrecimiento	
<b>Descripción</b>	Es la materia con un grupo y pensum asociados, para ser orientada a los estudiantes.	
<b>Atributos</b>	materia varchar(12) pensum varchar(12) grupo numeric(1) jornada varchar(1) cantidad alumnos numeric(2) aprobados numeric(2) reprobados numeric(2) estado varchar(9) fecha_inicializacion date fecha_finalizacion date	Código de la materia Código del pensum Grupo del ofrecimiento Jornada No. de alumnos Alumnos que aprueban al finalizar el semestre Alumnos que reprueban al finalizar el semestre Estado del ofrecimiento (Activo Inactivo) Fecha de inicio de clases Fecha de finalización de clases
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

Clase	Horario	
<b>Descripción</b>	Corresponde al ofrecimiento que se orientará a los estudiantes, con una hora inicial y hora final en un día determinado de la semana y con aula asociada.	
<b>Atributos</b>	materia varchar pensum varchar grupo numeric día numeric(1) hora_inicio time hora_final time aula varchar(7)	Código de la materia Código del pensum Grupo de ofrecimiento Día de la semana Hora Inicio Hora Final Aula para el horario
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>requisito</b>	
<b>Descripción</b>	Relación existente entre las materias de un pensum, la cual puede ser: Prerrequisito: una asignatura es el prerrequisito de una segunda cuando para poder formalizar la matrícula de la segunda es necesario tener aprobada la primera. Correquisito: una asignatura es correquisito de una segunda cuando para poder aprobar la segunda es necesario matricularse también de la primera o tenerla aprobada	
<b>Atributos</b>	materiabase varchar(12) , materianecesita varchar(12) pensum varchar(20), tipo varchar(12) not null	Código Materia base Código Materia analizada Código del pensum Tipo de requisito
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadofrecimiento</b>	
<b>Descripción</b>	Relación en la cual el docente orientará un determinado ofrecimiento.	
<b>Atributos</b>	docente varchar(12) pensum varchar(12) materia varchar(12) grupo numeric(1)	Cédula del docente Código del pensum Código de la materia No. de grupo
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>disponibilidad</b>	
<b>Descripción</b>	Horas se la semana por día, en el cual el docente se encuentra disponible para la orientación de una materia y/o para la asignación de una carga adicional.	
<b>Atributos</b>	empleado varchar(12) dia numeric(1) hora_inicial time hora_final time	Cédula del empleado Código del dia Hora inicio de bloque disponible Hora final de bloque disponible

<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()
----------------	---

<b>Clase</b>	<b>aula</b>	
<b>Descripción</b>	Lugar registrado por Planeación para la orientación de una materia.	
<b>Atributos</b>	aula varchar(7) capacidad varchar(3) tipo numeric(2) zona varchar(2)	Código del aula Capacidad del aula Tipo de aula Zona del aula
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>recursoaula</b>	
<b>Descripción</b>	Recurso físico que se encuentra dentro del aula y su estado asociado.	
<b>Atributos</b>	aula varchar(7) recurso numeric(2) Descripción varchar(40) estado varchar(1)	Código del aula Recurso del recurso Descripción del recurso Estado del recurso
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>recursomateria</b>	
<b>Descripción</b>	Recursos requeridos por la materia para su adecuada orientación.	
<b>Atributos</b>	materia varchar(12) recurso numeric(2)	Código de la materia Código del recurso asociado a la materia
<b>Métodos</b>	Adicionar() Modificar() Eliminar()	

	Consultar()
--	-------------

<b>Clase</b>	<b>horarioproyecto</b>	
<b>Descripción</b>	Días y horas asociadas a un proyecto o carga adicional por docente registrado.	
<b>Atributos</b>	docente varchar(12) adicional numeric(2) hora_inicio time hora_final time dia numeric(1) cod_aula varchar(7)	Cédula del docente Código del proyecto Hora inicio asociado al día Hora final asociado al día Día de asignación del proyecto Código del aula para la carga adicional (si es necesario)
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

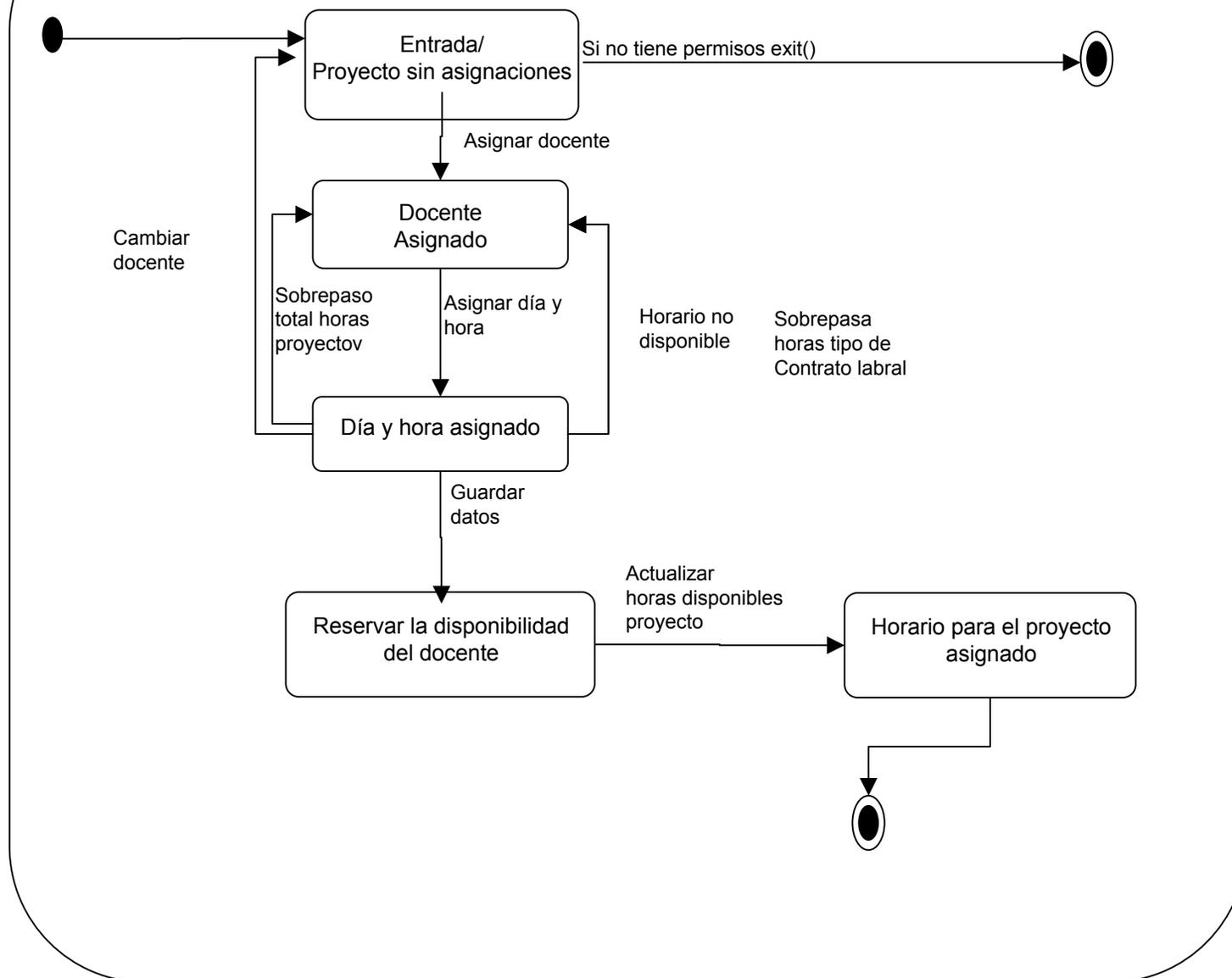
<b>Clase</b>	<b>reservaaula</b>	
<b>Descripción</b>	Fecha y horas en las cuales se requiere un aula para una actividad determinada, asociada a un empleado registrado.	
<b>Atributos</b>	cod_aula varchar(7) fecha_reserva date fecha_prestamo date hora_inicio time hora_final time empleado varchar(12)	Código del aula Fecha en que se utilizará el aula Fecha en que se reserva el aula Hora inicio de la reserva Hora final de la reserva Cédula del empleado que reserva
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

**11.4.4.1**  
**ANÁLISIS Y DISEÑO**  
**MODELO DINAMICO**  
**MAQUINA DE ESTADOS**

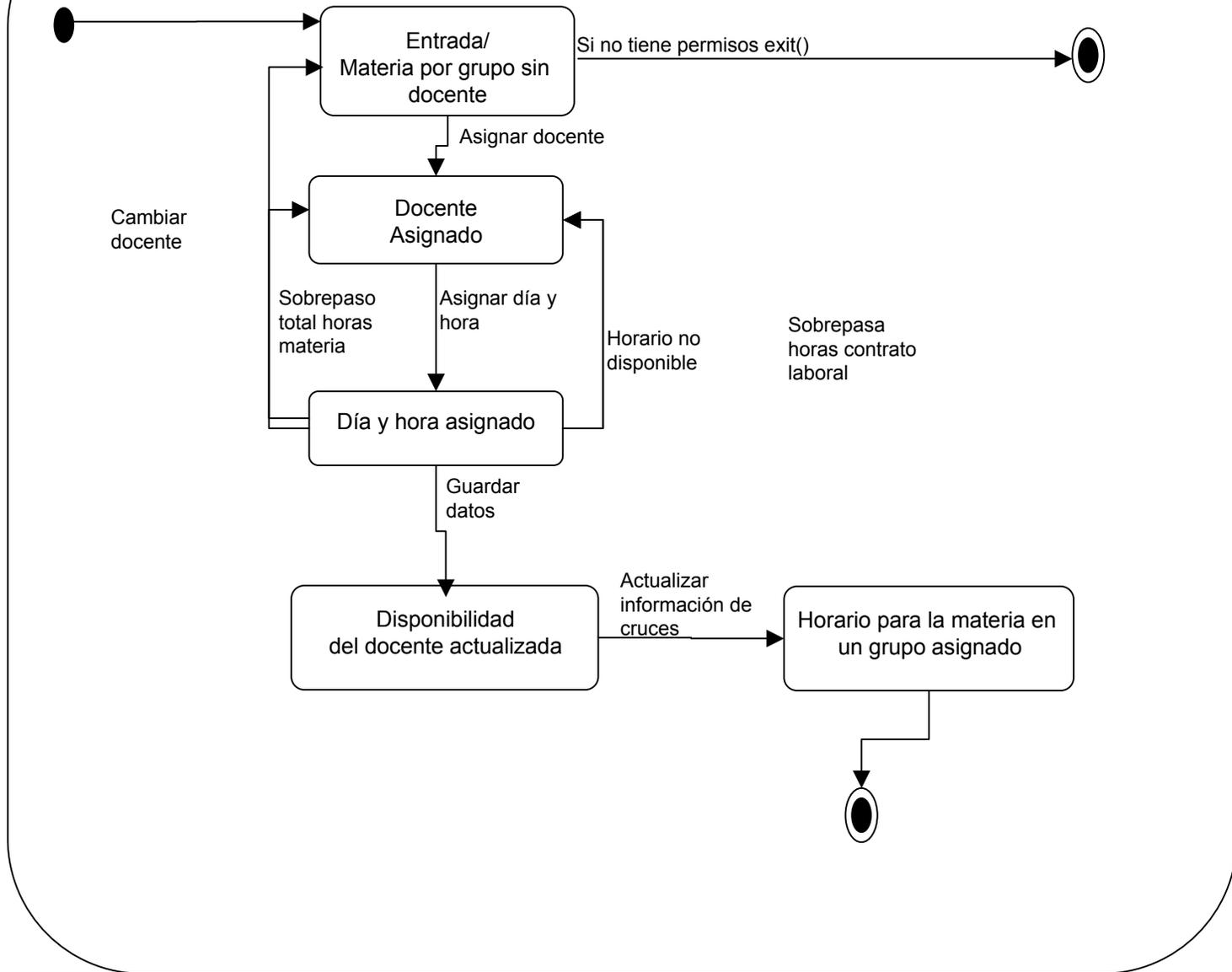
**CRONOS: SISTEMA DE CARGA ACADÉMICA Y PLANEACIÓN DE HORARIOS**



# CARGA ADICIONAL



# HORARIO

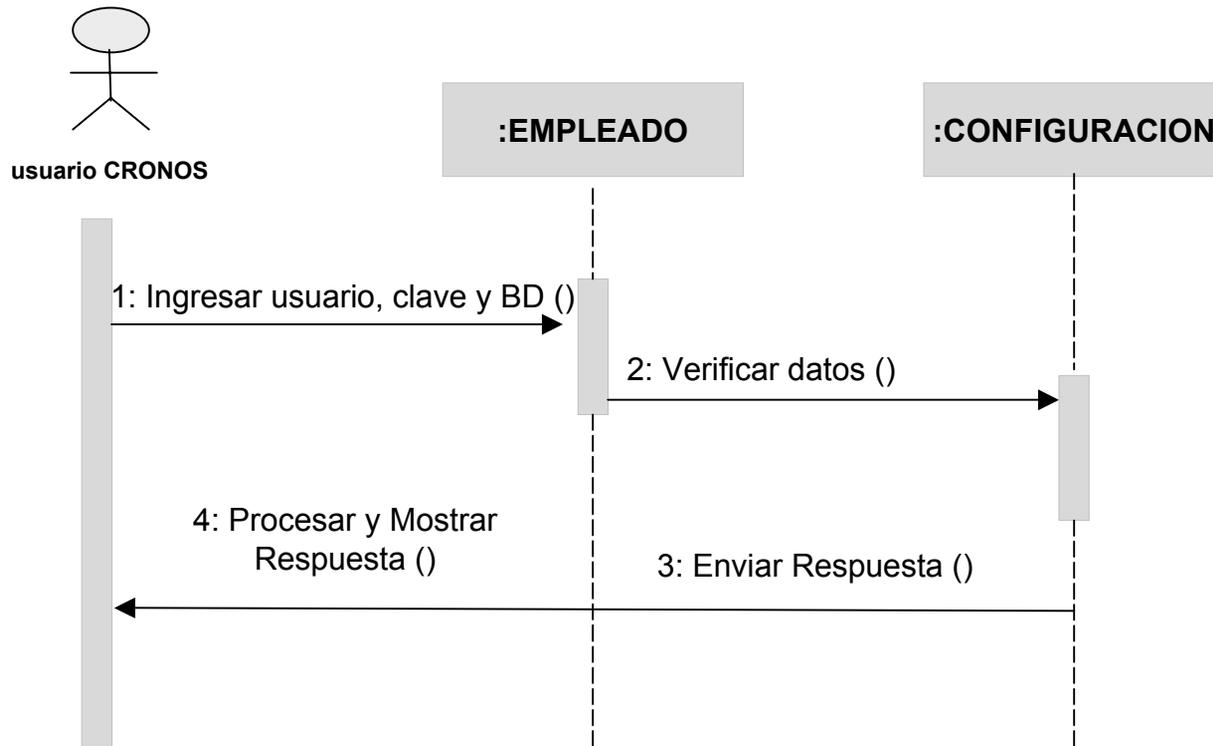


**ANEXO J**  
**ANÁLISIS Y DISEÑO**  
**DIAGRAMA DE SECUENCIA**

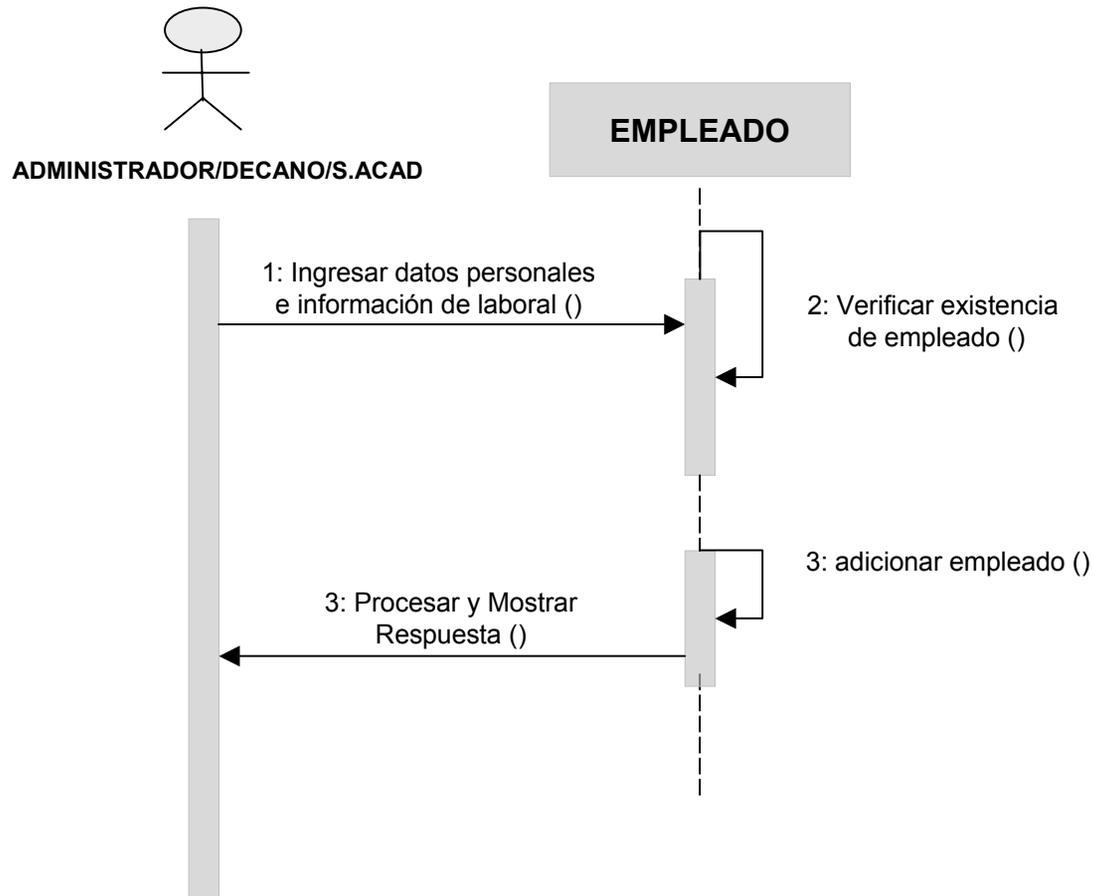
**CRONOS: SISTEMA DE CARGA ACADÉMICA Y PLANEACIÓN DE HORARIOS**



# INGRESAR A CRONOS

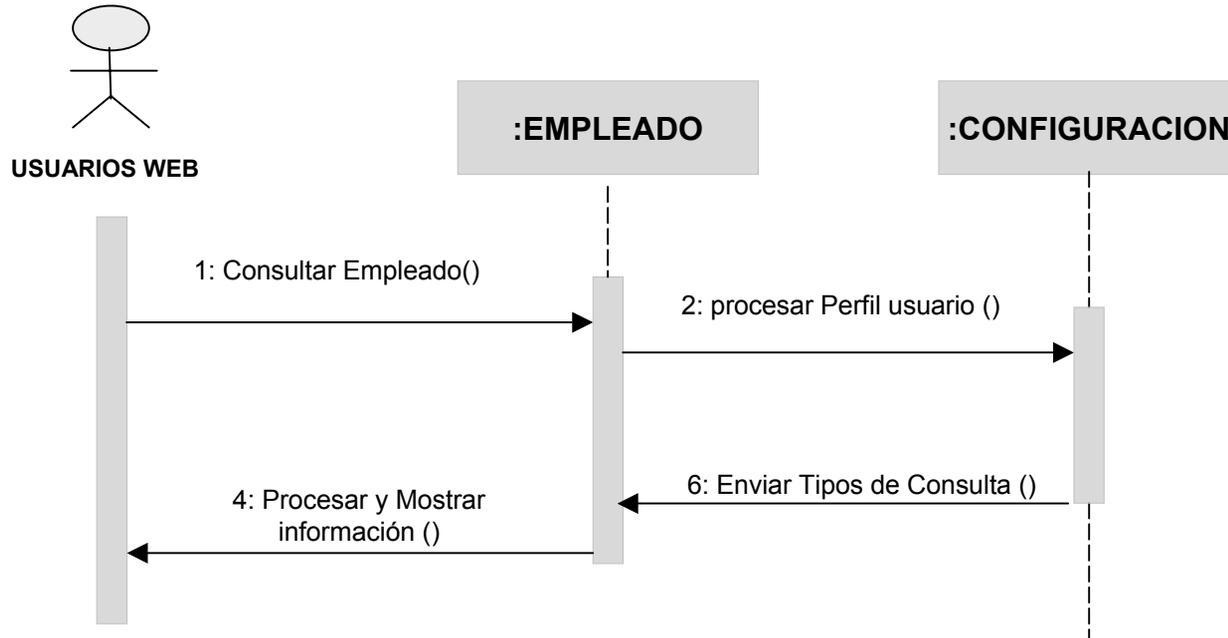


# CREAR USUARIOS ADMINISTRATIVOS

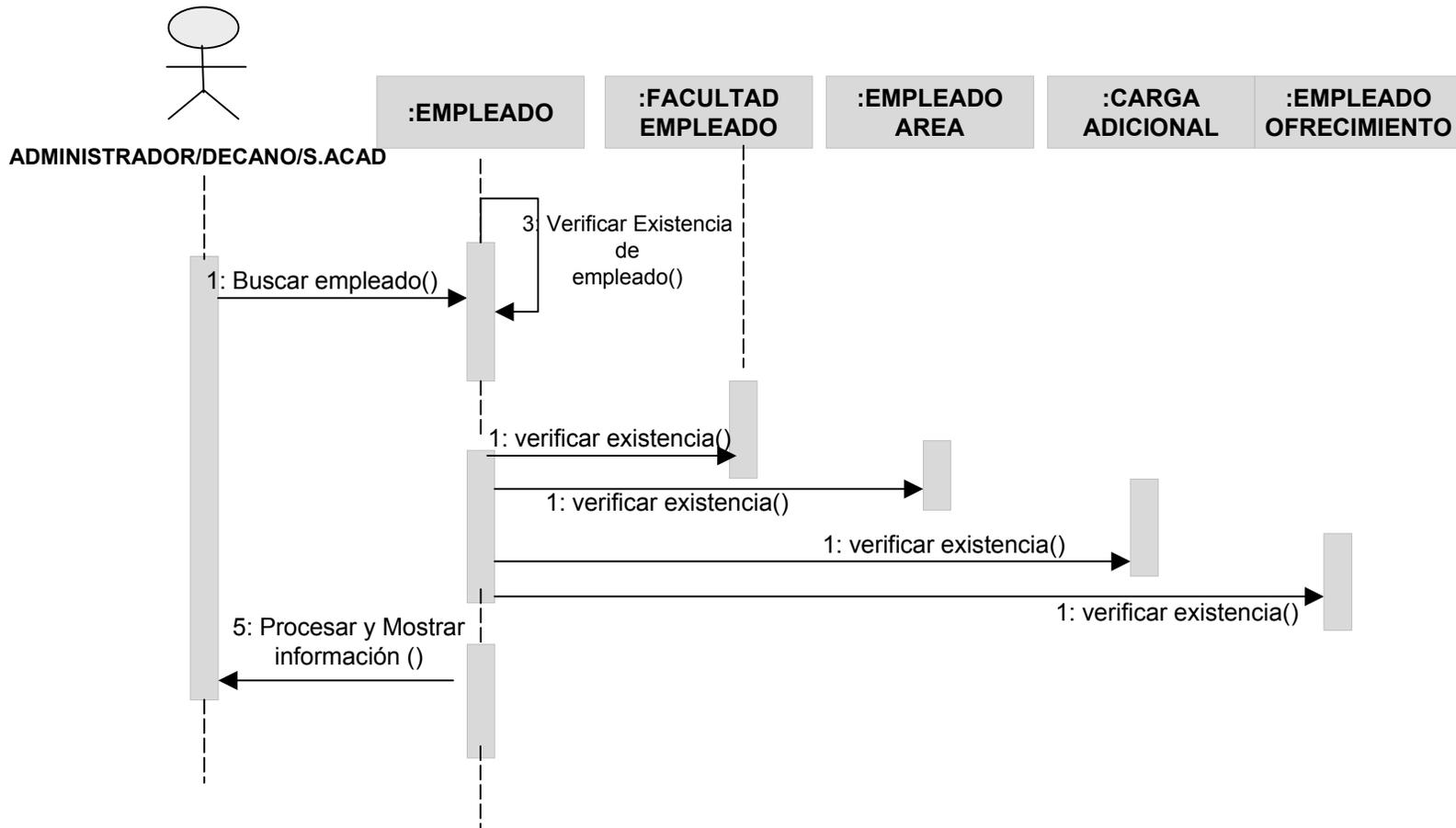




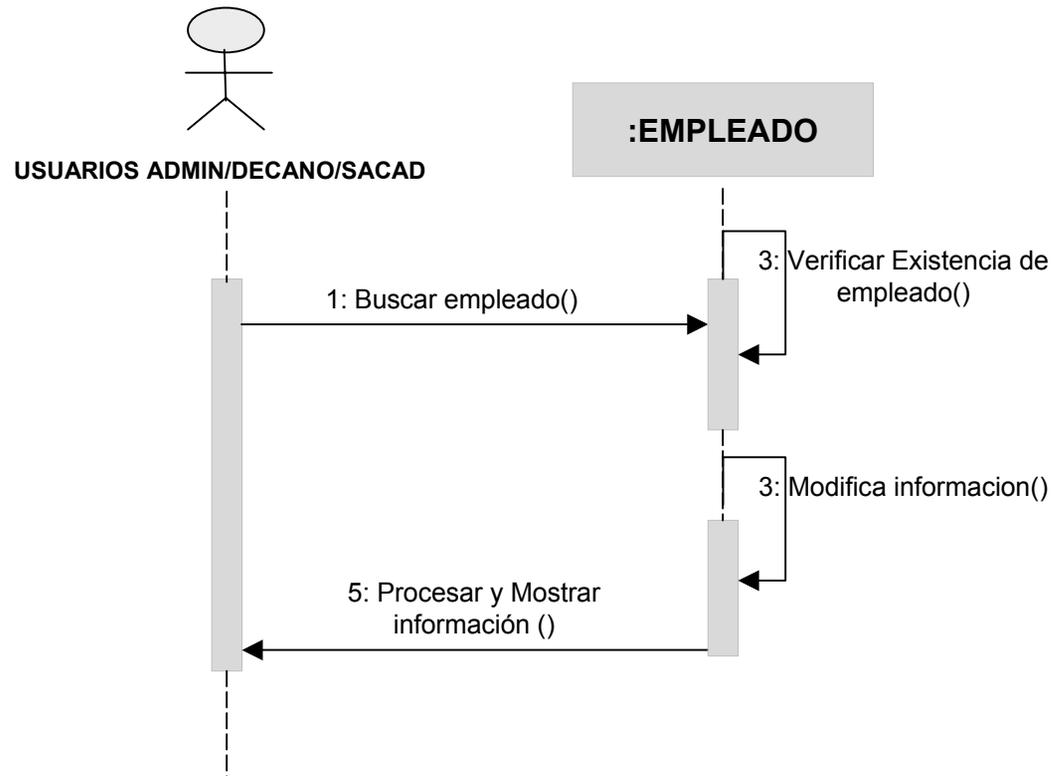
# CONSULTAR EMPLEADOS



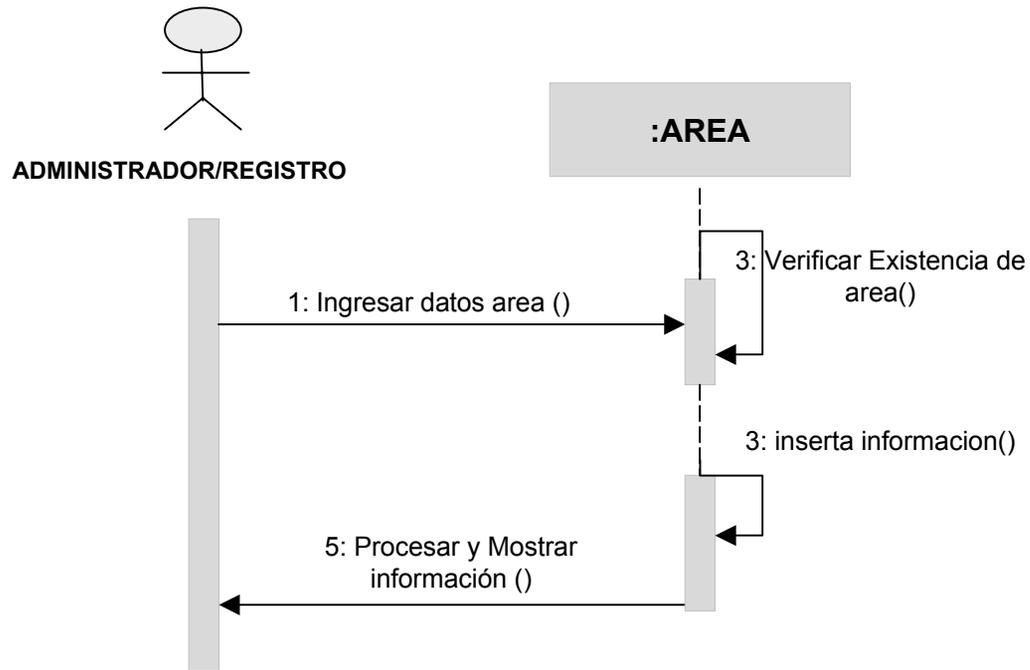
# ELIMINAR EMPLEADOS



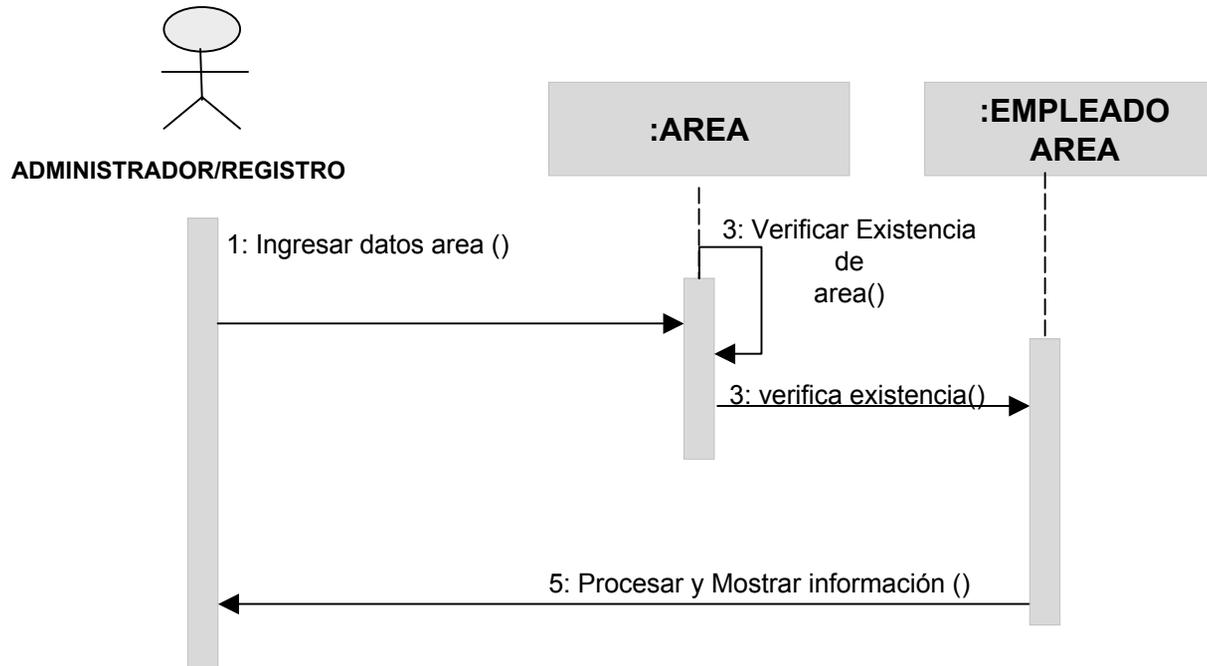
# MODIFICAR EMPLEADOS



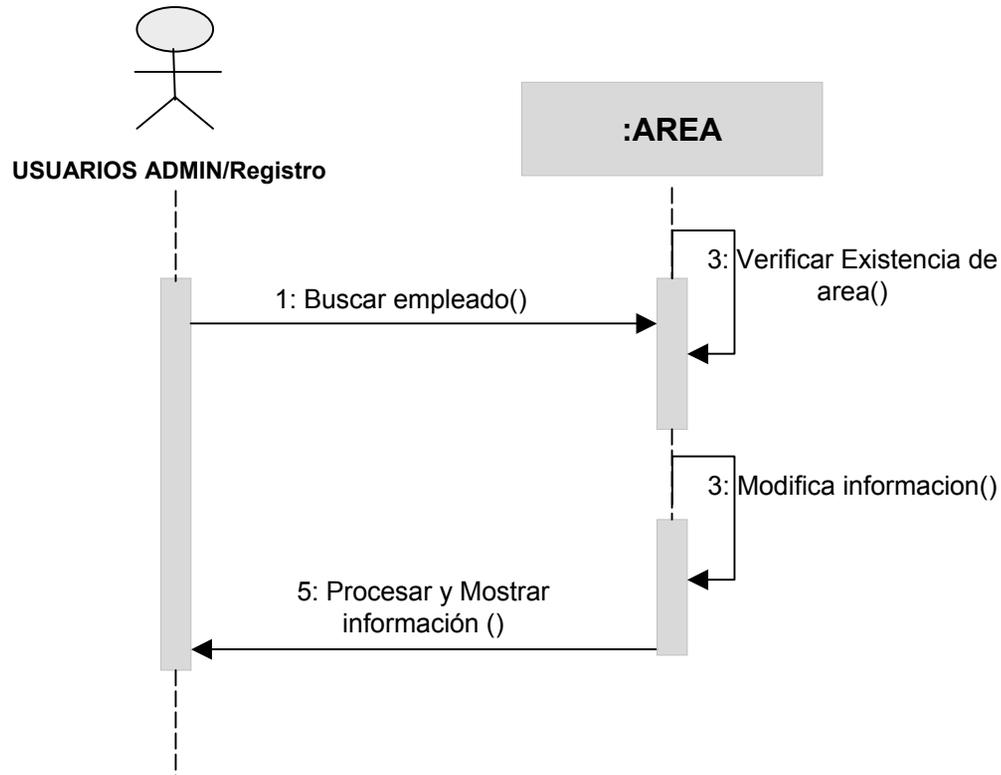
# CREAR AREAS



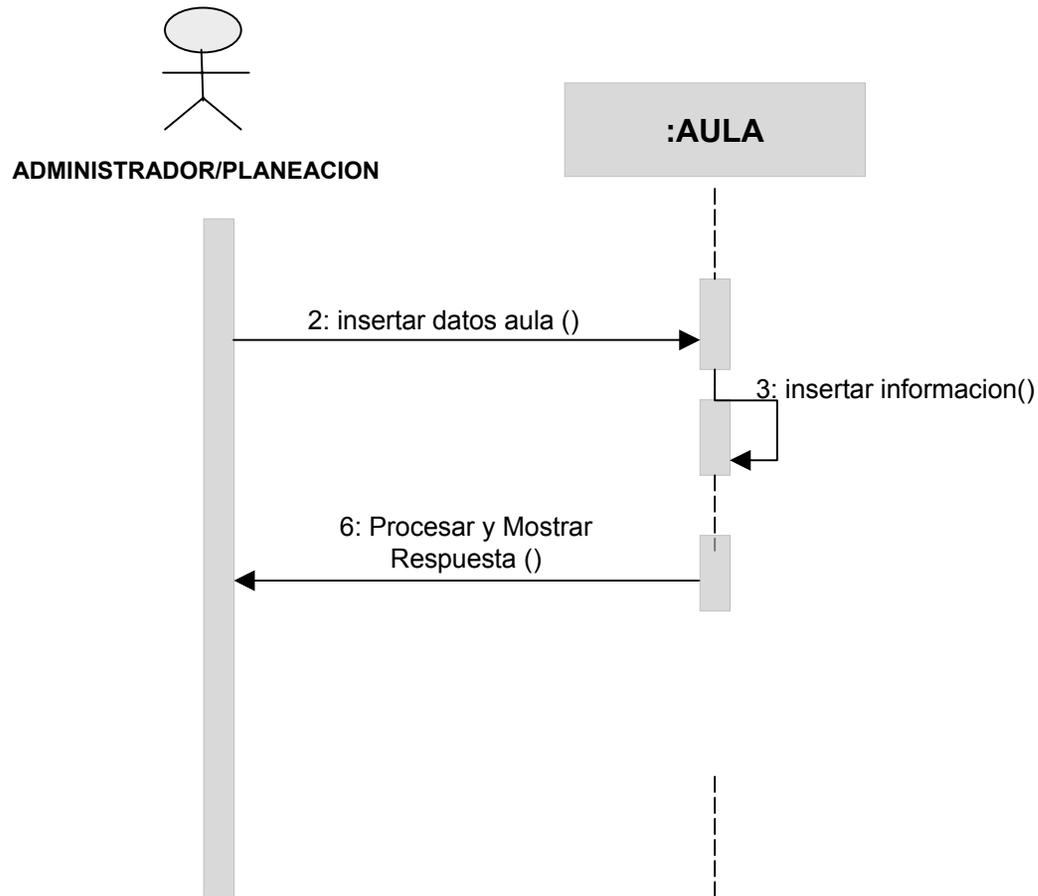
# ELIMINAR AREAS



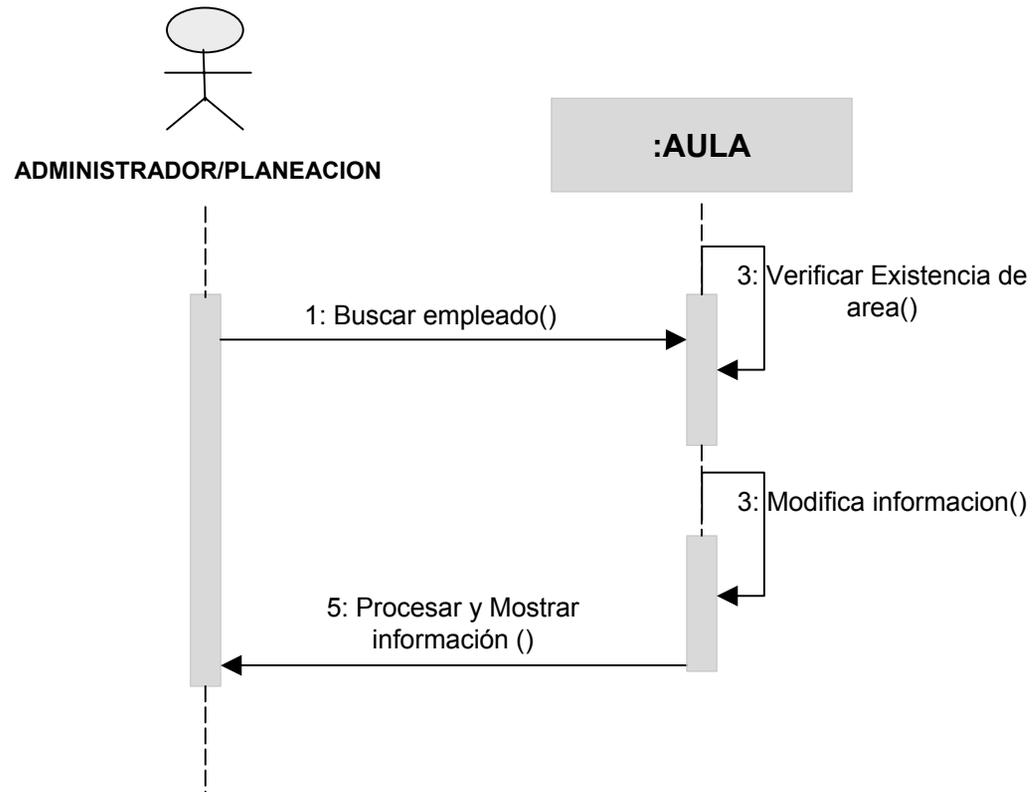
# MODIFICAR AREAS



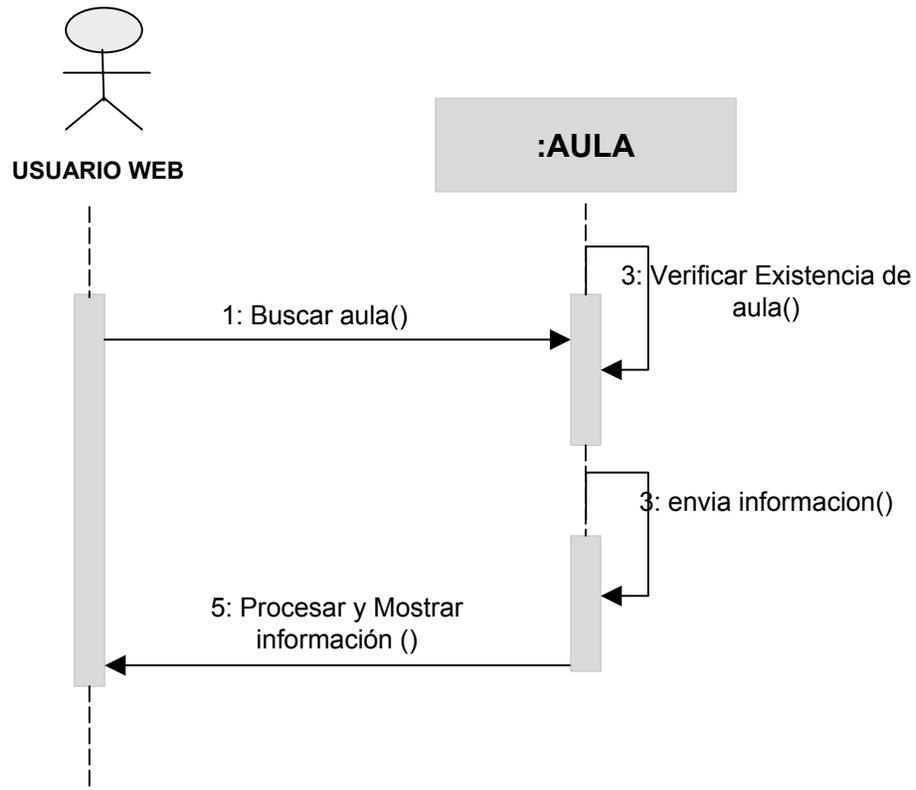
# CREAR AULAS



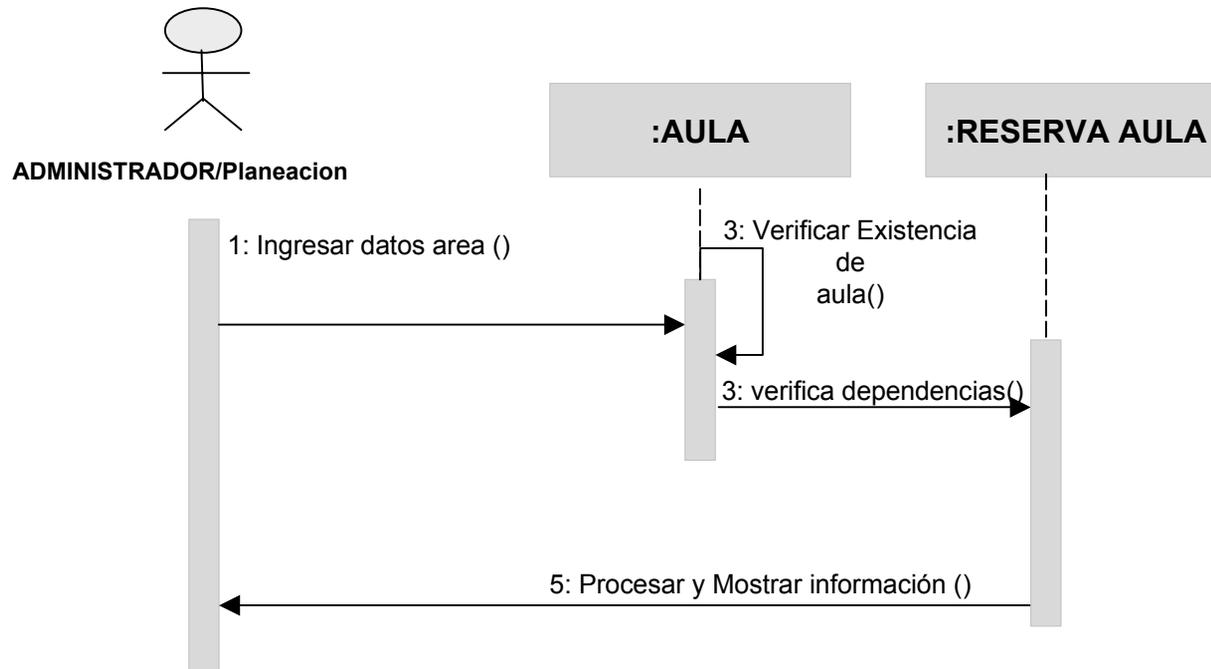
# MODIFICAR AULAS



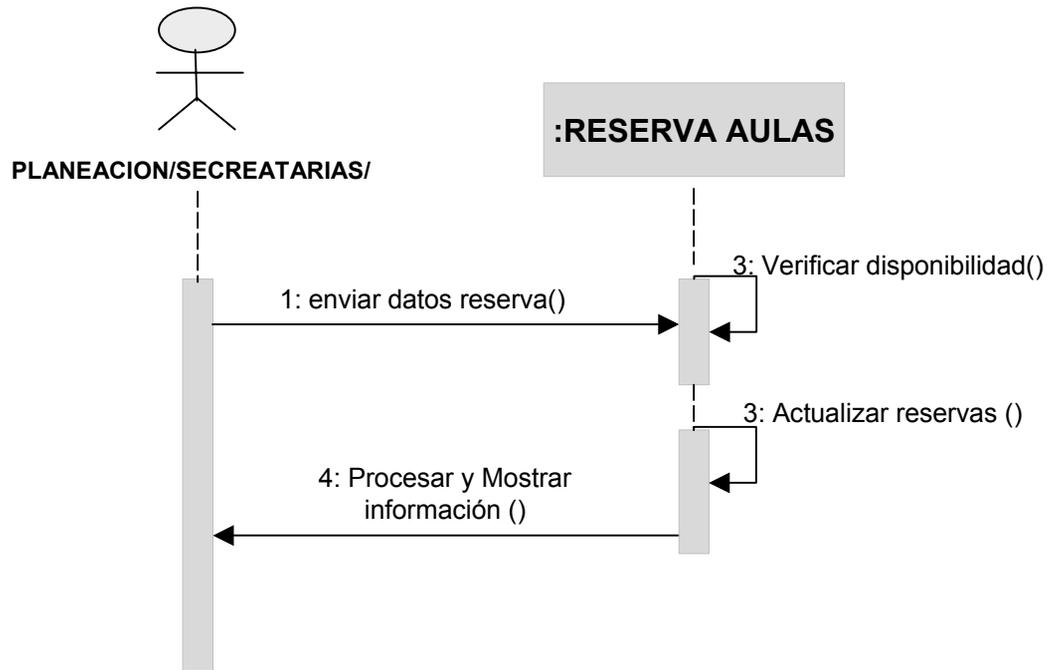
# CONSULTAR AULAS



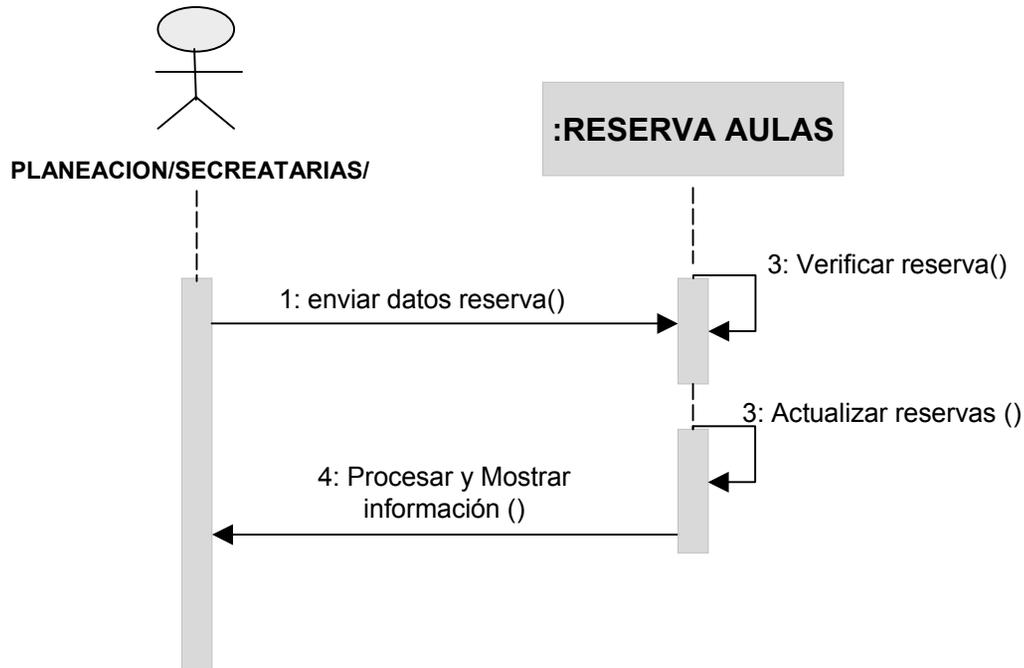
# ELIMINAR AULAS



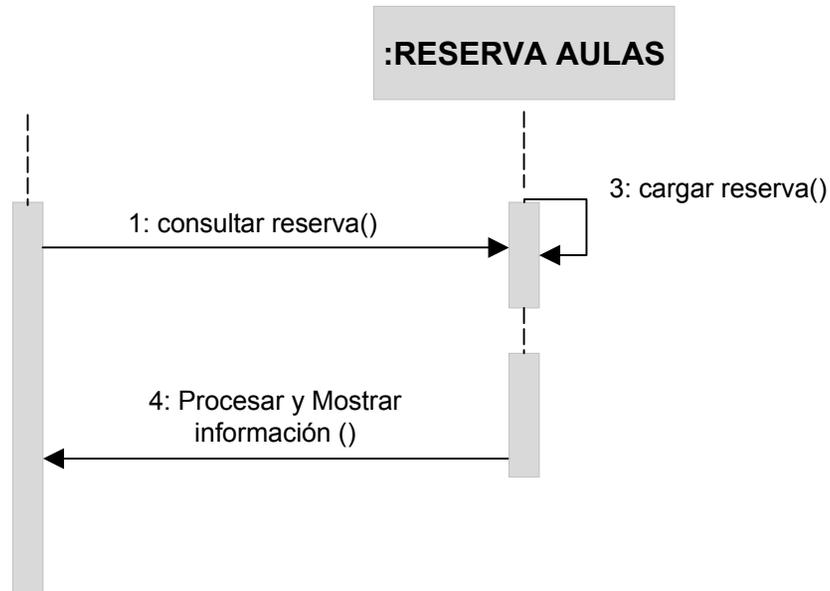
# RESERVAR AULAS



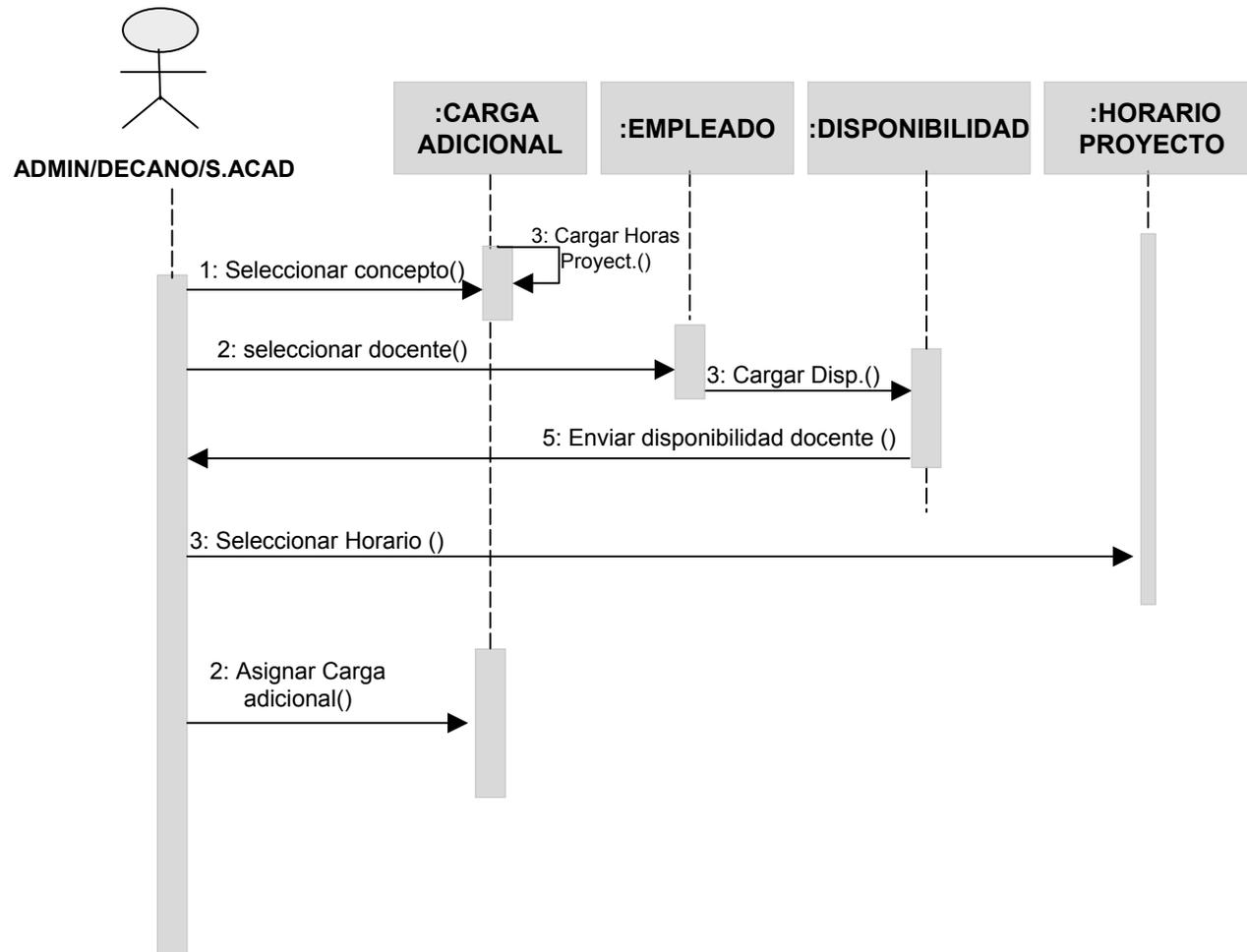
# ELIMINAR RESERVAS



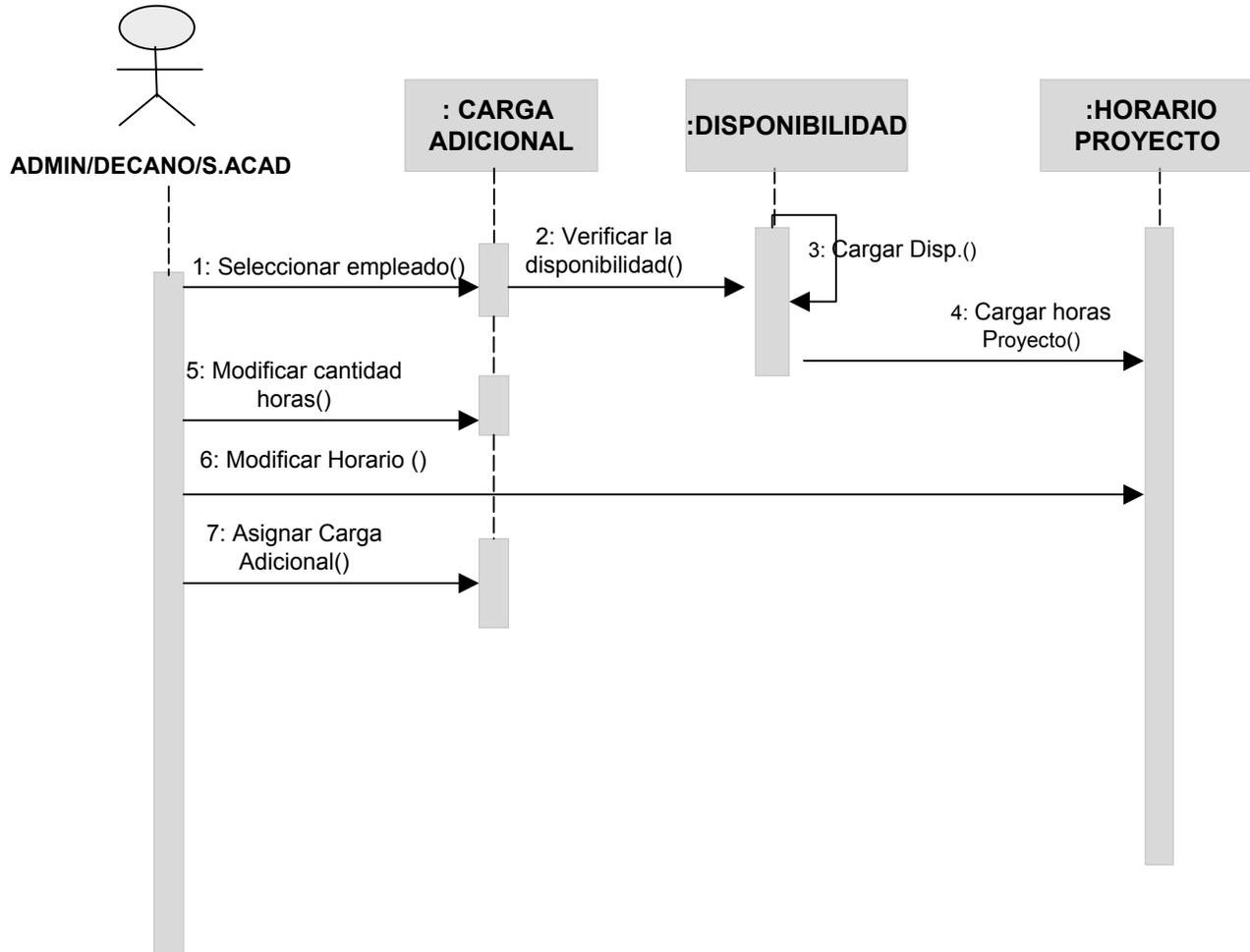
# CONSULTAR RESERVAS



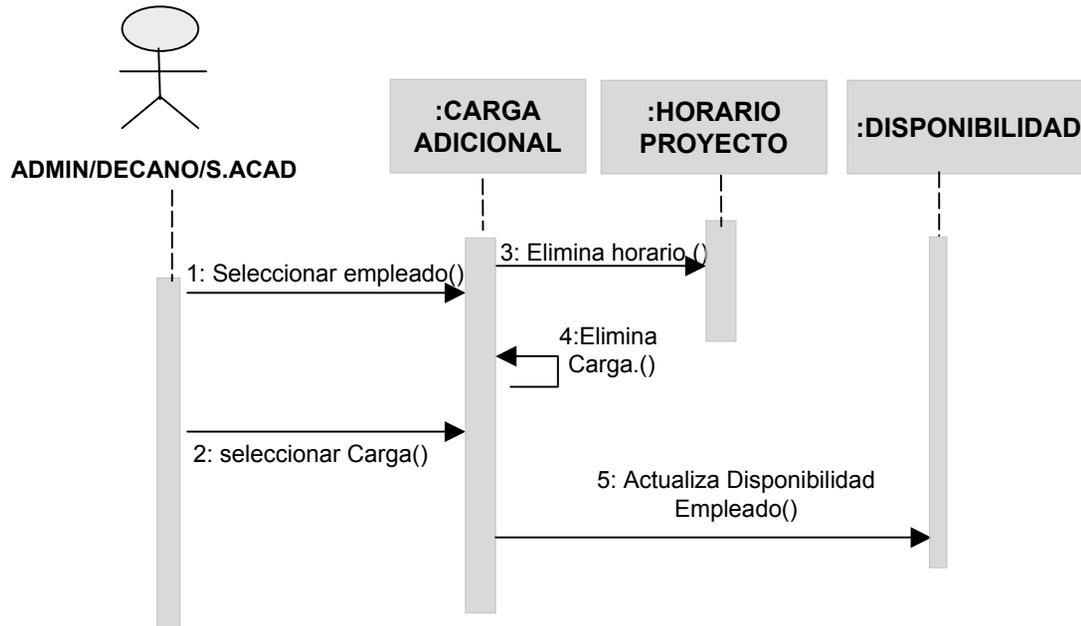
# ASIGNAR CARGA ADICIONAL ACADEMICOS



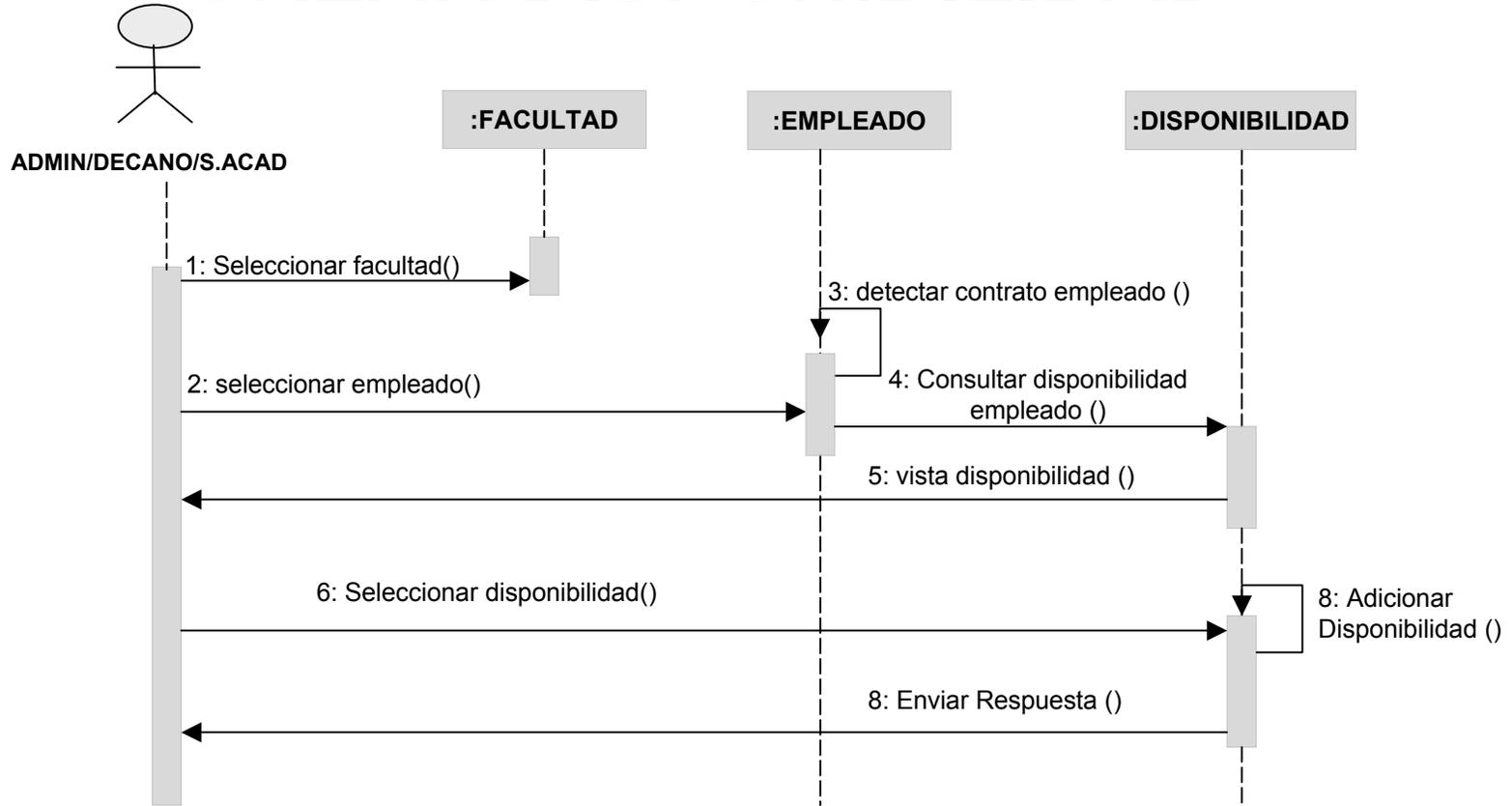
# MODIFICAR CARGA ADICIONAL ACADEMICOS



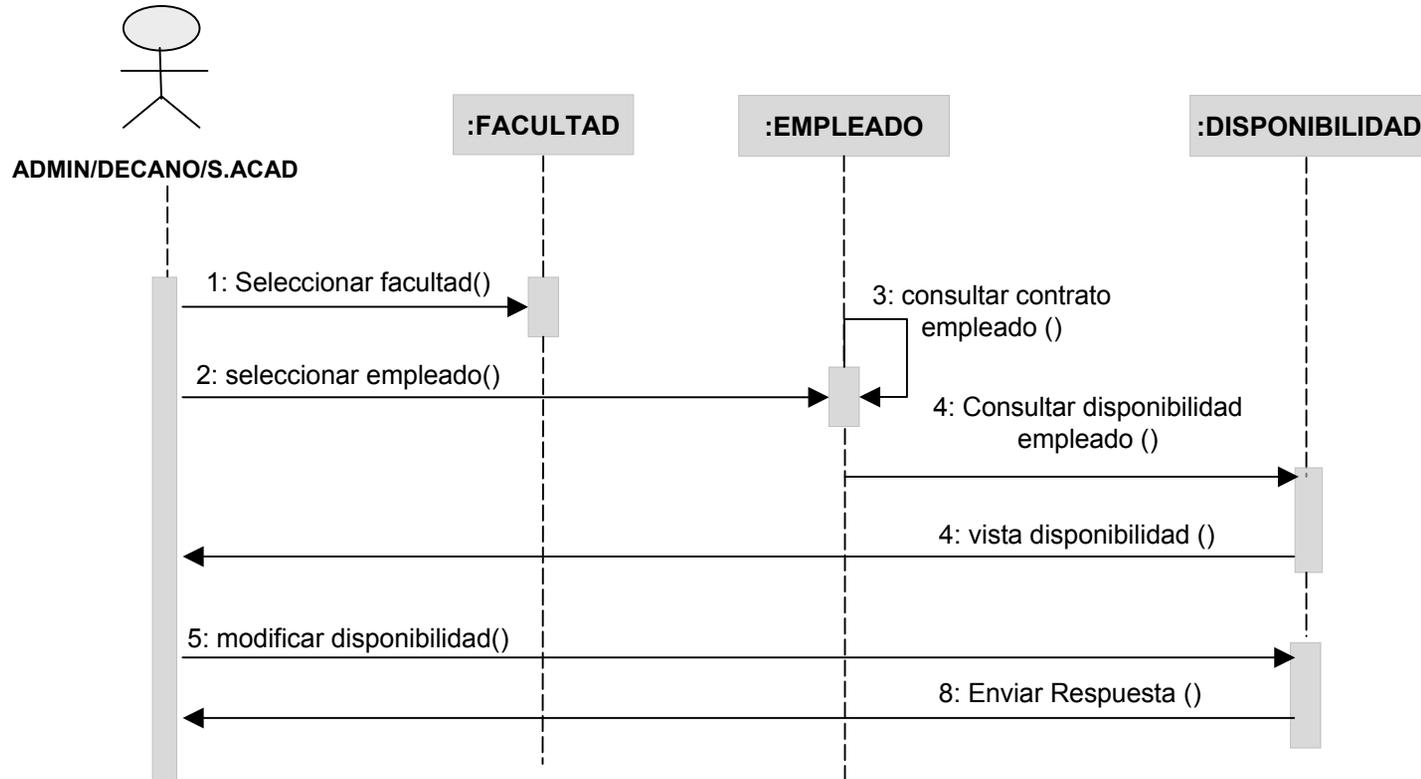
# ELIMINAR CARGA ADICIONAL ACADEMICOS



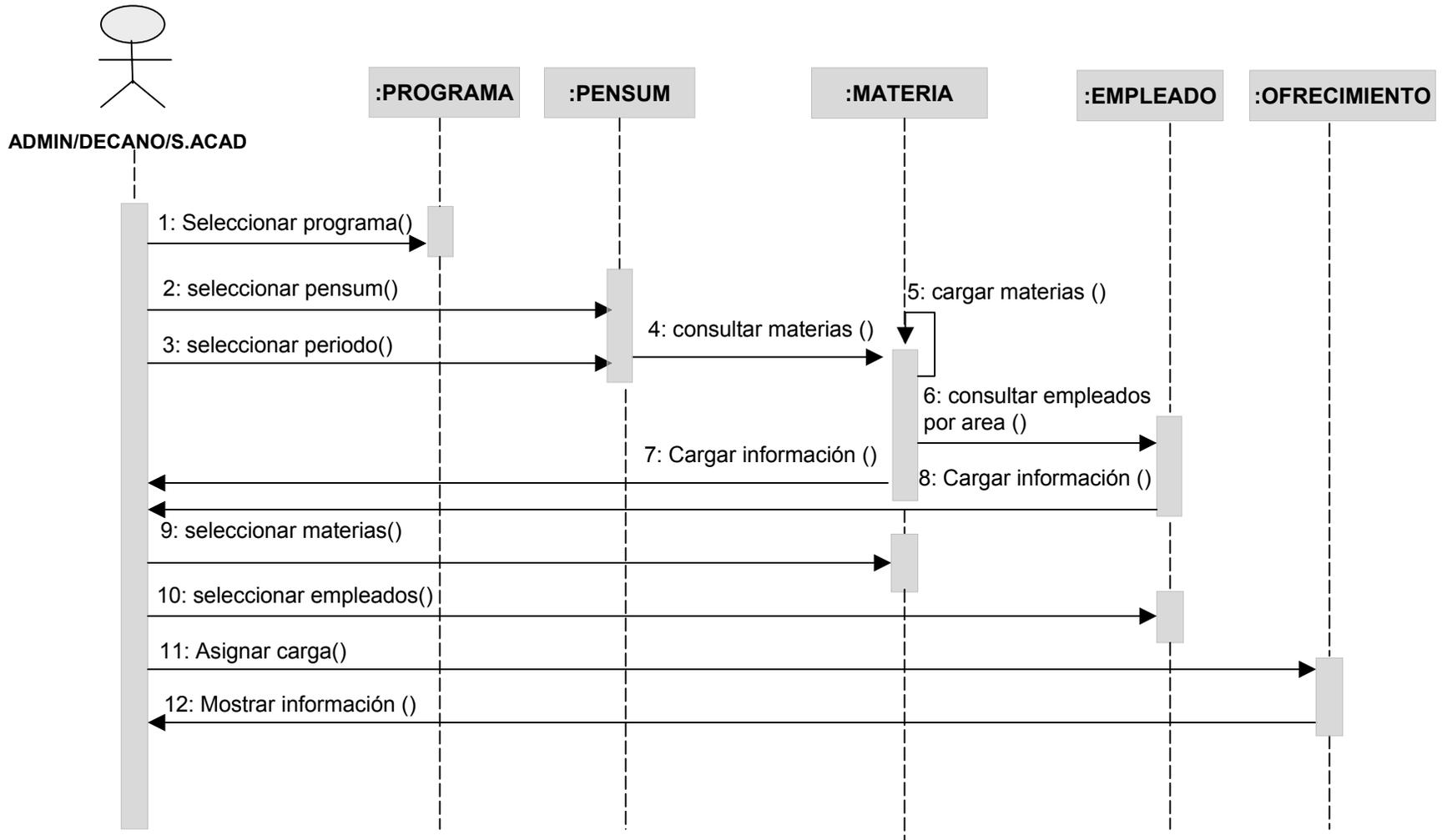
# CREAR DISPONIBILIDAD



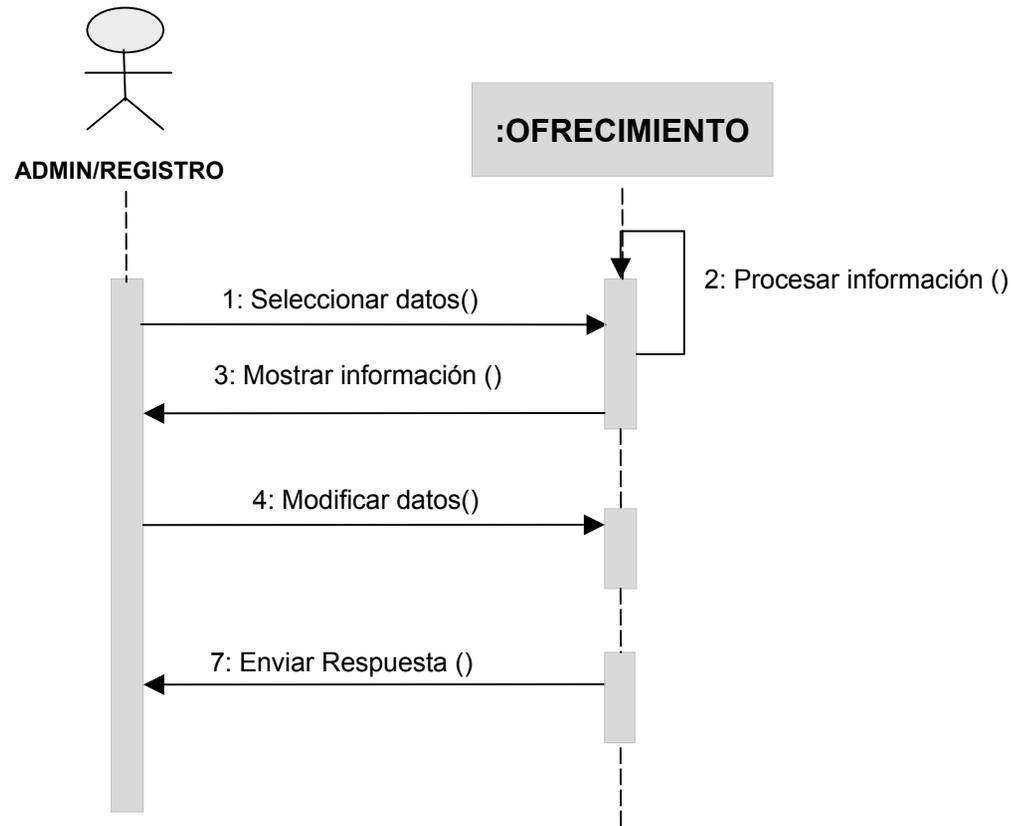
# MODIFICAR DISPONIBILIDAD



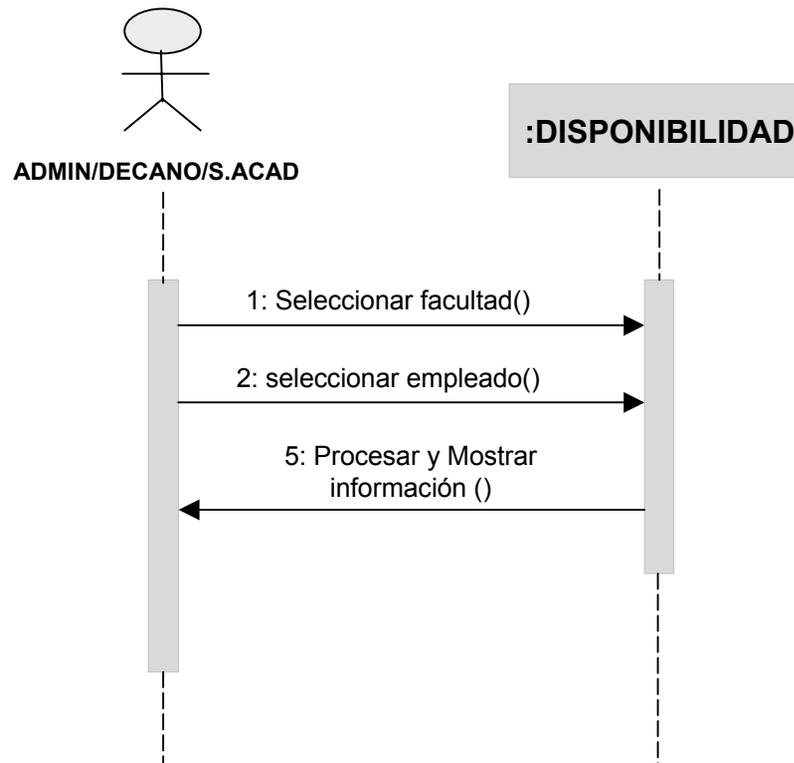
# CREAR OFRECIMIENTO



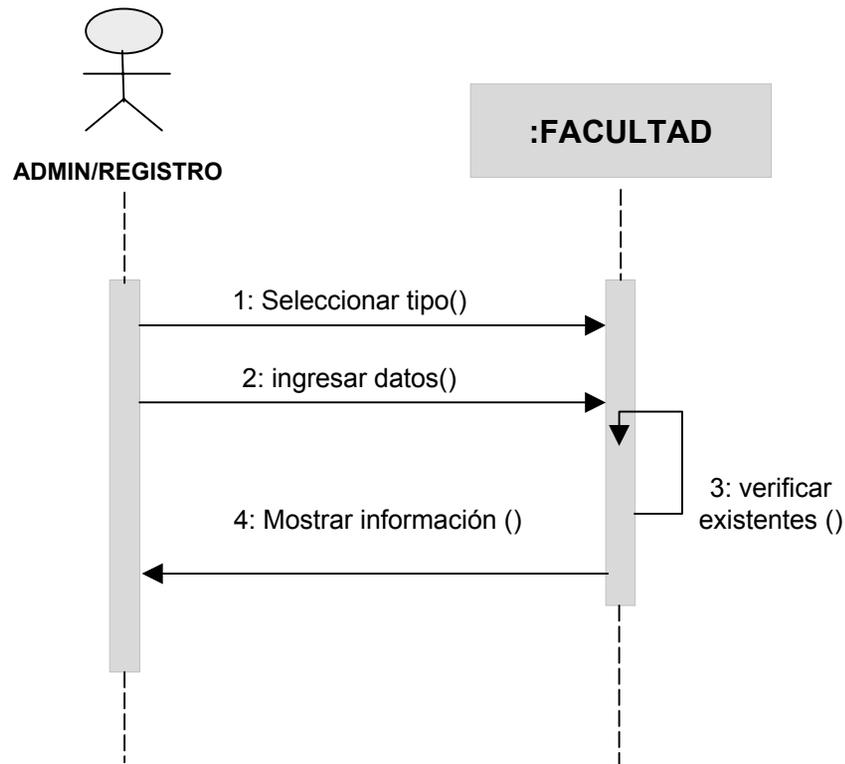
# MODIFICAR OFRECIMIENTO



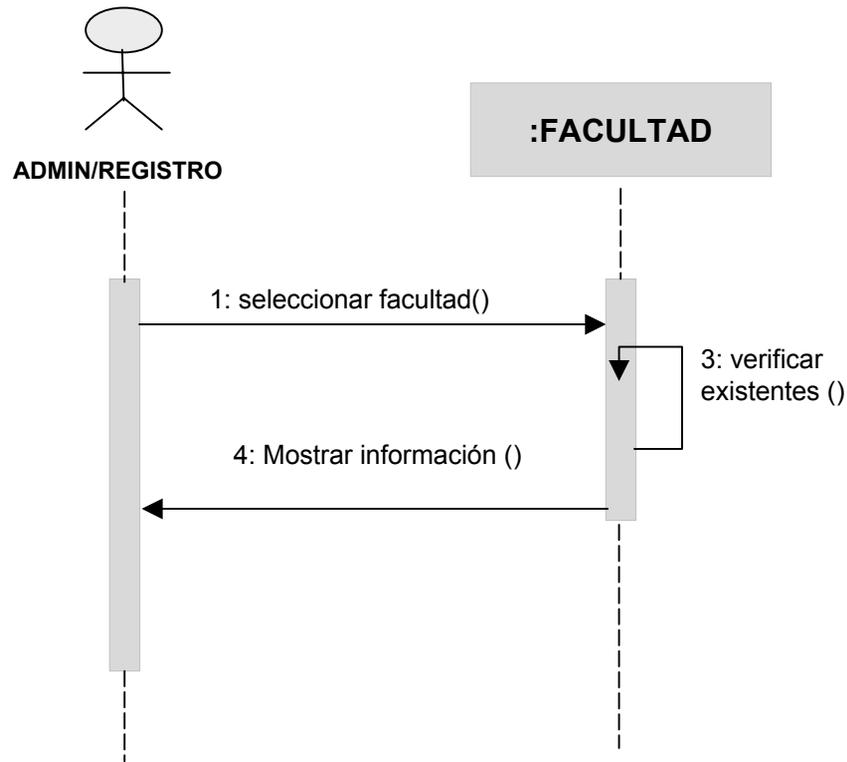
# CONSULTAR DISPONIBILIDAD



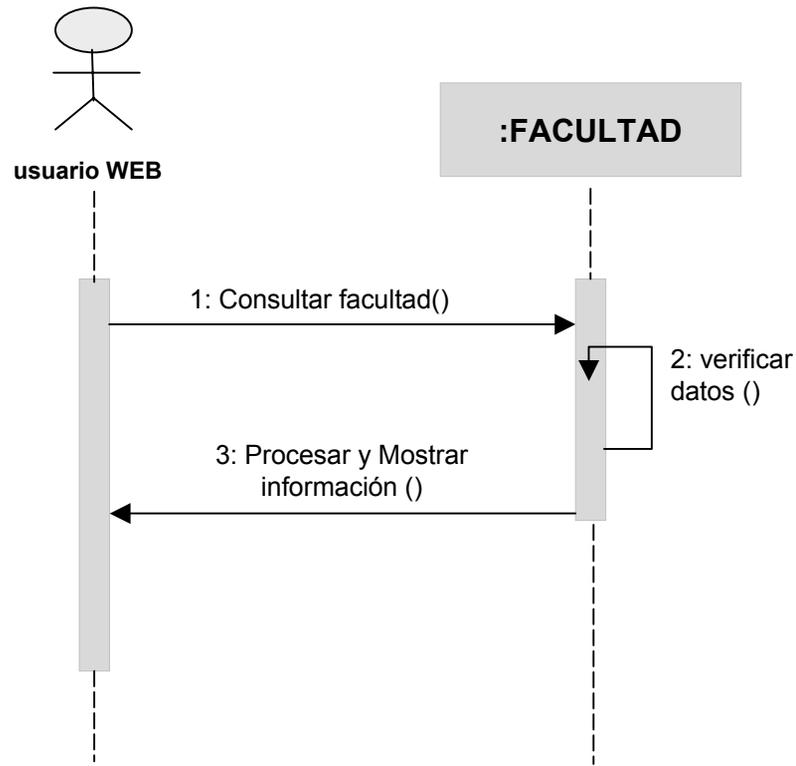
# CREAR FACULTAD



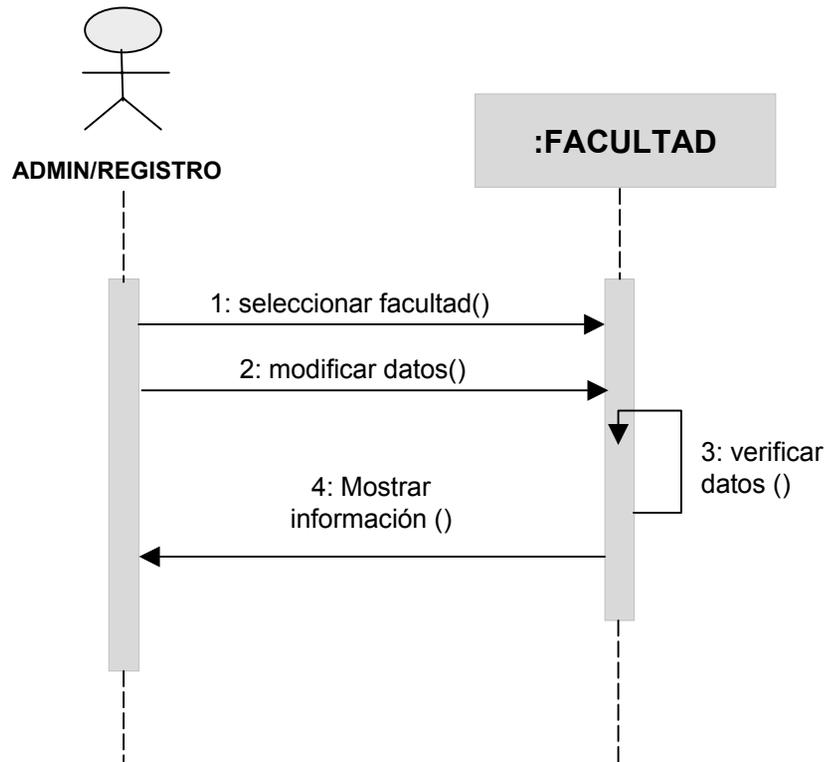
# ELIMINAR FACULTAD



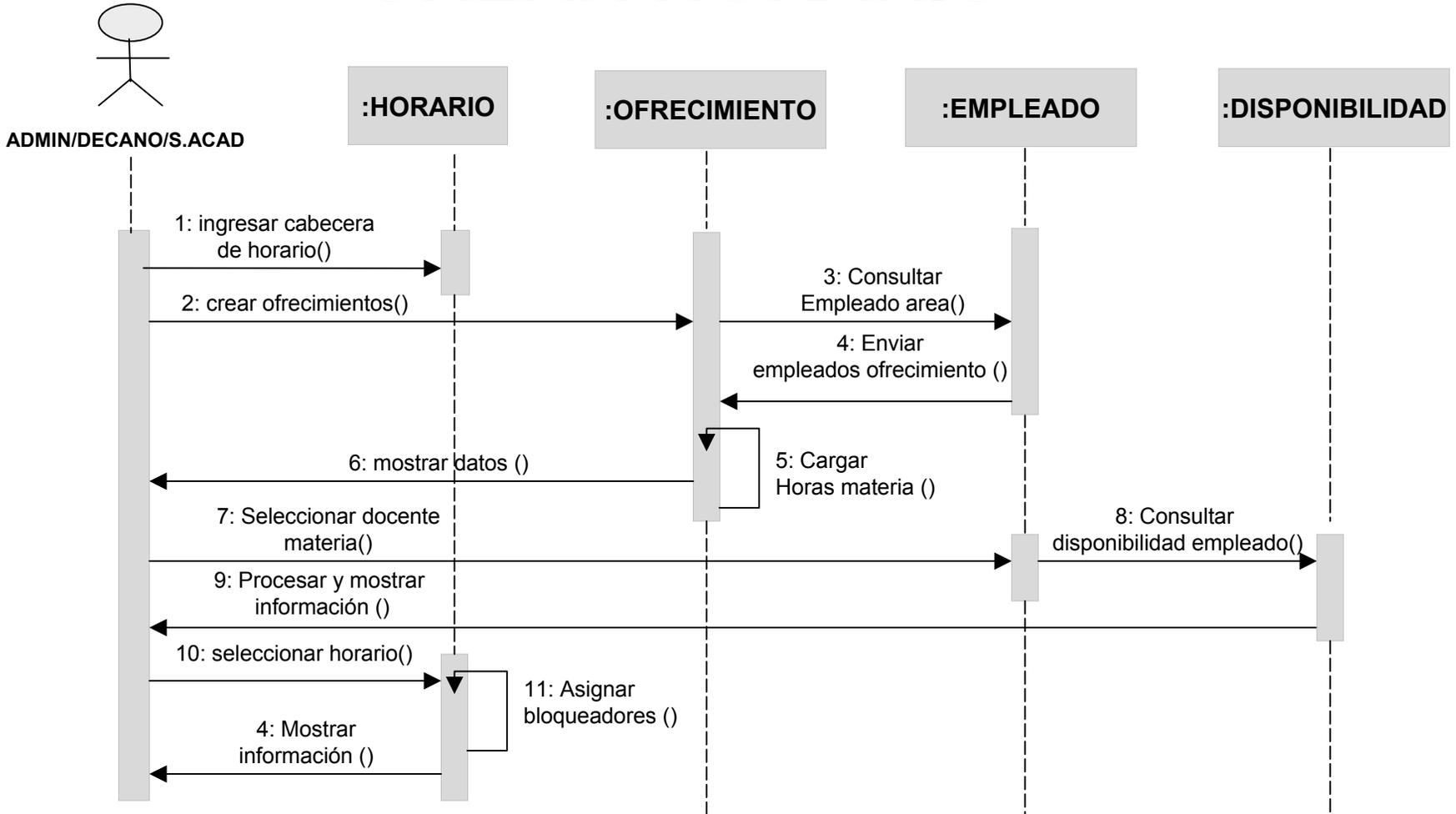
# CONSULTAR FACULTAD



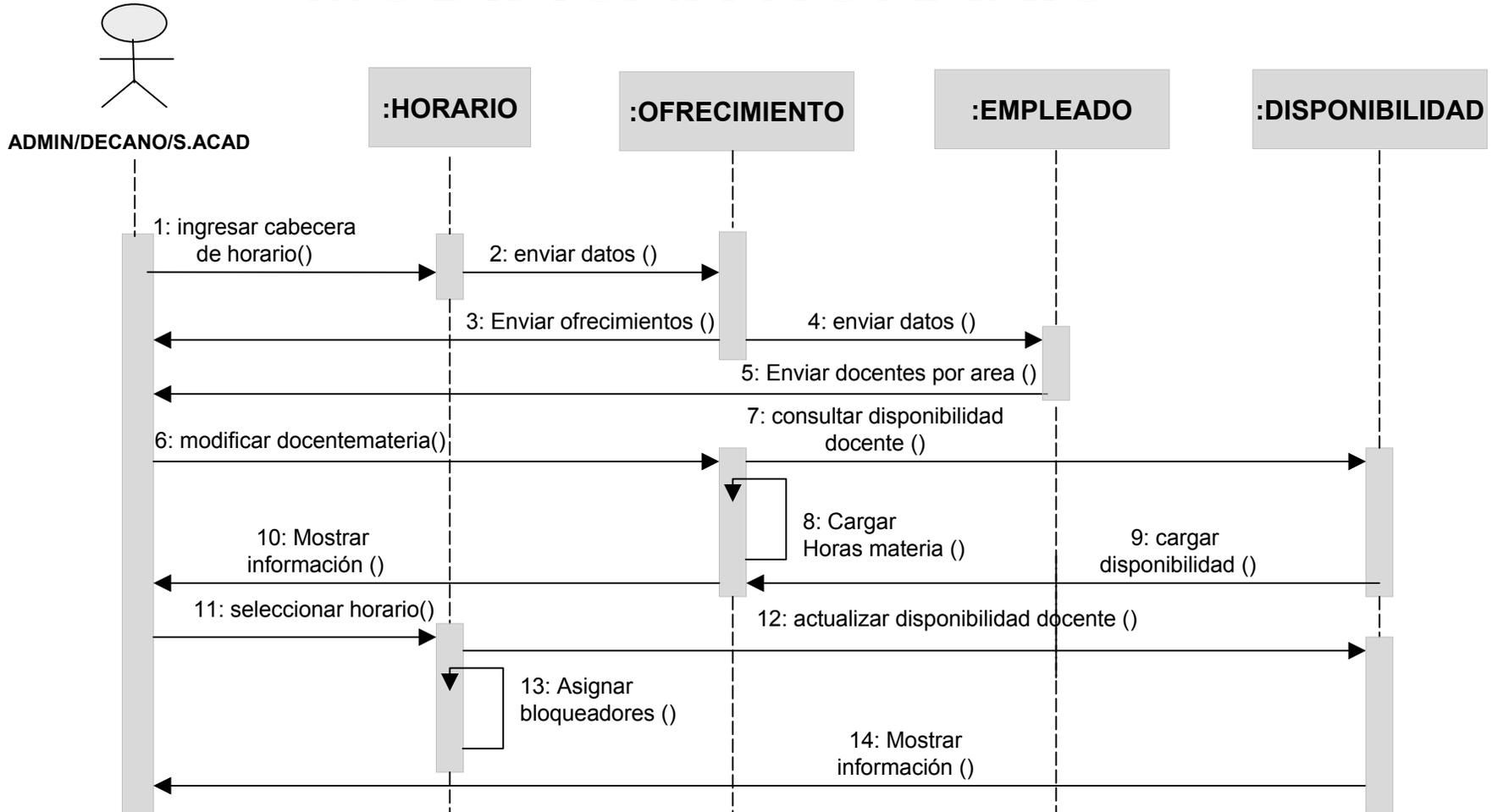
# MODIFICAR FACULTAD



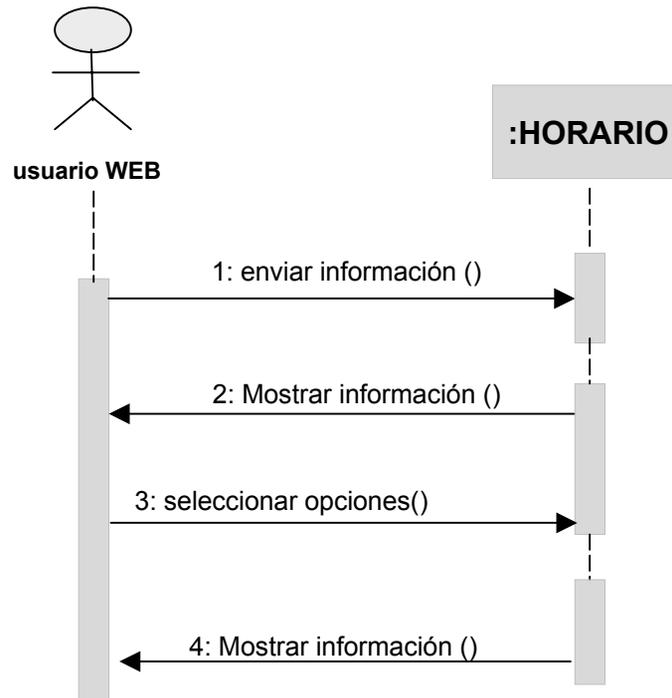
# CREAR HORARIO



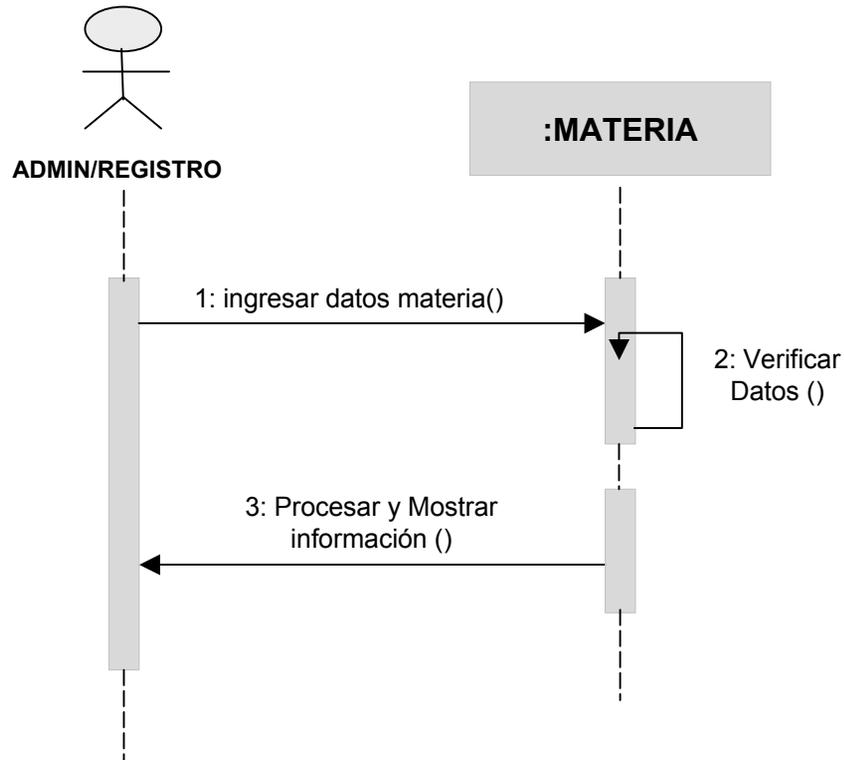
# MODIFICAR HORARIO



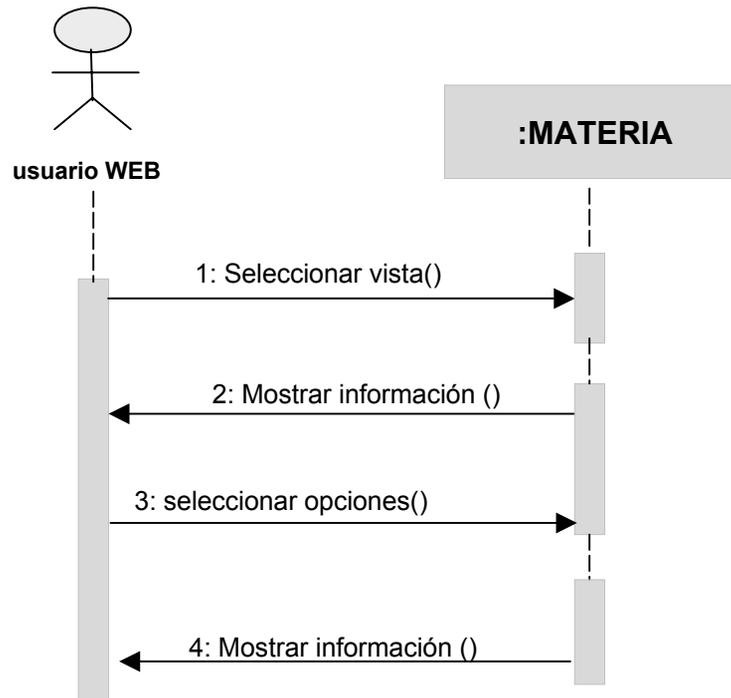
# CONSULTAR HORARIO



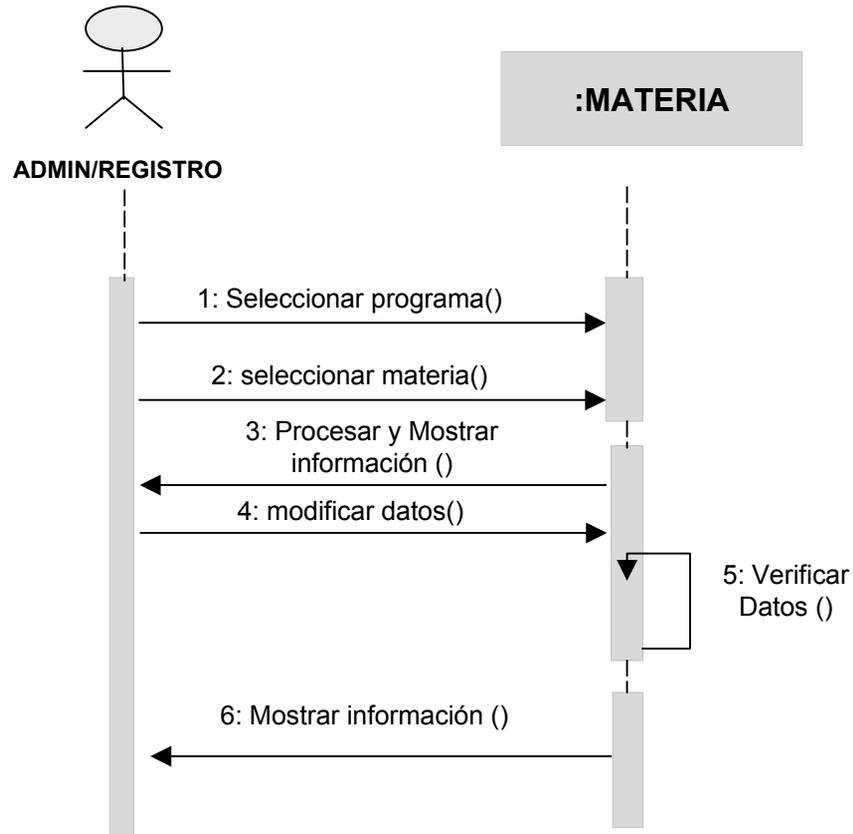
# CREAR MATERIA



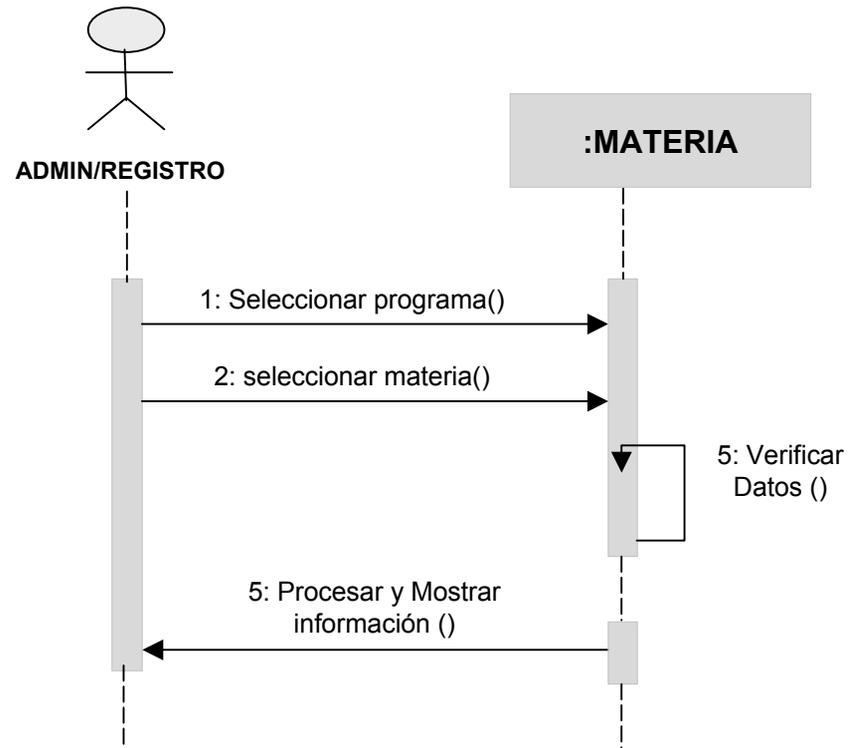
# CONSULTAR MATERIA



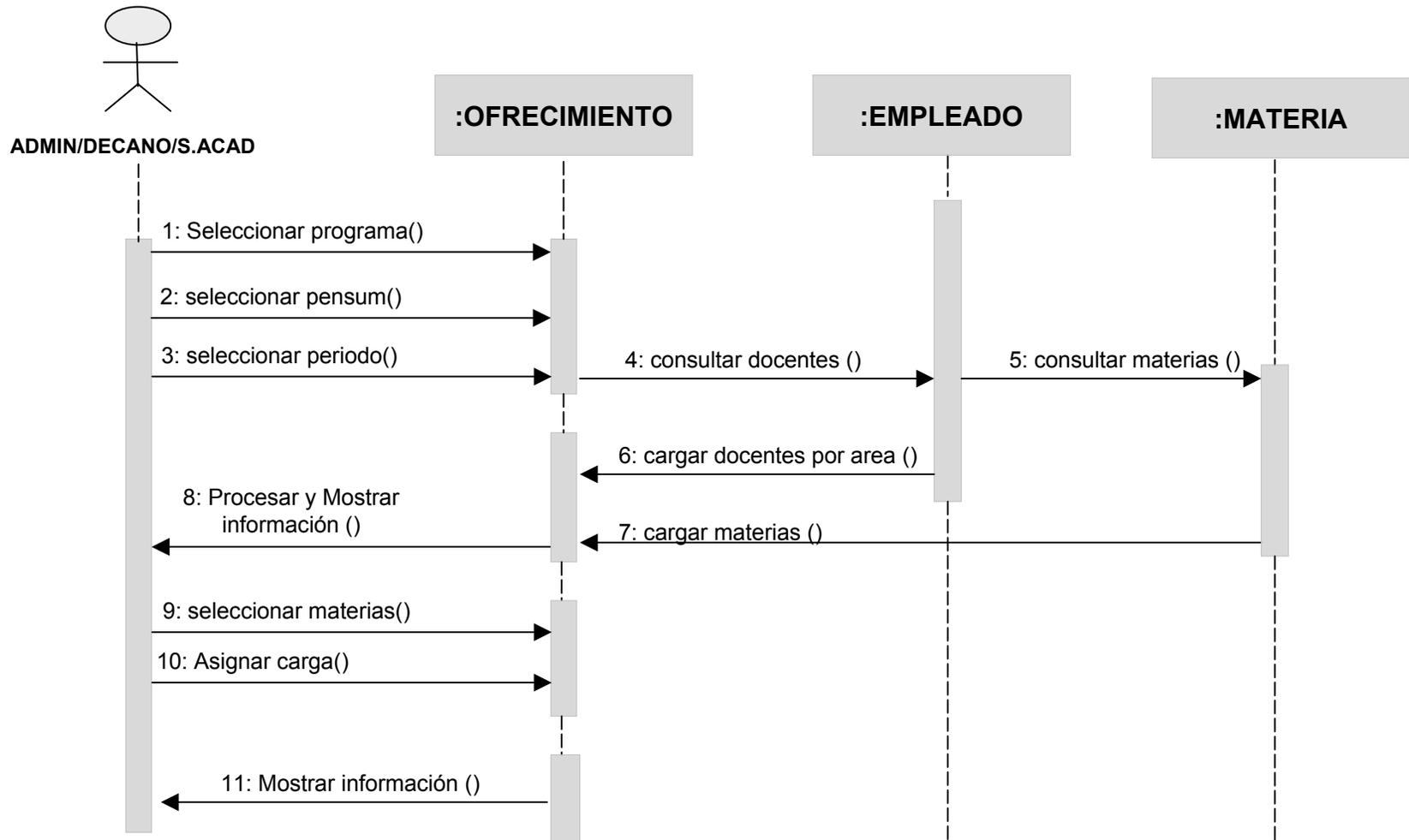
# MODIFICAR MATERIA



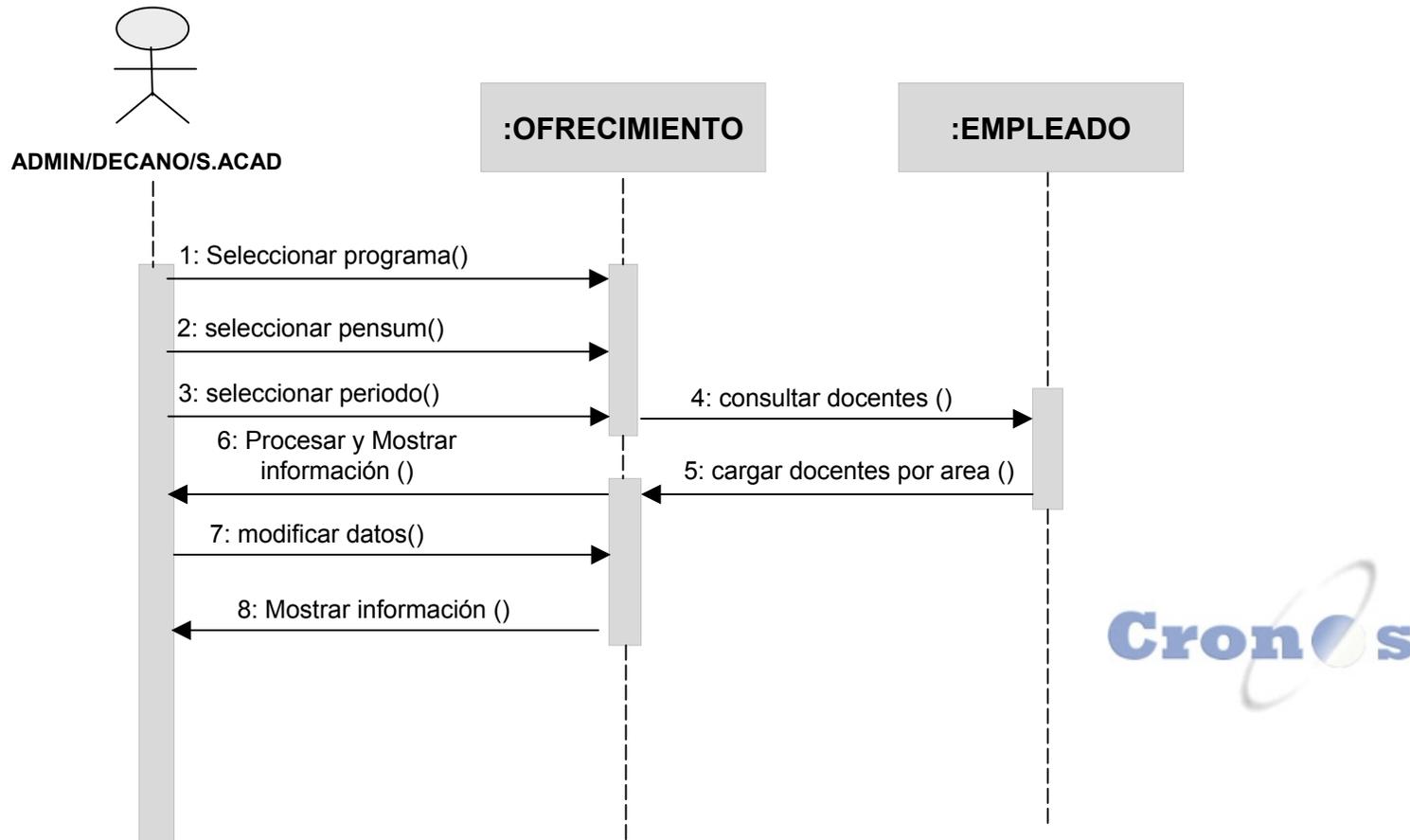
# ELIMINAR MATERIA



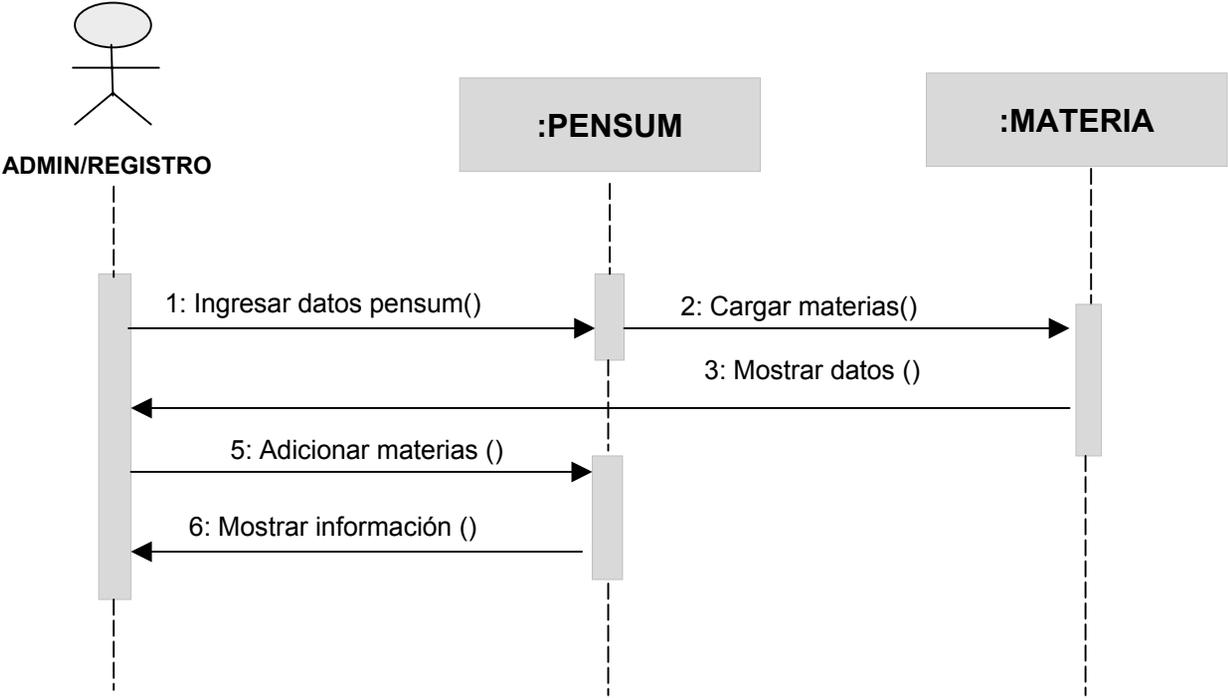
# CREAR OFRECIMIENTO



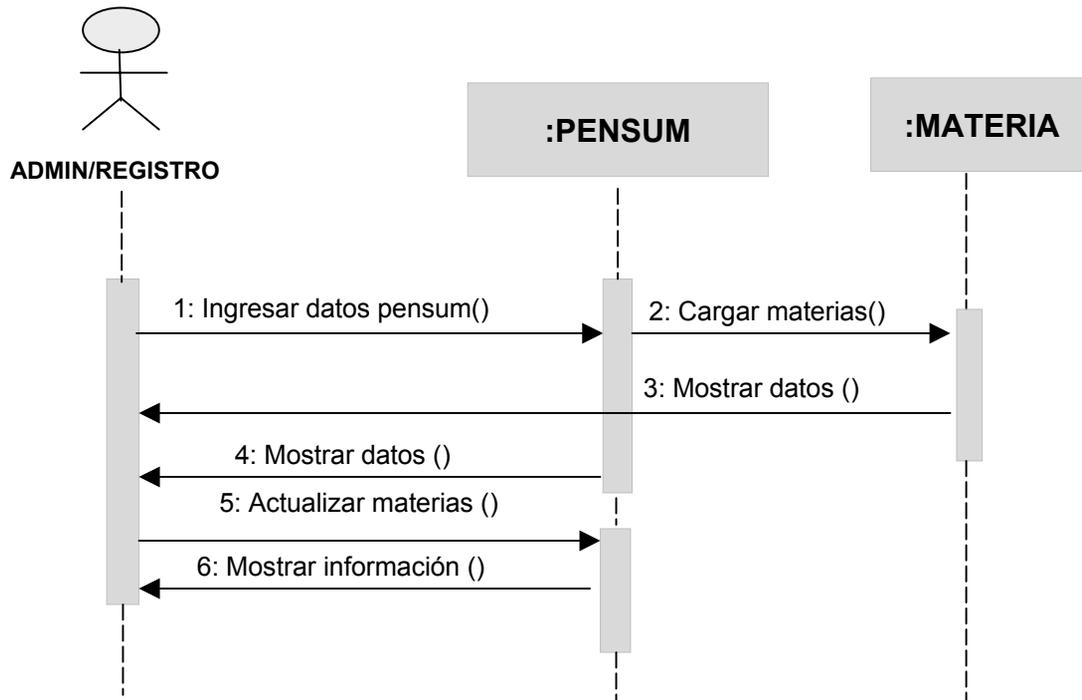
# MODIFICAR OFRECIMIENTO



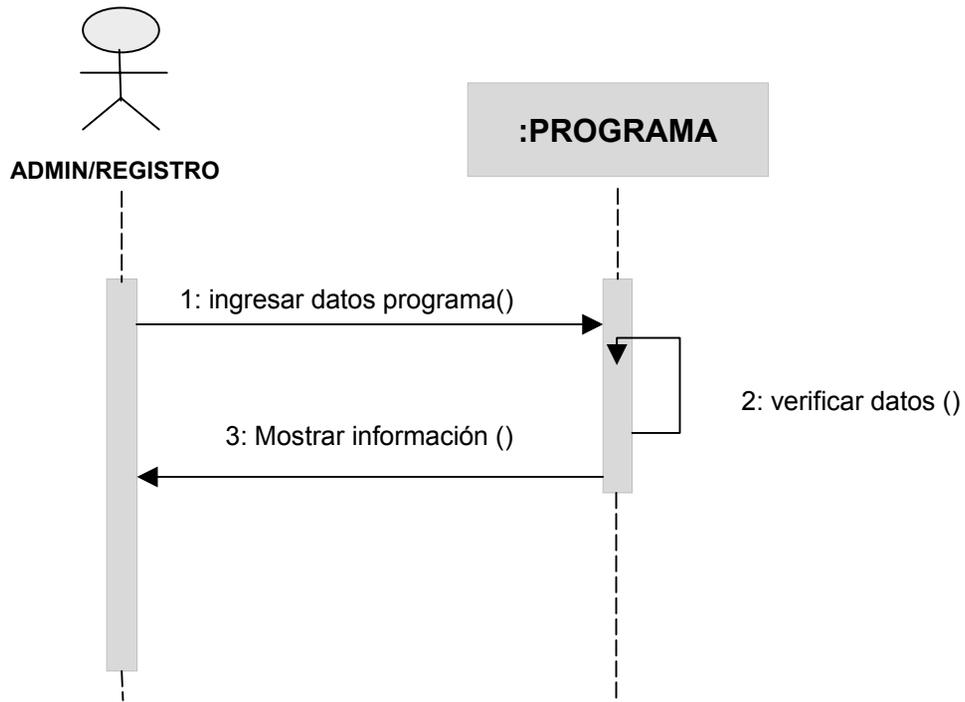
# CREAR PENSUM



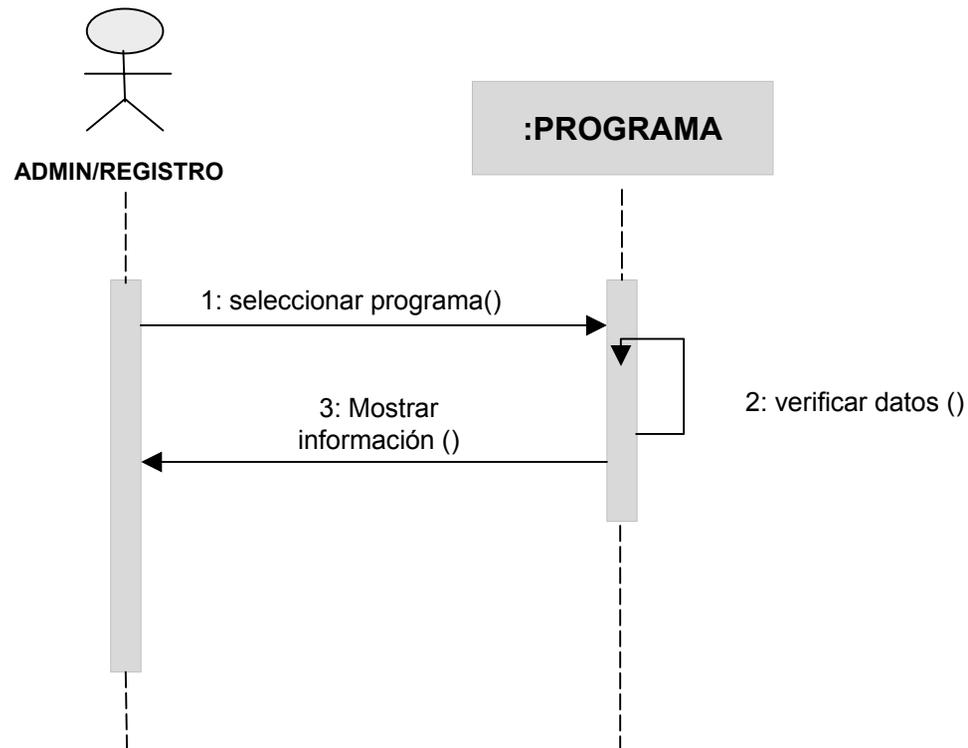
# MODIFICAR PENSUM



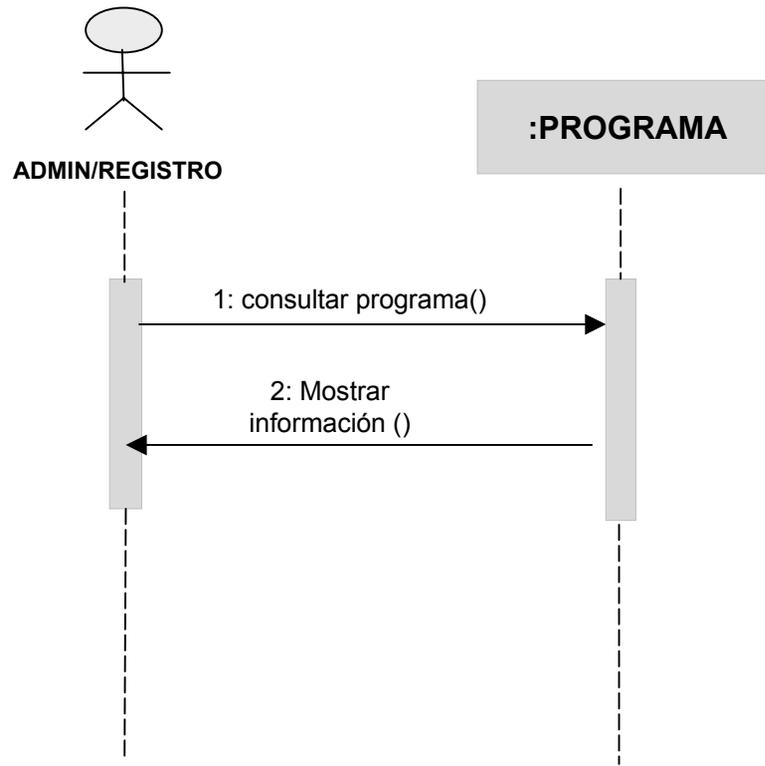
# CREAR PROGRAMA



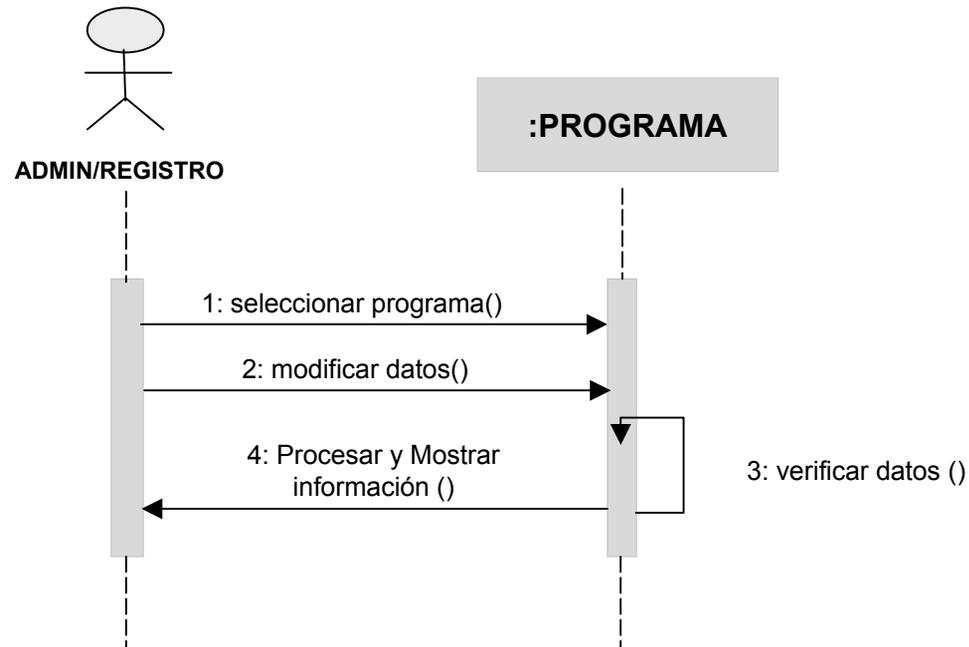
# ELIMINAR PROGRAMA



# CONSULTAR PROGRAMA



# MODIFICAR PROGRAMA



**ANEXO K**  
**MODELO FUNCIONAL**  
**INTRODUCCIÓN A LOS CASOS DE USO**

El Sistema CRONOS permitirá la asignación de la carga académica mediante la intervención un administrador del sistema, decano de facultad, secretario académico, secretaria de facultad, planeación y los docentes.

El administrador será el encargado de asignar la Primera clave principal por facultad(Decano) para ingresar al sistema y el tipo perfil que le permita acceder a un determinado menú. Es también en encargado de definir qué puede hacer un determinado perfil dentro del sistema, especificando por tabla que acciones son habilitadas.

El Decano tendrá la autonomía de decidir que materias y proyectos serán orientados por determinados docentes, que docentes son los encargados de administrar los grupos de investigación o extensión, asignándoles para estas actividades un determinado número de horas.

El Secretario académico será el encargado de generar los horarios según la asignación realizada por el Decano, teniendo en cuenta aspectos como la disponibilidad del docente, el horario de los prerrequisitos, entre otros.

Planeación realizará la asignación de aulas mediante la asistencia permanente del sistema que permitirá visualizar los requerimientos del aula y materia a ofrecer, para un adecuado uso de los recursos de la Universidad.

En todo momento el sistema asistirá al decano y al secretario académico para evitar cruces de docentes entre facultades, imposibilidades del docente en un horario determinado para la asignación de carga, cargas por encima del tipo de contrato del docente, entre otras posibles validaciones. También el sistema provee otros ayudantes como la creación de nuevos tipos de aulas y recursos académicos(audiovisuales, material didáctico e inmuebles), reserva de aulas y horarios para cada tipo de proyecto que se manejen en las facultades o dependencias.

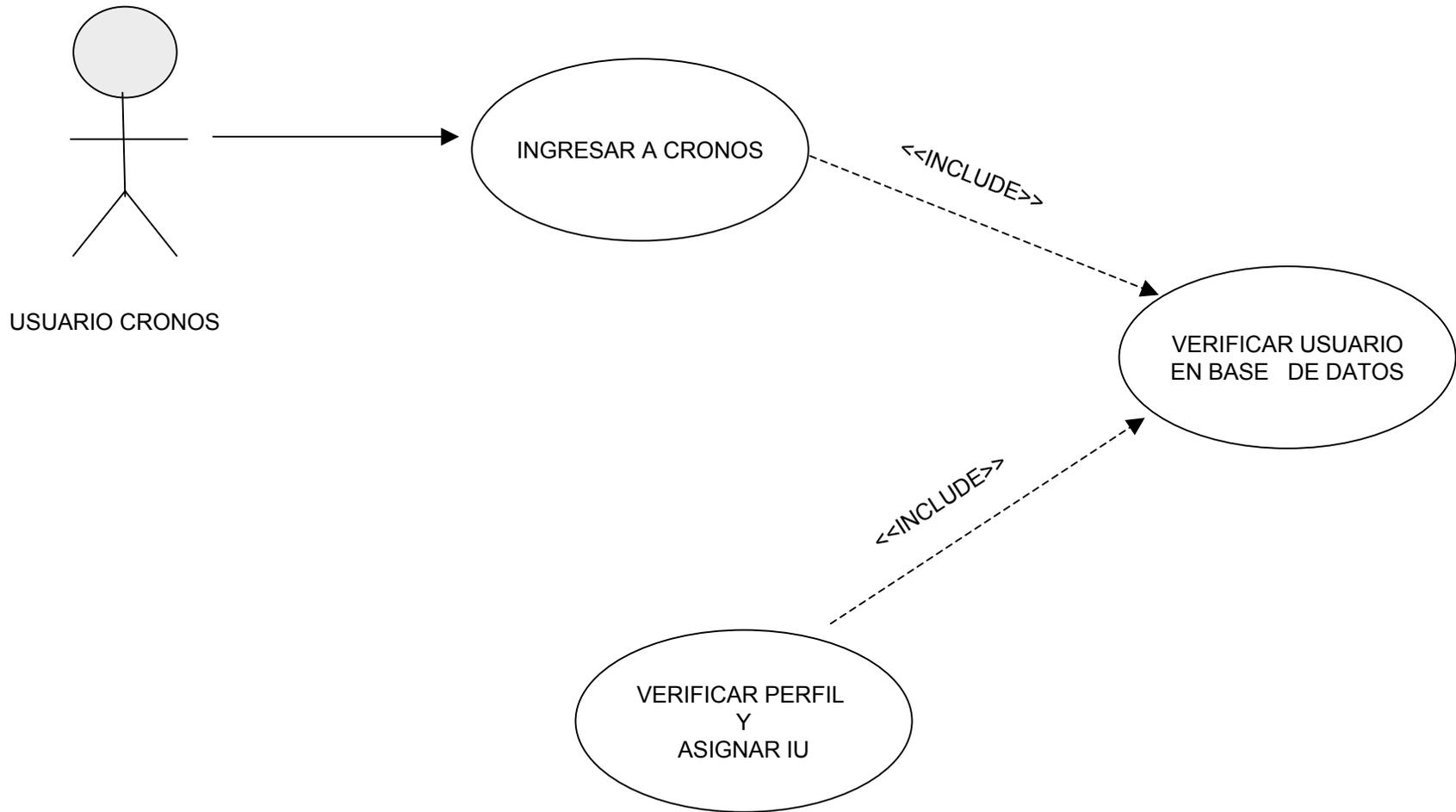
La existencia de Perfiles de usuario adicionales como Talento Humano, registro académico y planeación, permitirán algunas de las comunes tareas que ellos realizan al momento de crear las cargas académicas, asignar horarios, reservar aulas, estadísticas de empleados por escalafón y contrato, creación de áreas y materias, creación de nuevos programas y facultades e informes detallados de cómo queda finalmente la carga académica de los docentes y administrativos incluyendo horarios por proyecto y docentes.

**ANEXO L  
ANÁLISIS Y DISEÑO  
MODELO FUNCIONAL**

**CASOS DE USO  
CRONOS: SISTEMA DE CARGA ACADÉMICA Y PLANEACIÓN DE HORARIOS**



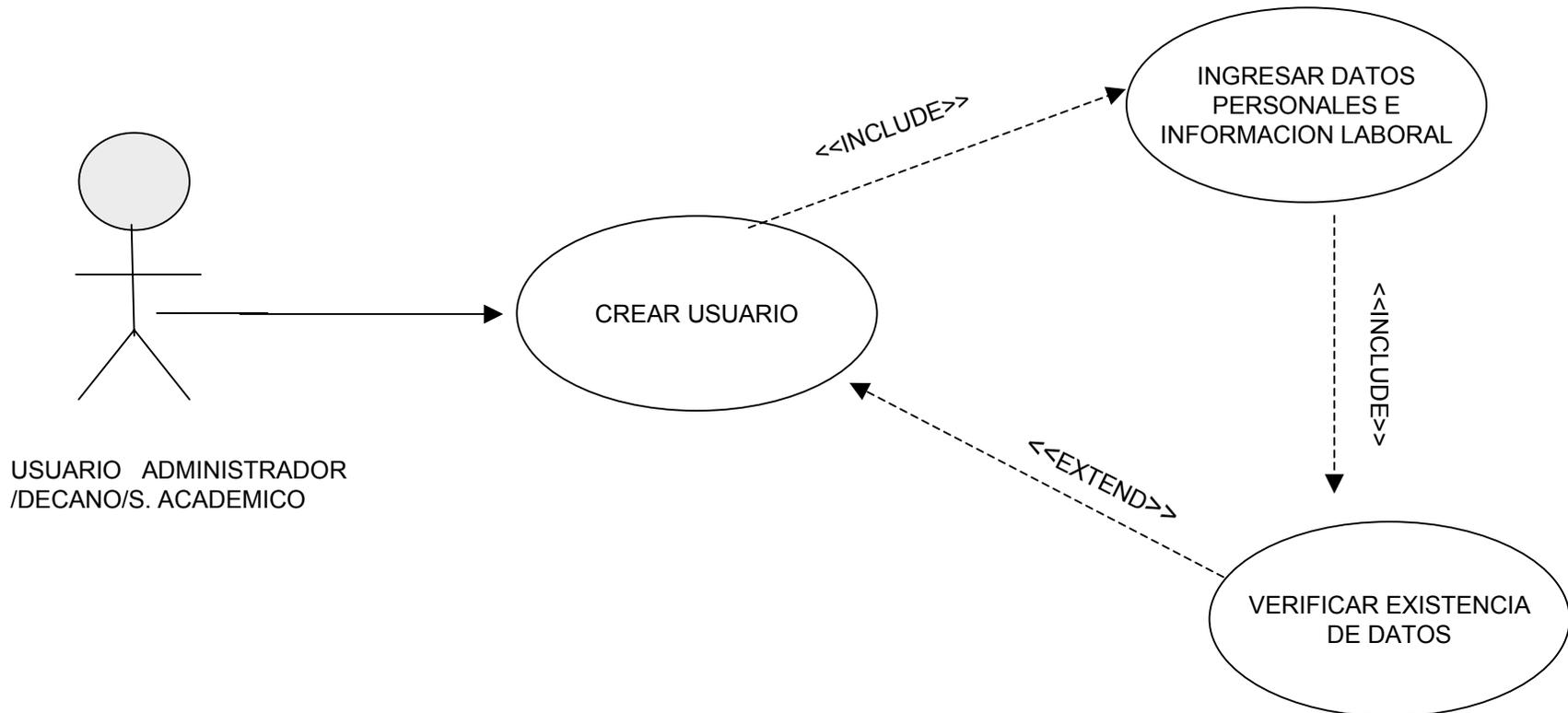
# ID:1. INGRESAR A CRONOS



# Tarjeta Descriptiva Caso de Uso 1

<b>Id caso de uso</b>	<b>1</b>
<b>Nombre caso de uso</b>	Ingresar a Cronos
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Empleado
<b>Descripción</b>	Cuando un usuario desea ingresar al sistema, debe ser autenticado.
<b>Precondición</b>	El usuario debe existir en la base de datos y seleccionar el periodo académico al cual desea acceder
<b>Postcondiciones</b>	Acceder a los servicios del sistema.
<b>Flujo Normal</b>	1. Ingresar usuario(que seria la cedula del empleado) 2. Ingresar la clave asignada por el sistema
<b>Flujos Alternativo</b>	3. Seleccionar periodo académico Se puede alterar el orden sin problema.
<b>Excepciones</b>	1.Si el usuario o la clave son erróneos se asigna una IU no autorizado 2.Si no selecciona periodo académico el sistema asignara una IU de error de acceso a base de datos. Pues en realidad no ha seleccionado ninguna
<b>Incluidos</b>	
<b>Prioridad</b>	ALTA
<b>Frecuencia de uso</b>	Cada vez que se desee acceder al sistema
<b>Requerimientos especiales</b>	Conexión al servidor web y base de datos

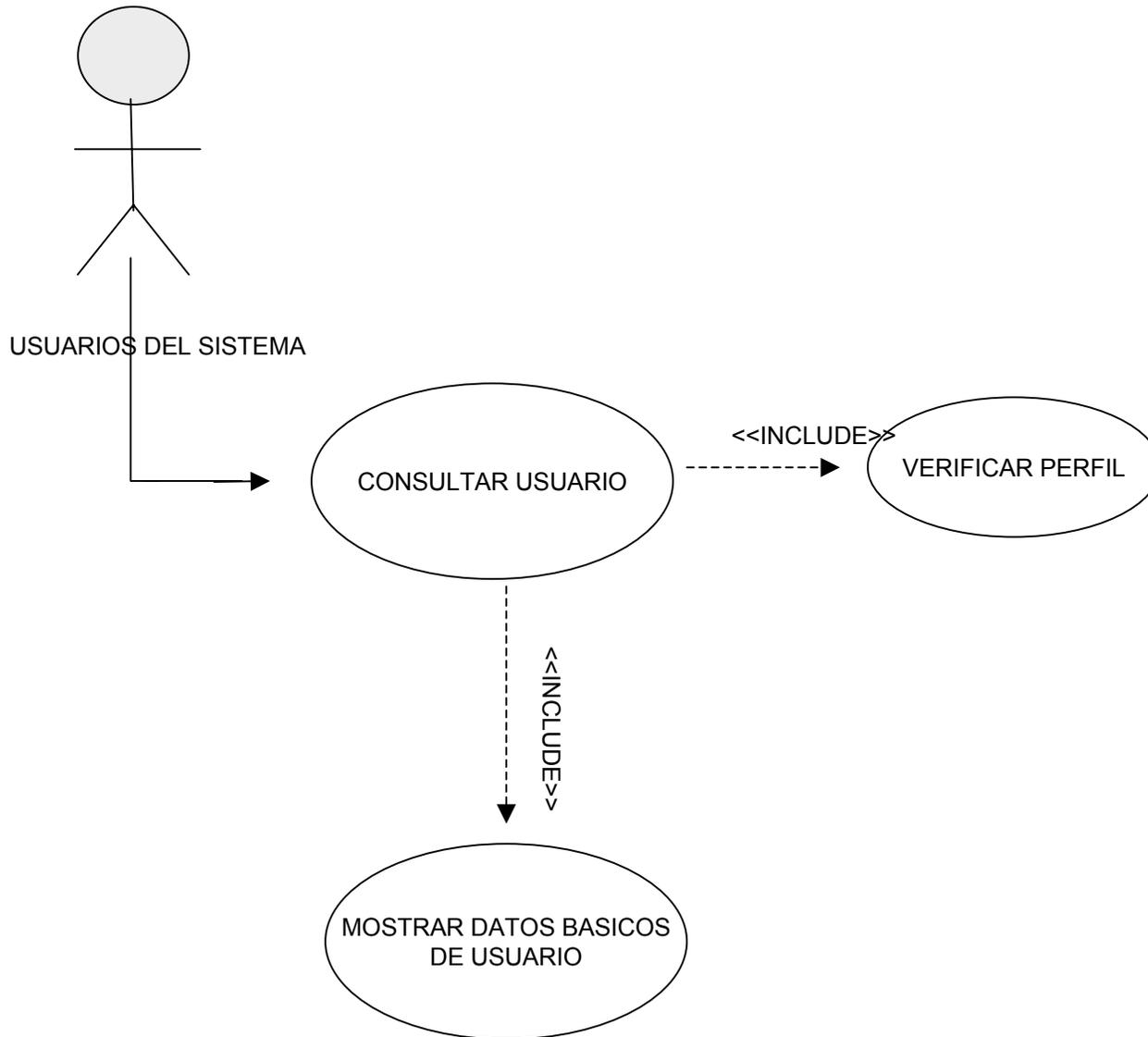
# ID:2. CREAR USUARIOS ADMINISTRATIVOS



## Tarjeta Descriptiva Caso de Uso 2

<b>Id caso de uso</b>	<b>2</b>
<b>Nombre caso de uso</b>	Crear Usuario
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decano, S. Académico
<b>Descripción</b>	El empleado es registrado en el sistema. Durante el proceso de registro se selecciona el perfil al cual va a pertenecer el usuario acorde a su vinculación laboral y tareas propias del empleado. Después de esto el sistema se encargara de crear dinámicamente las IU's donde el usuario podrá acceder.
<b>Precondición</b>	Validar permisos de acceso a creación de usuario.
<b>Postcondiciones</b>	Acceder a las funciones asignadas al perfil perfil.
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. El usuario creador de empleados ingresa los datos del nuevo usuario en el sistema.</li><li>2. El sistema confirma el ingreso de datos a través de validaciones de campos</li><li>3. Envío de mensaje de bienvenida, usuario y contraseña al e-mail del nuevo usuario Cronos(empleado)</li></ol>
<b>Flujos Alternativo</b>	Paso 1. Si el sistema detecta que no puede crear usuarios, este habilita una interfaz de usuario de error.
<b>Excepciones</b>	Si algún campo obligatorio del formulario de registro falta o es digitado de una manera errónea(ej: campo numérico se colocan números) el sistema no lo dejara proceder con el registro hasta que no lo modifique.
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Cada que se deseen crear nuevos usuarios
<b>Requerimientos especiales</b>	El Empleado que desee crear nuevos usuarios debe tener los permisos necesarios, los cuales fueron asignados en el momento de asignarle su perfil.

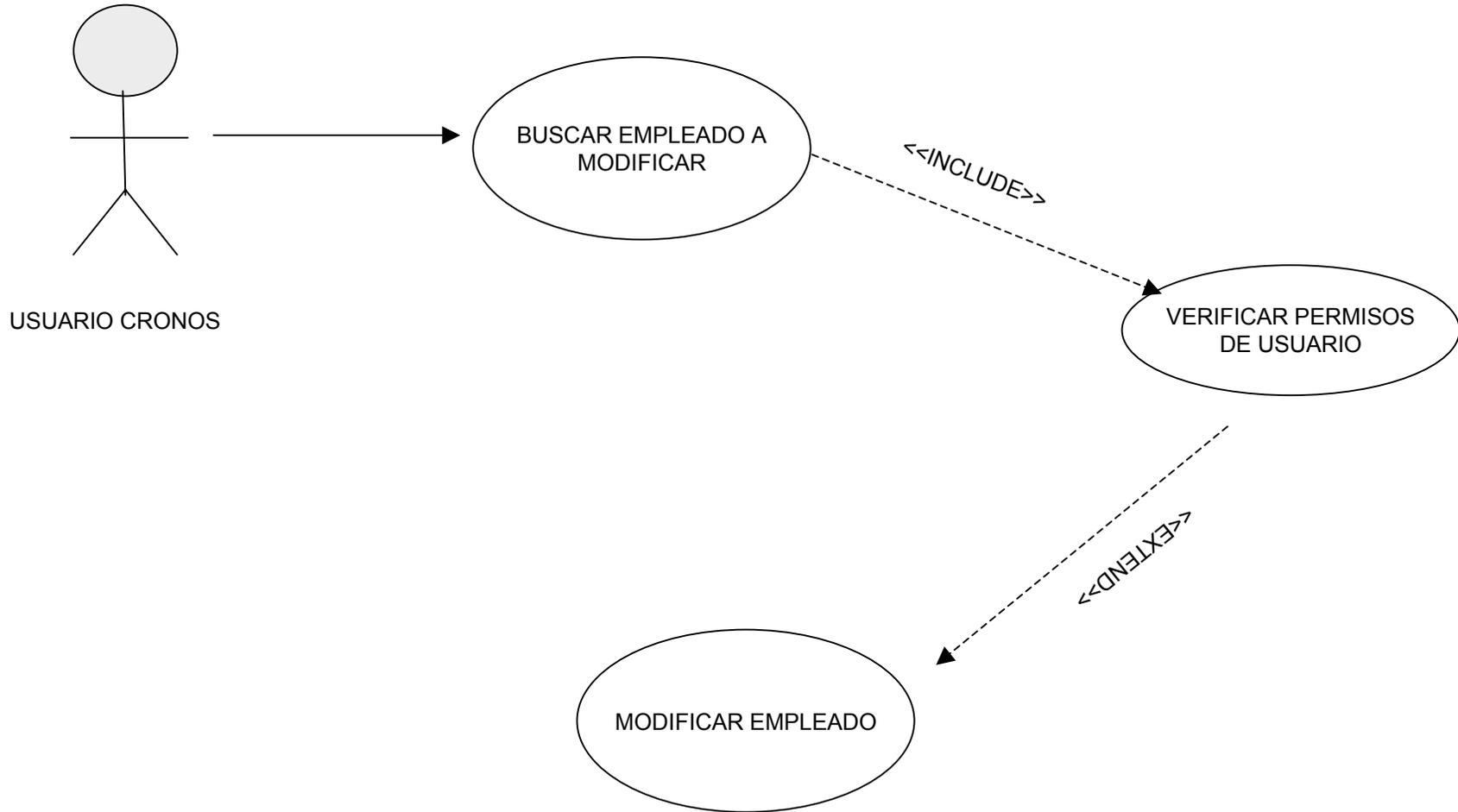
# ID:3. CONSULTAR USUARIOS



# Tarjeta Descriptiva Caso de Uso 3

<b>Id caso de uso</b>	<b>3</b>
<b>Nombre caso de uso</b>	Consultar Usuario
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Usuarios Cronos
<b>Descripción</b>	Muestra los usuarios del sistema
<b>Precondición</b>	Tener Permisos de Consulta sobre objeto
<b>Postcondiciones</b>	Mostrar Datos acerca de su vinculación laboral
<b>Flujo Normal</b>	1 Buscar Usuario 2 Cargar datos usuario
<b>Flujos Alternativo</b>	1. Vista rápida de Usuario 2. Seleccionar usuario para ver toda la información
<b>Excepciones</b>	IU de error si el usuario no tiene permisos sobre el objeto y quiere acceder
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	Tener permisos de consulta sobre el tipo de usuario que desea consultar.

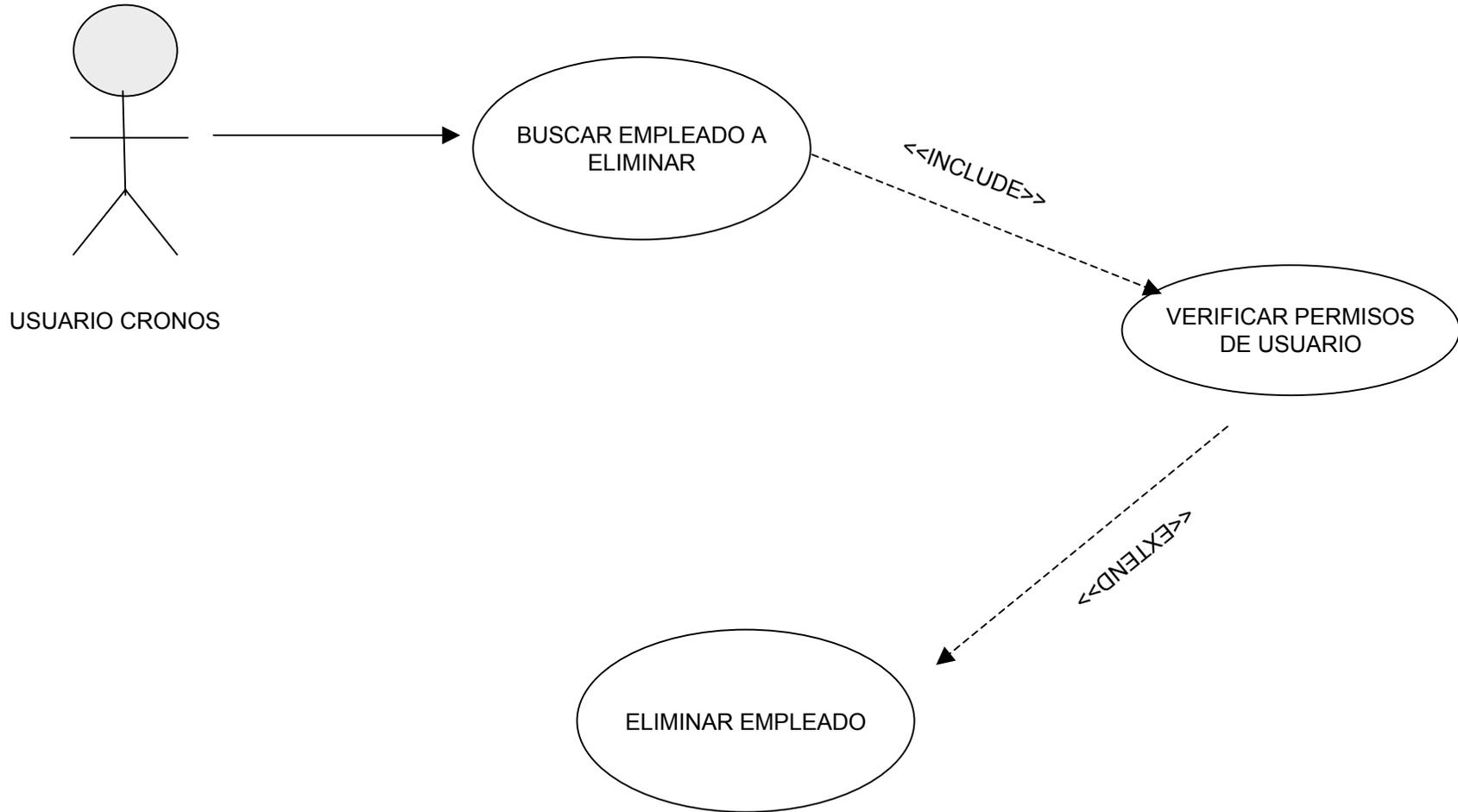
# ID:4. MODIFICAR USUARIOS



# Tarjeta Descriptiva Caso de Uso 4

<b>Id caso de uso</b>	<b>4</b>
<b>Nombre caso de uso</b>	Modificar Usuario
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador. Decano. Sec Académico
<b>Descripción</b>	Modifica los usuarios del sistema
<b>Precondición</b>	Tener Permisos de modificar sobre objeto
<b>Postcondiciones</b>	Permitir Modificar datos del empleado
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1.Buscar Usuario</li><li>2.Cargar datos</li><li>3.Modificar datos</li></ol>
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	IU de error si el usuario no tiene permisos sobre el objeto y quiere acceder
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	Tener permisos de modificar sobre el objeto.

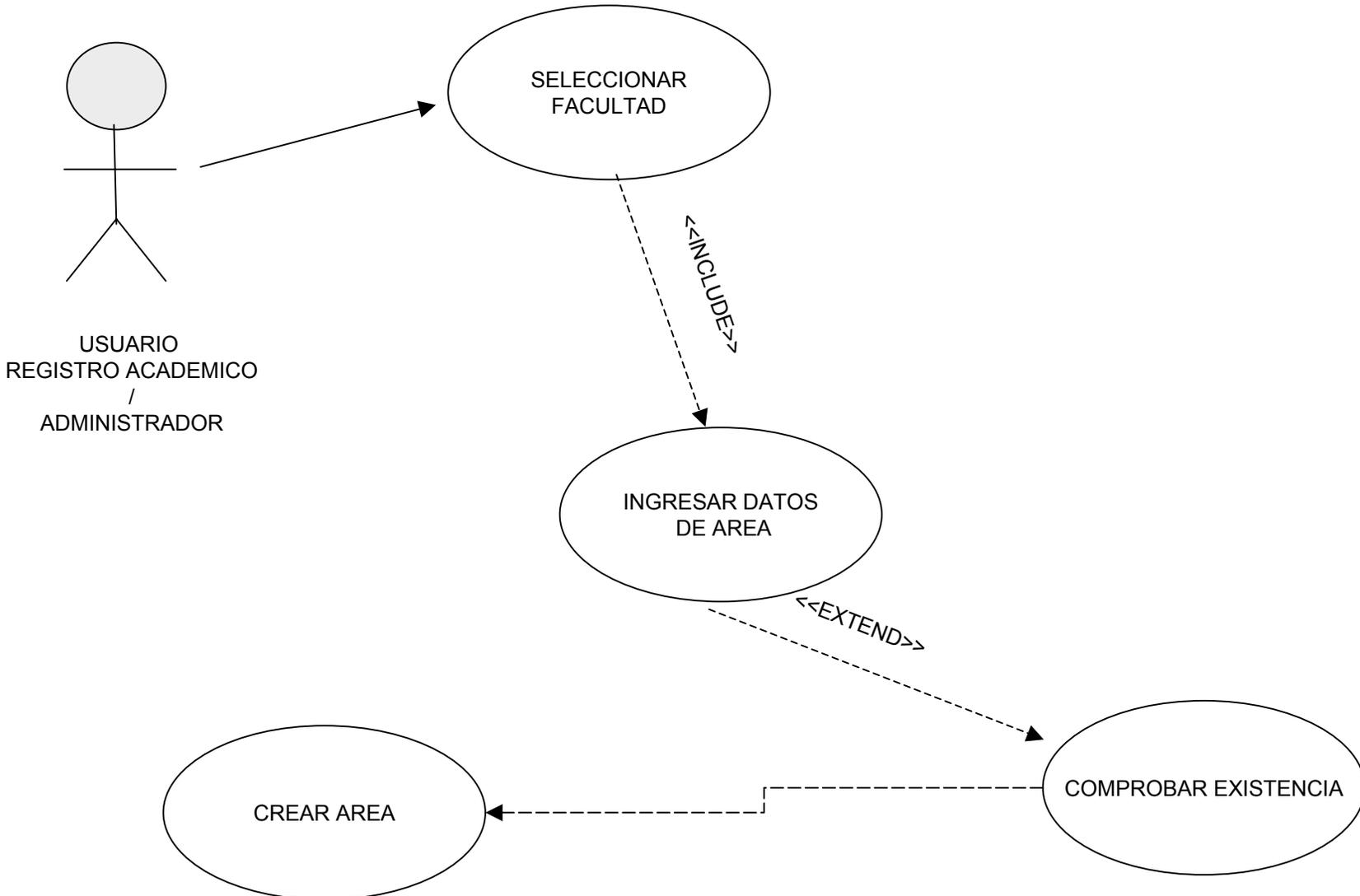
# ID:5. ELIMINAR USUARIOS



# Tarjeta Descriptiva Caso de Uso 5

<b>Id caso de uso</b>	<b>5</b>
<b>Nombre caso de uso</b>	Eliminar Usuario
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador. Decano. Sec Académico
<b>Descripción</b>	Elimina los usuarios del sistema
<b>Precondición</b>	Tener Permisos de eliminar sobre objeto
<b>Postcondiciones</b>	Permitir eliminar datos del empleado
<b>Flujo Normal</b>	Buscar Usuario 1.Cargar datos 2.Eliminar usuario
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	IU de error si el usuario no tiene permisos sobre el objeto y quiere acceder. No deja eliminar si del usuario dependen acciones del sistema
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	Tener permisos de eliminar sobre el objeto.

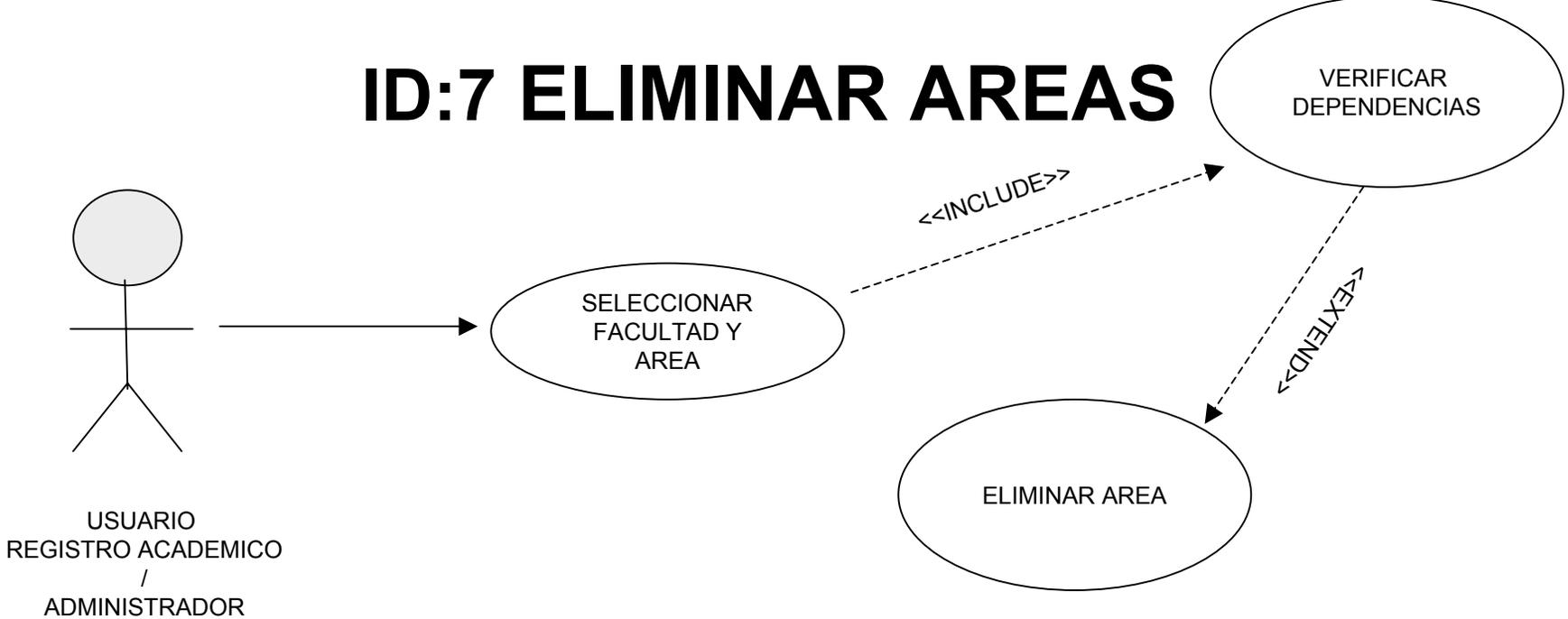
# ID:6 CREAR AREAS



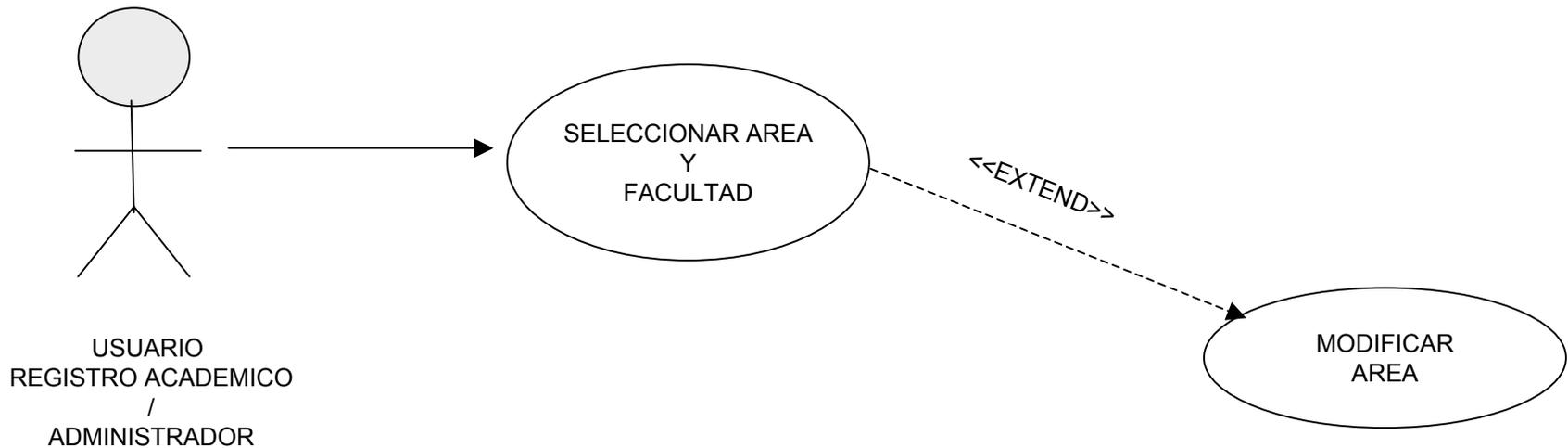
# Tarjeta Descriptiva Caso de Uso 6

<b>Id caso de uso</b>	<b>6</b>
<b>Nombre caso de uso</b>	Crear Áreas
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Registro Académico
<b>Descripción</b>	Crea las áreas para cada facultad
<b>Precondición</b>	Deben estar creadas las facultades
<b>Postcondiciones</b>	Creación de áreas
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. Seleccionar facultad</li><li>2. Ingresar datos de área</li></ol>
<b>Flujos Alternativo</b>	Puede hacerse de forma inversa
<b>Excepciones</b>	Se deben ingresar todos los campos si no, no se podrá crear el área
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Baja
<b>Requerimientos especiales</b>	

# ID:7 ELIMINAR AREAS



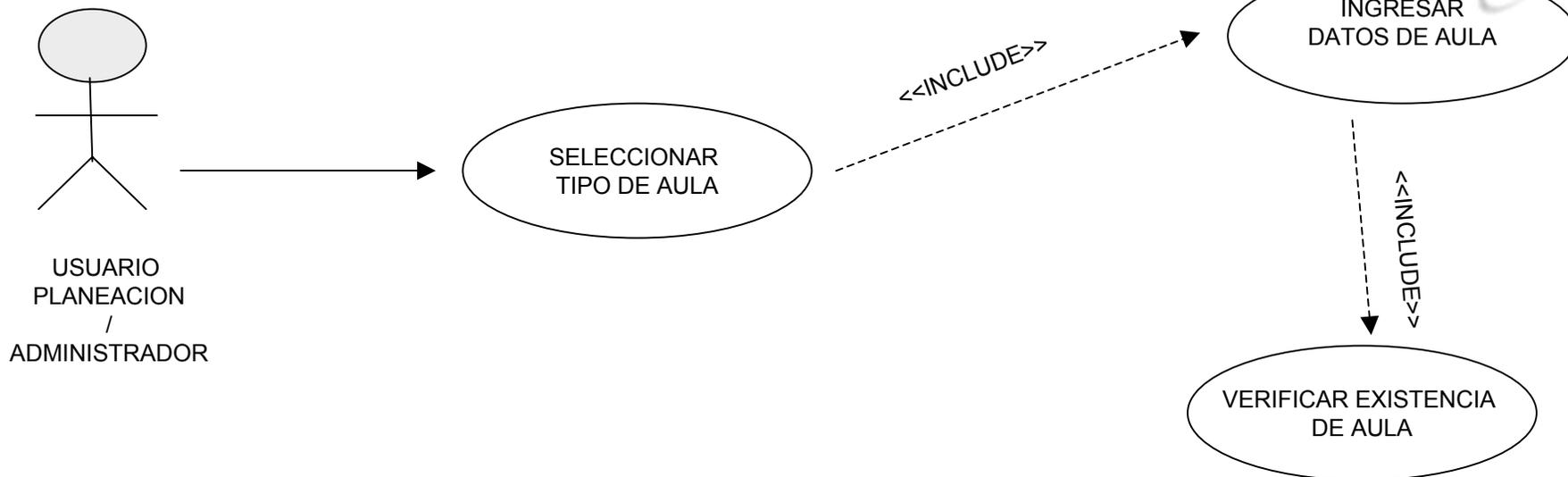
# MODIFICAR AREAS



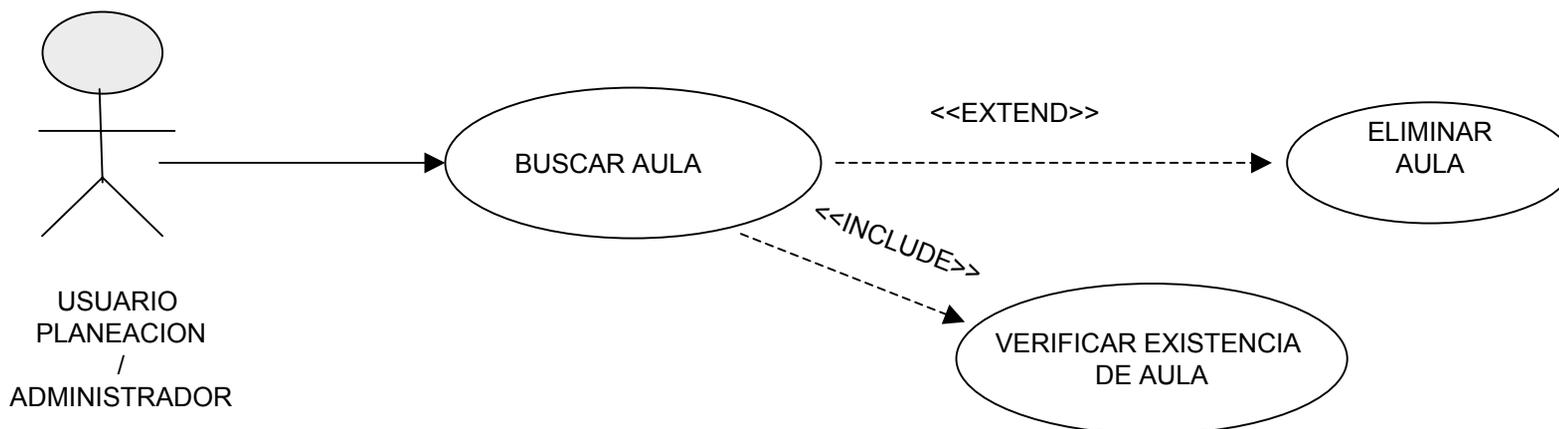
# Tarjeta Descriptiva Caso de Uso 7

<b>Id caso de uso</b>	7
<b>Nombre caso de uso</b>	Eliminar Áreas
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Registro Académico
<b>Descripción</b>	Elimina las áreas para cada facultad
<b>Precondición</b>	
<b>Postcondiciones</b>	No deben existir relaciones con otros componentes del sistema
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. Seleccionar facultad</li><li>2. Seleccionar área</li></ol>
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	Error de eliminación cuando encuentra relaciones
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Baja
<b>Requerimientos especiales</b>	

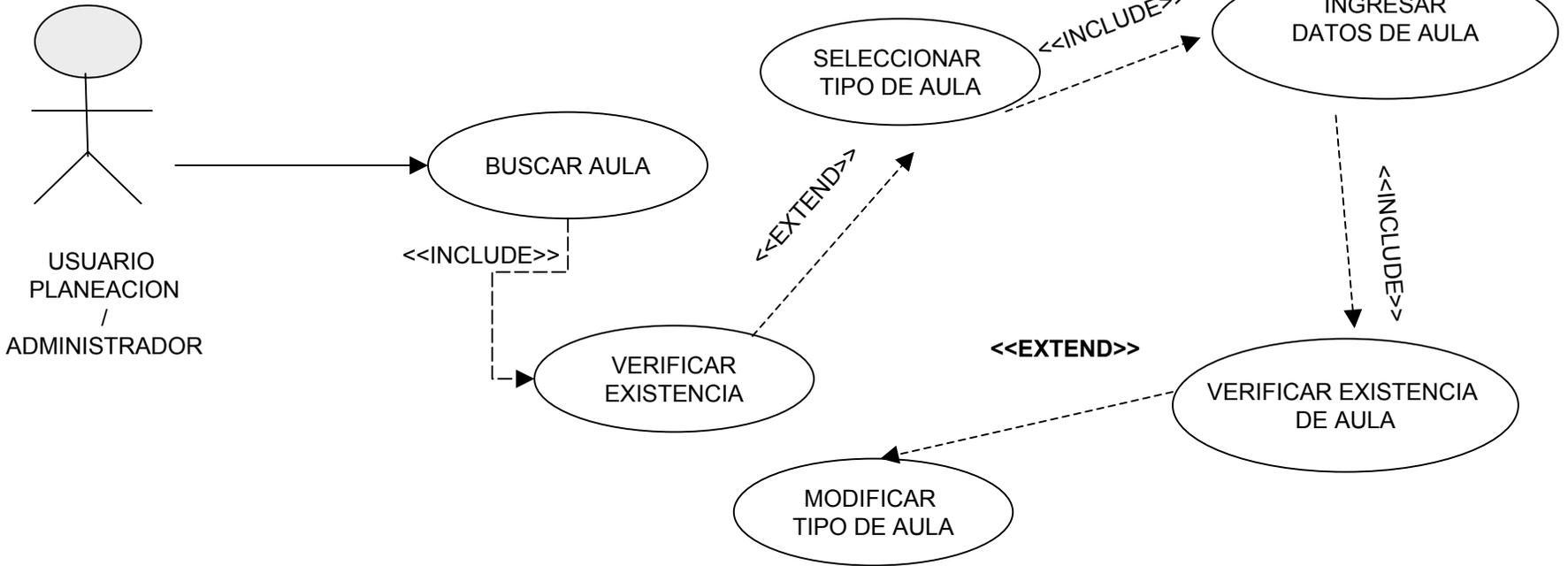
# ID:8 CREAR AULAS



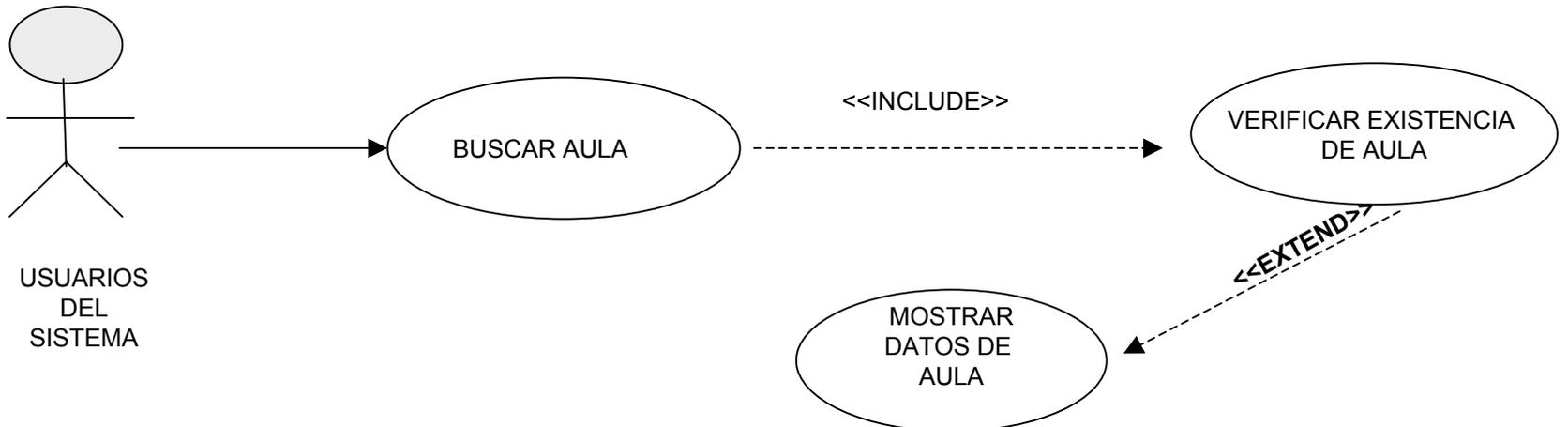
# ELIMINAR AULAS



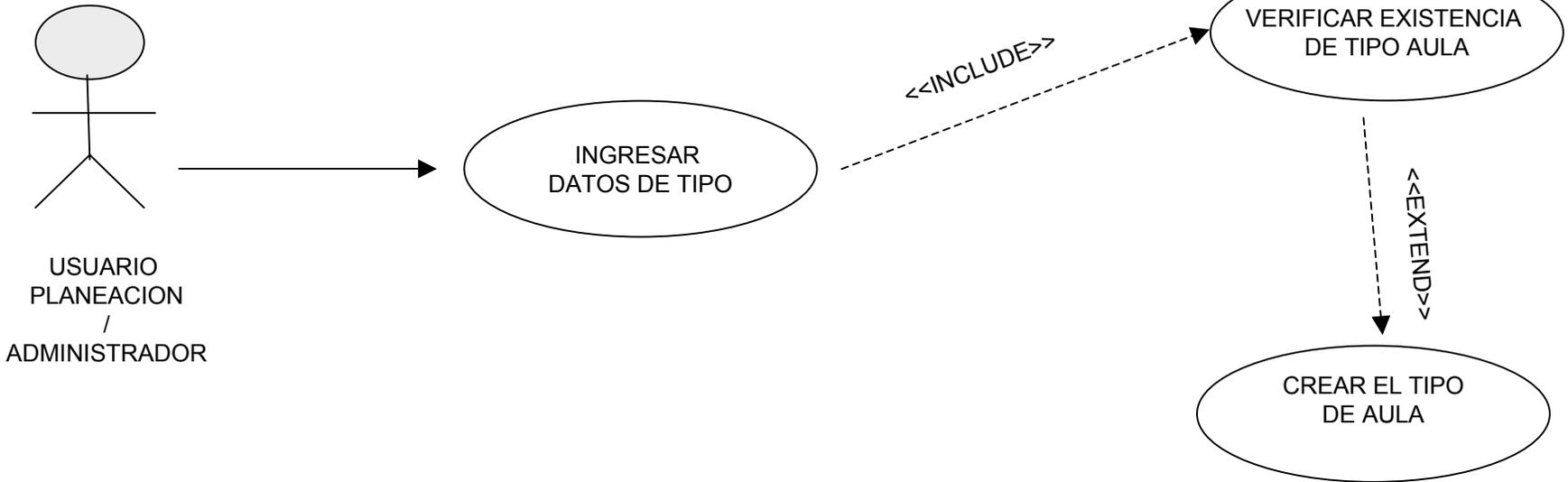
# ID:9 MODIFICAR AULAS



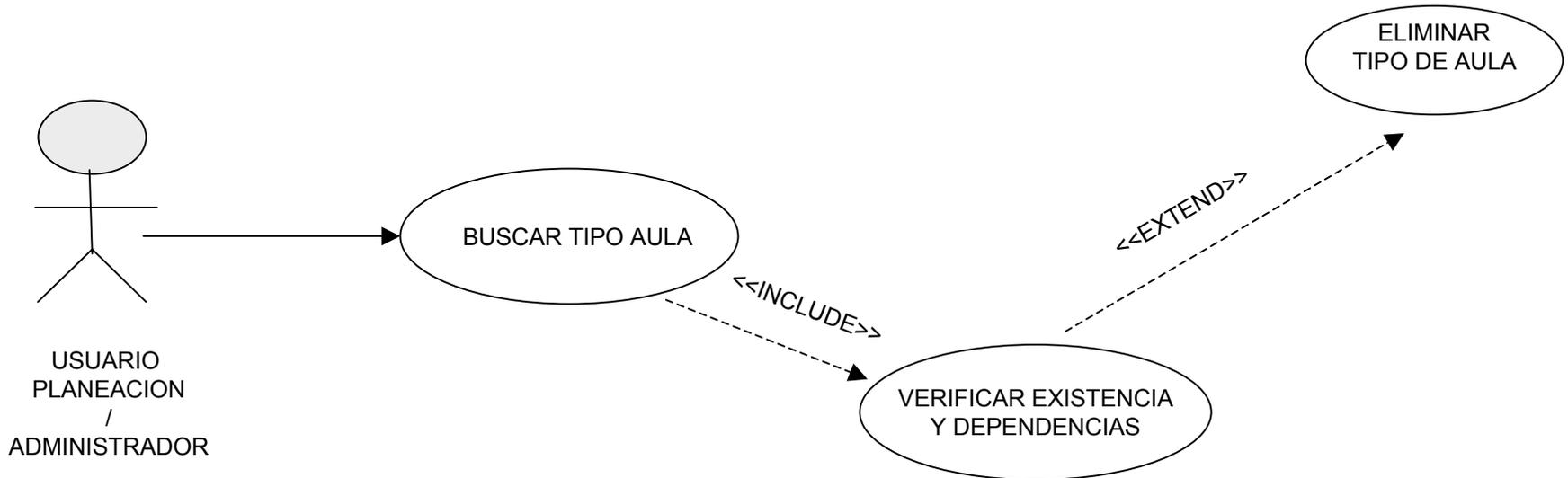
# CONSULTAR AULAS



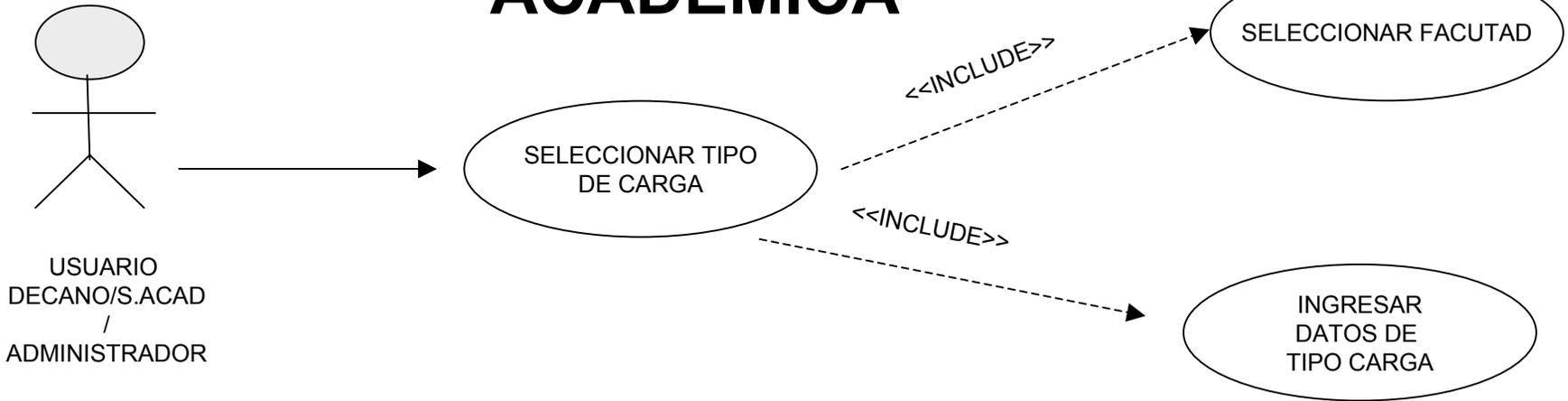
# ID:10 CREAR TIPO AULAS



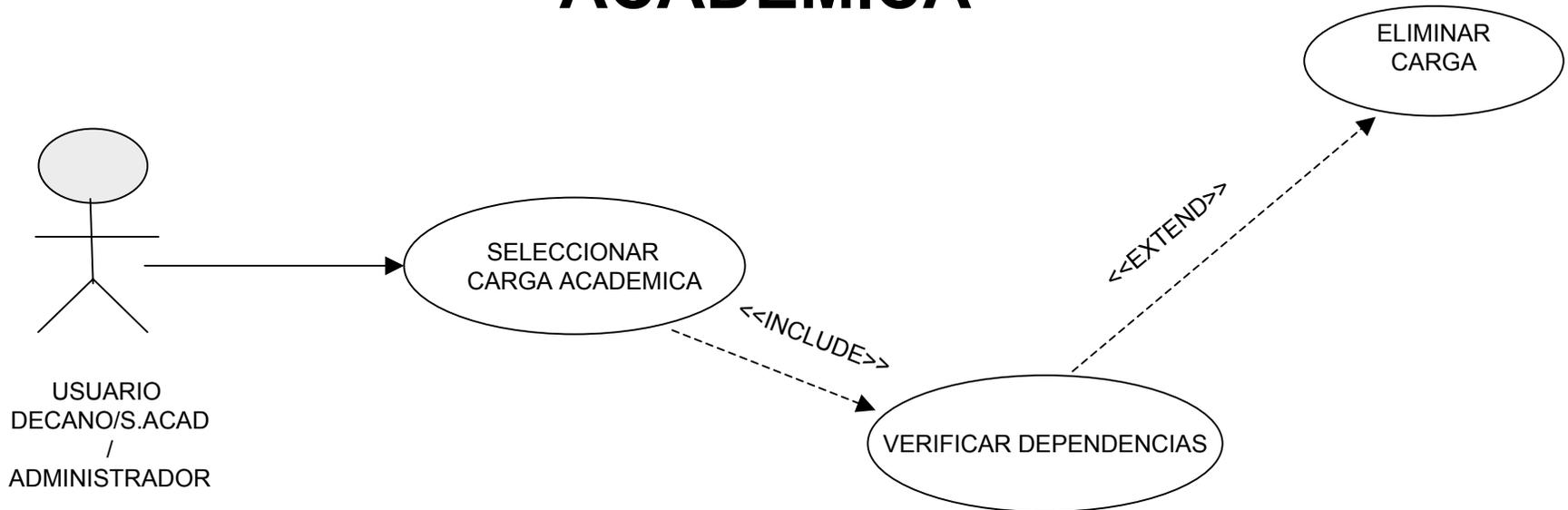
# ELIMINAR TIPO AULAS



# ID:11 CREAR TIPOS DE CARGA ACADEMICA



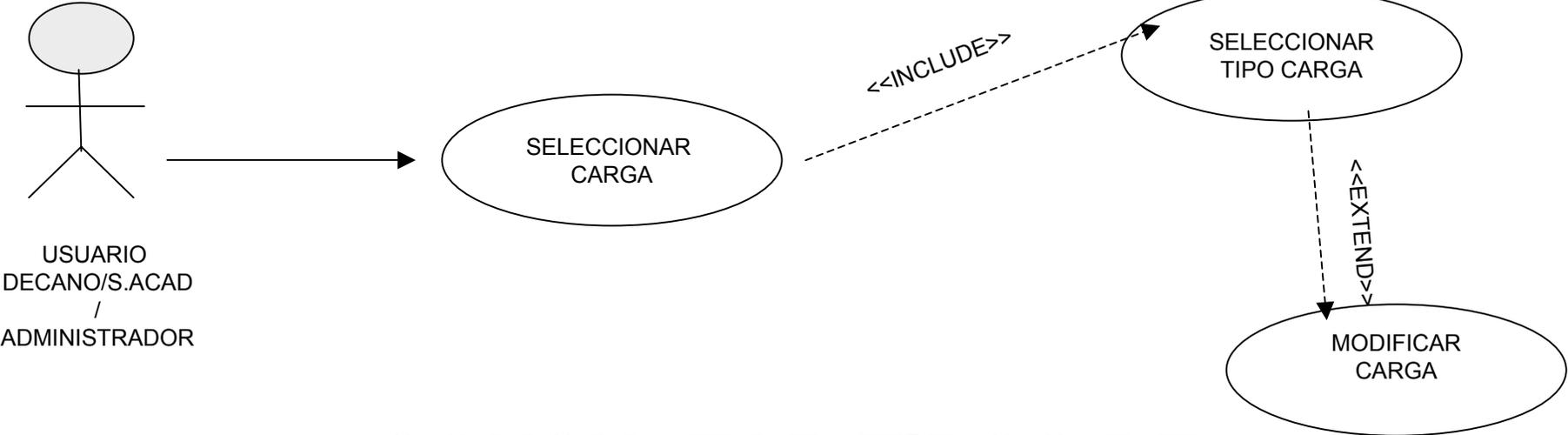
# ELIMINAR TIPOS DE CARGA ACADEMICA



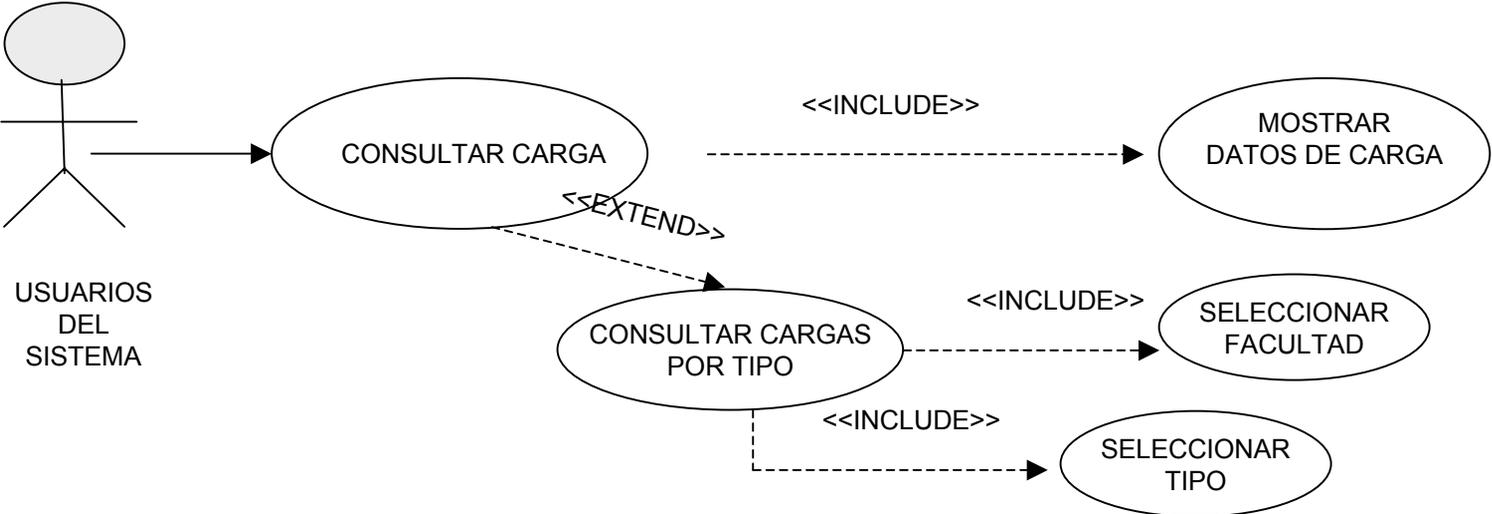
# Tarjeta Descriptiva Caso de Uso 11

<b>Id caso de uso</b>	<b>11</b>
<b>Nombre caso de uso</b>	Crear Tipos de Carga Académica
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador. Decano Sec Académico
<b>Descripción</b>	Se crea una carga adicional diferente a la cátedra clasificándola en alguno de los tipos manejados por la universidad(Investigación, Extensión, Producción escrita. etc
<b>Precondición</b>	
<b>Postcondiciones</b>	
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. Seleccionar el Tipo de carga</li><li>2. Seleccionar facultad en caso de ser administrador</li><li>3. Insertar datos de la carga</li></ol>
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	En caso de faltar algún campo por llenar se activa el sistema de validación que no permite la creación de la carga hasta que no se llenen correctamente todos los campos
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	

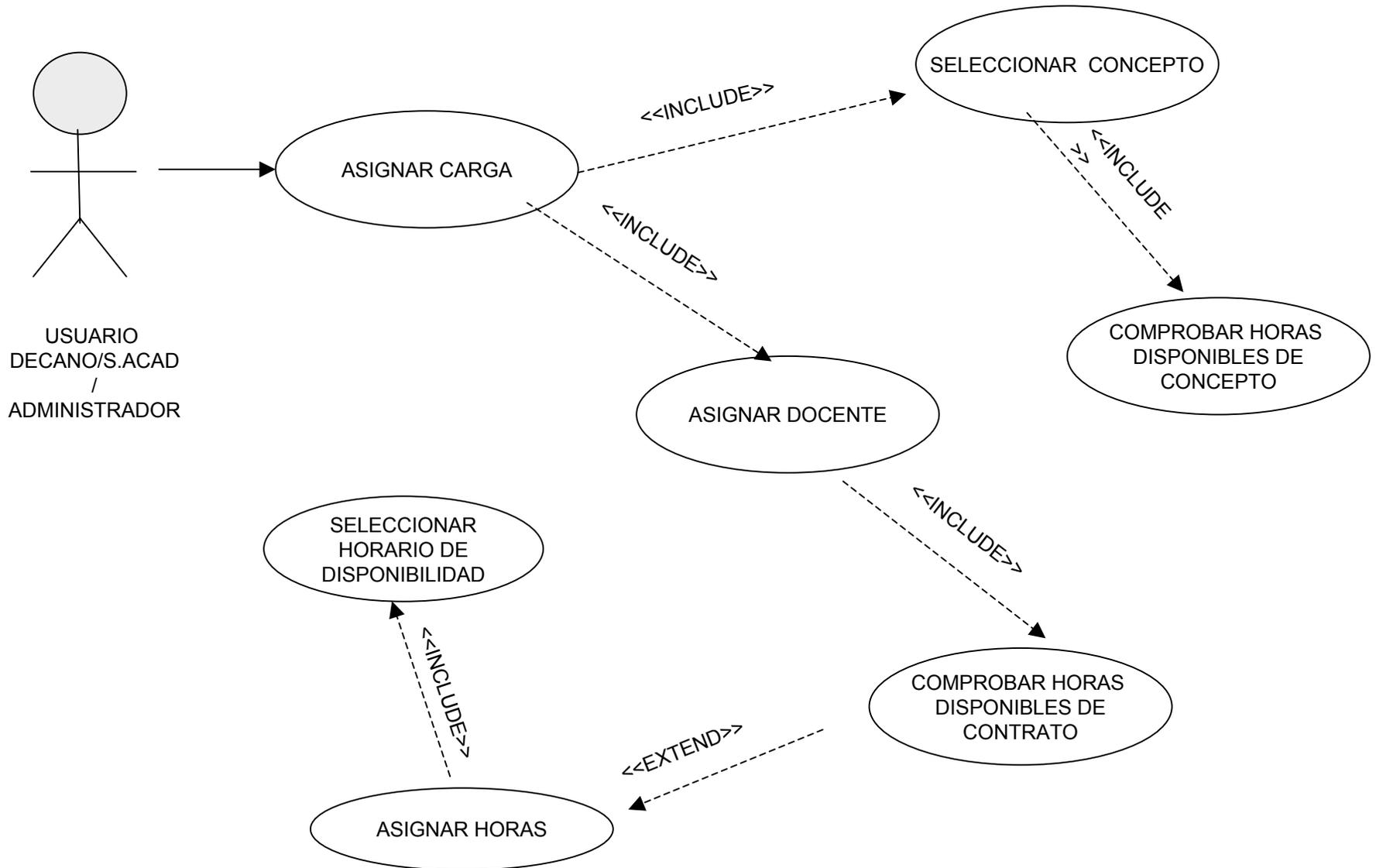
# ID:12 MODIFICAR TIPOS DE CARGA



# CONSULTAR TIPOS DE CARGAS



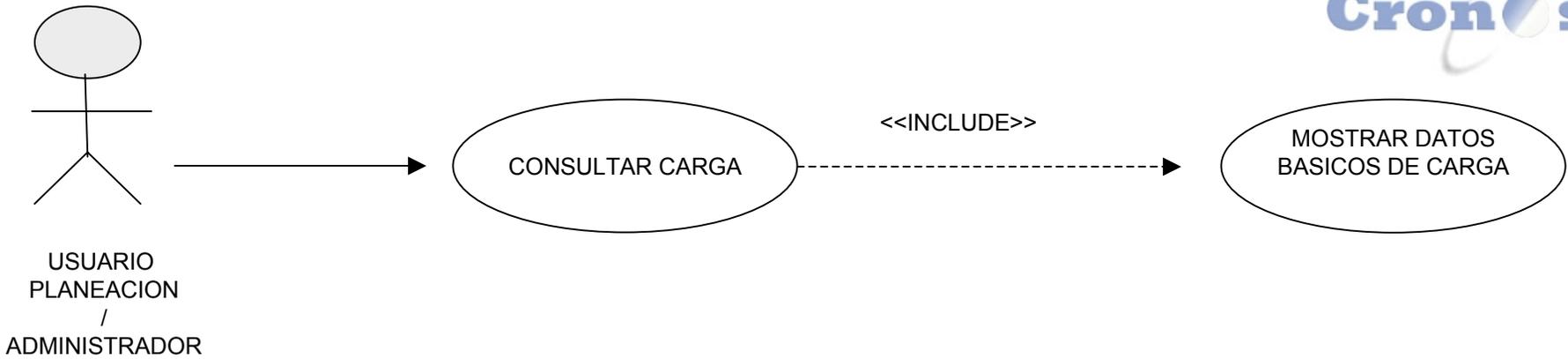
# ID:13 ASIGNAR CARGA ADICIONAL A ACADEMICOS



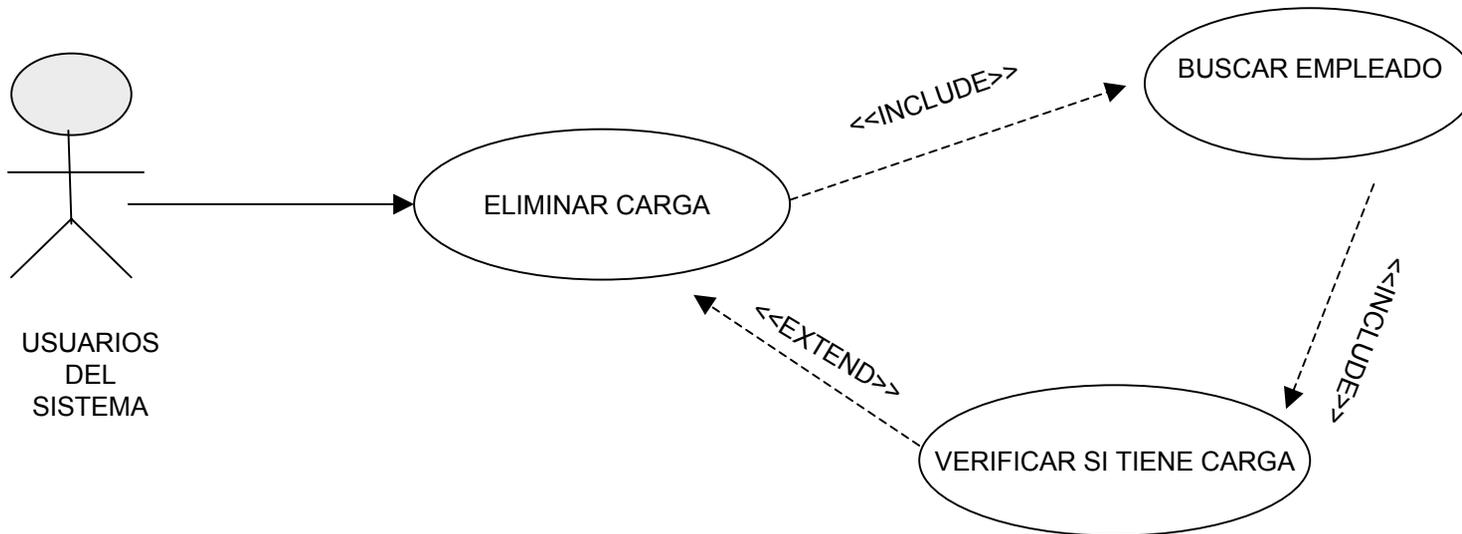
# Tarjeta Descriptiva Caso de Uso 13

<b>Id caso de uso</b>	<b>13</b>
<b>Nombre caso de uso</b>	Adicionar Carga Adicional a académicos
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decano, Sec Académico
<b>Descripción</b>	Asigna la carga adicional a los académicos.
<b>Precondición</b>	Existencia de cargas adicionales, horas disponibles por carga, y vacantes en la cantidad de empleados. El empleado tener horas disponibles
<b>Postcondiciones</b>	Asignar la carga adicional
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. Seleccionar la carga</li><li>2. Seleccionar empleado</li><li>3. Asignar Horas del proyecto al empleado.</li><li>4. Asignar Horario semanal para la carga</li></ol>
<b>Flujos Alternativo</b>	Cualquier paso siempre y cuando cumpla con las precondiciones.
<b>Excepciones</b>	No se podrá acceder a los diferentes campos si no selecciona o llena los campos requeridos.
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	

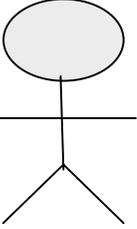
# ID:14 CONSULTAR CARGA EN ACADEMICOS



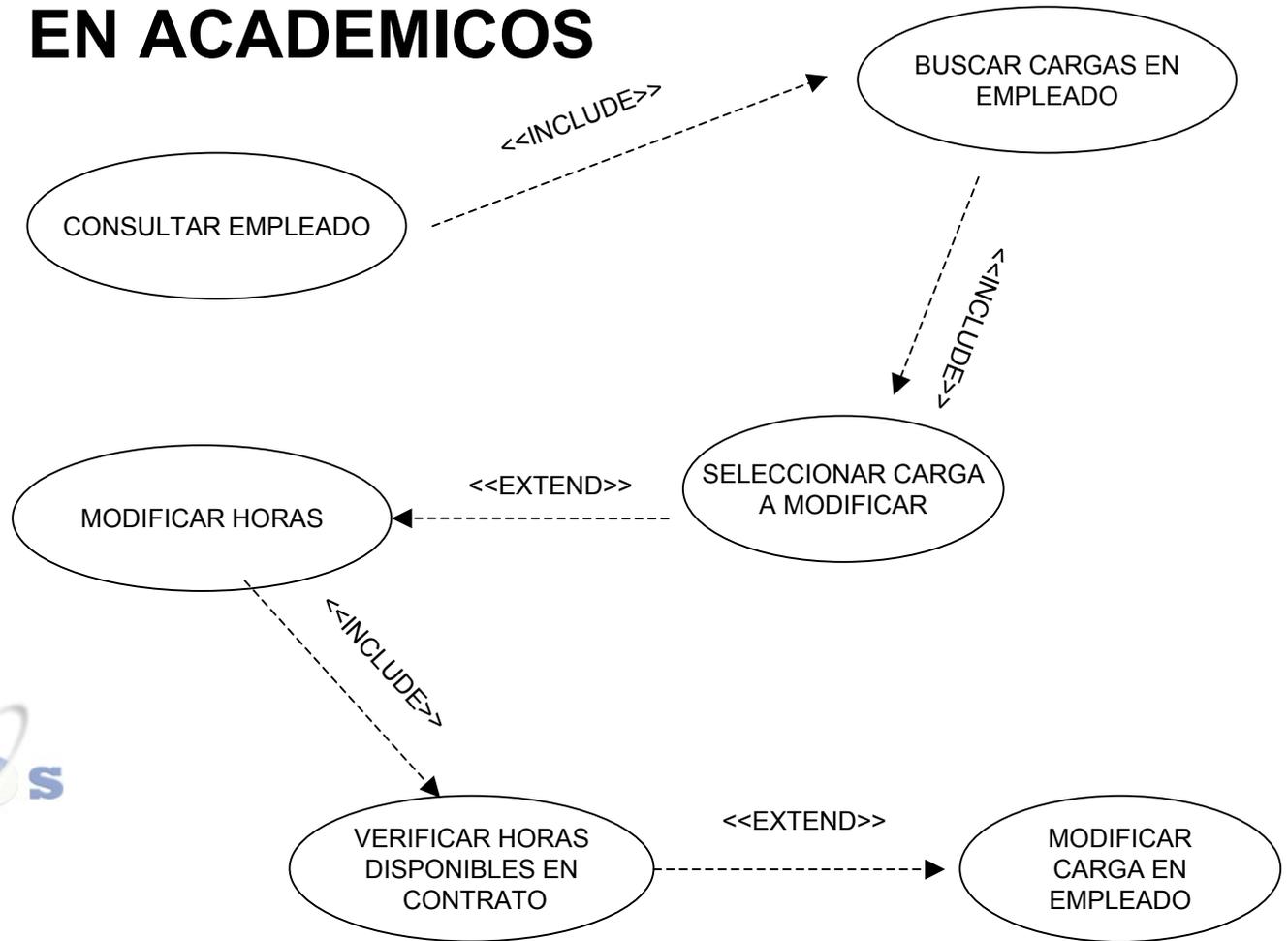
# ID:15 ELIMINAR CARGA ADICIONAL EN ACADEMICOS



# ID:16 MODIFICAR CARGA ADICIONAL EN ACADEMICOS



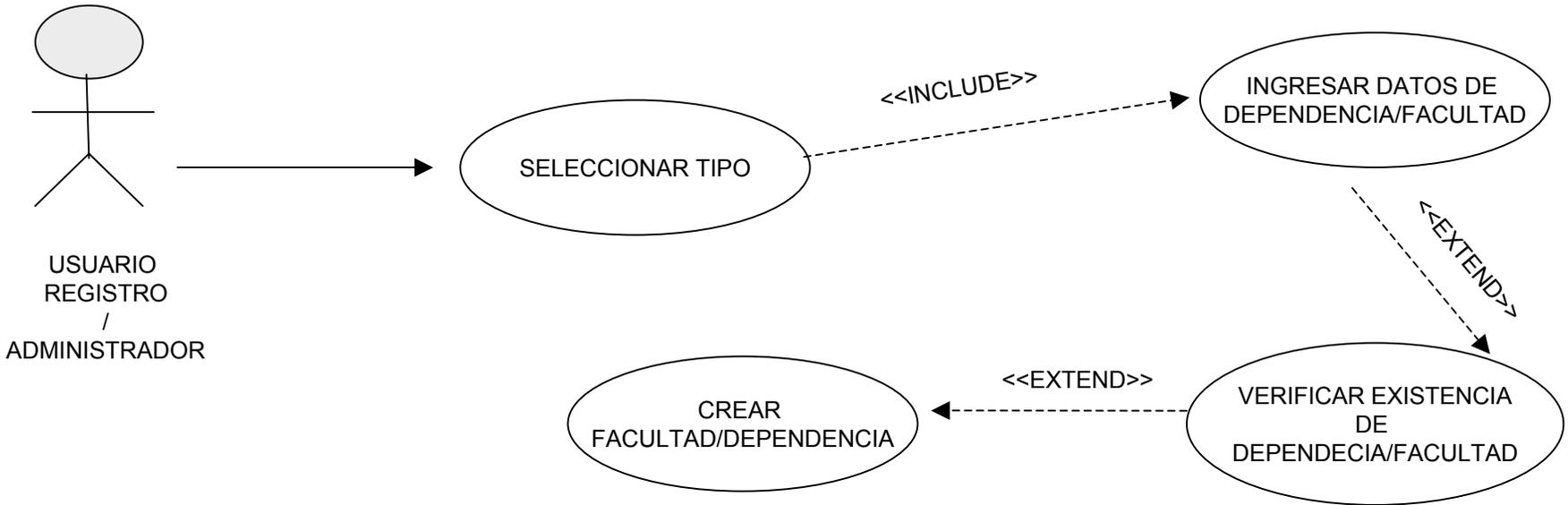
USUARIO  
DECANO/S.ACAD  
/  
ADMINISTRADOR



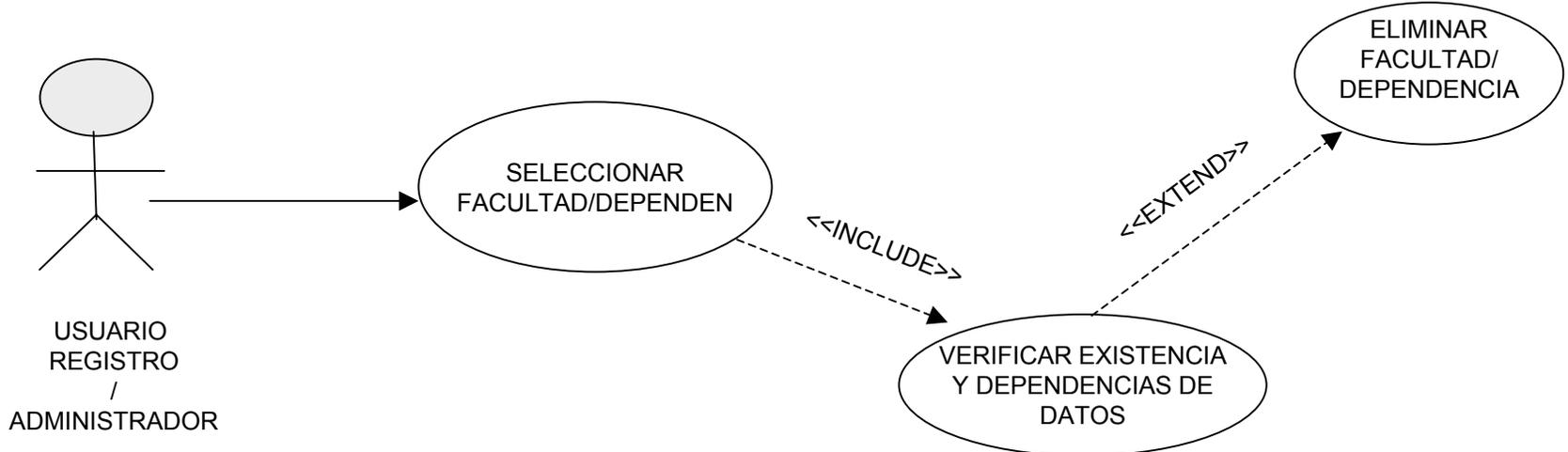
# Tarjeta Descriptiva Caso de Uso 16

<b>Id caso de uso</b>	<b>16</b>
<b>Nombre caso de uso</b>	Modificar carga adicional en académicos
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decanos, Sec Académico
<b>Descripción</b>	Modifica datos de cargas adicionales en los empleados
<b>Precondición</b>	Tener Cargas
<b>Postcondiciones</b>	Modificar las cargas que solicite el usuario
<b>Flujo Normal</b>	Seleccionar el empleado 1. Seleccionar la carga a modificar 2. Modificar las Horas
<b>Flujos Alternativo</b>	Solo el flujo normal por que se alimentan de forma secuencial los campos
<b>Excepciones</b>	En el paso 3 del flujo normal deben existir horas disponibles en la carga adicional en caso que desee ampliar, además debe existir disponibilidad en el empleado para asumir la carga
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	

# ID:17 CREAR FACULTADES Y DEPENDENCIAS



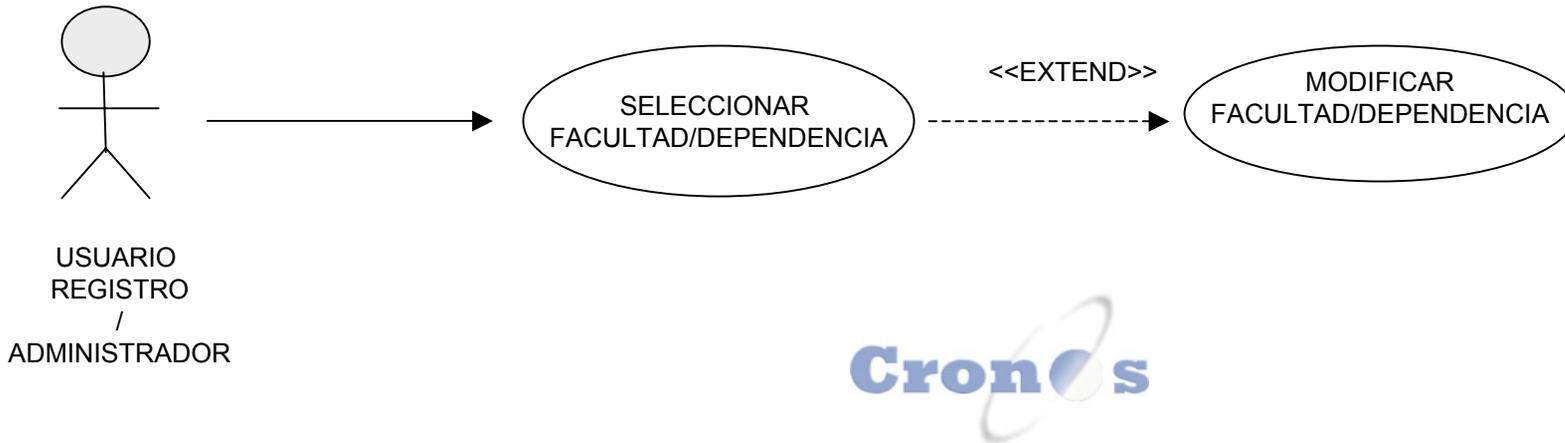
# ID:18 ELIMINAR FACULTADES Y DEPENDENCIAS



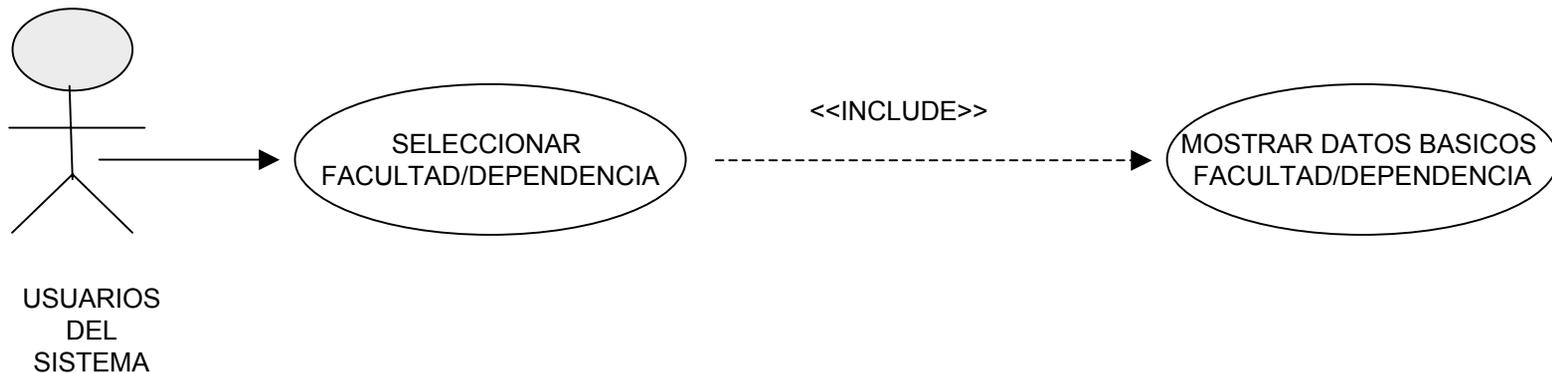
# Tarjeta Descriptiva Caso de Uso 17

<b>Id caso de uso</b>	<b>17</b>
<b>Nombre caso de uso</b>	Crear Facultades y Dependencias
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Registro Académico
<b>Descripción</b>	Permite crear facultades, dependencias de la Universidad. Para que posteriormente se puedan asignar programas, areas, materias y empleados.
<b>Precondición</b>	Ninguna
<b>Postcondiciones</b>	Crear las facultades y/o dependencias
<b>Flujo Normal</b>	1. Seleccionar tipo (facultad o dependencia) 2. Ingresar datos básicas
<b>Flujos Alternativo</b>	3. Crear la facultad El paso uno se puede alternar sin problema
<b>Excepciones</b>	Si el código que se ingresa existe ya en el sistema, la facultad no se dejara crear hasta no ser cambiado
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Baja
<b>Requerimientos especiales</b>	

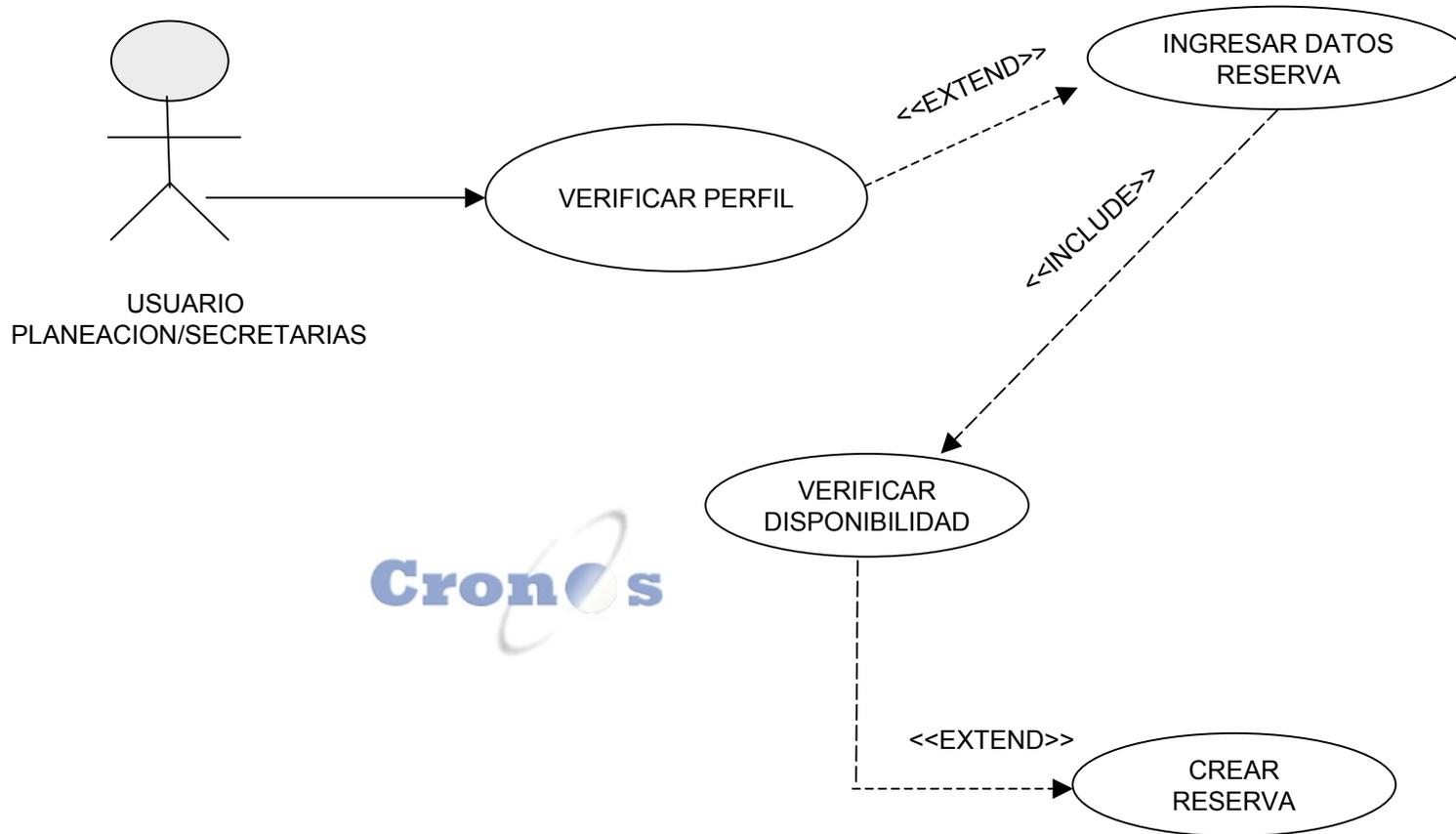
# ID:19 MODIFICAR FACULTADES/DEPENDENCIAS



# ID:20 CONSULTAR FACULTADES/DEPENDENCIAS



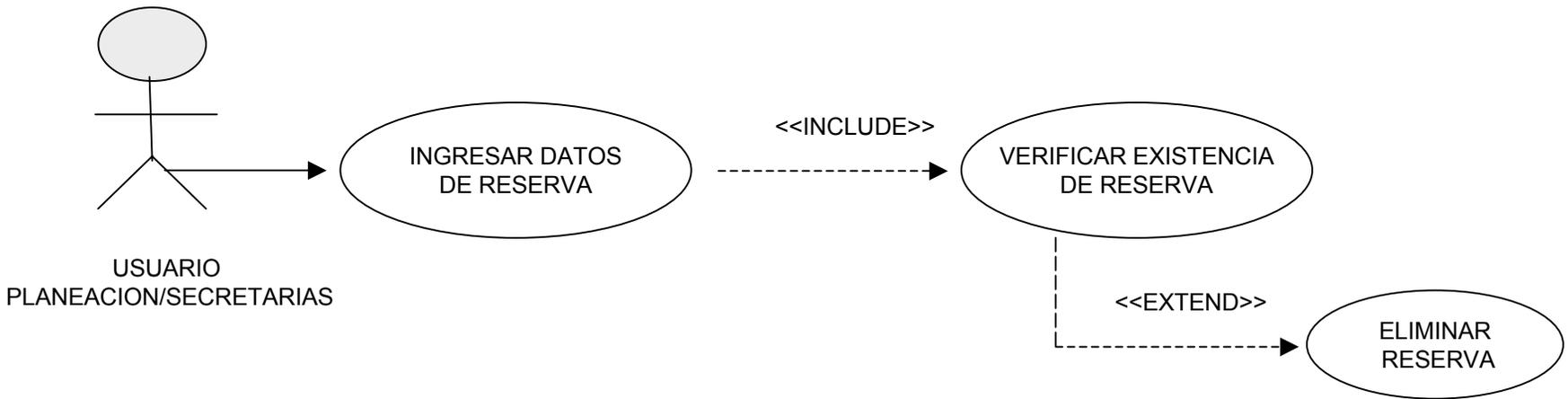
# ID:21 RESERVAR AULAS



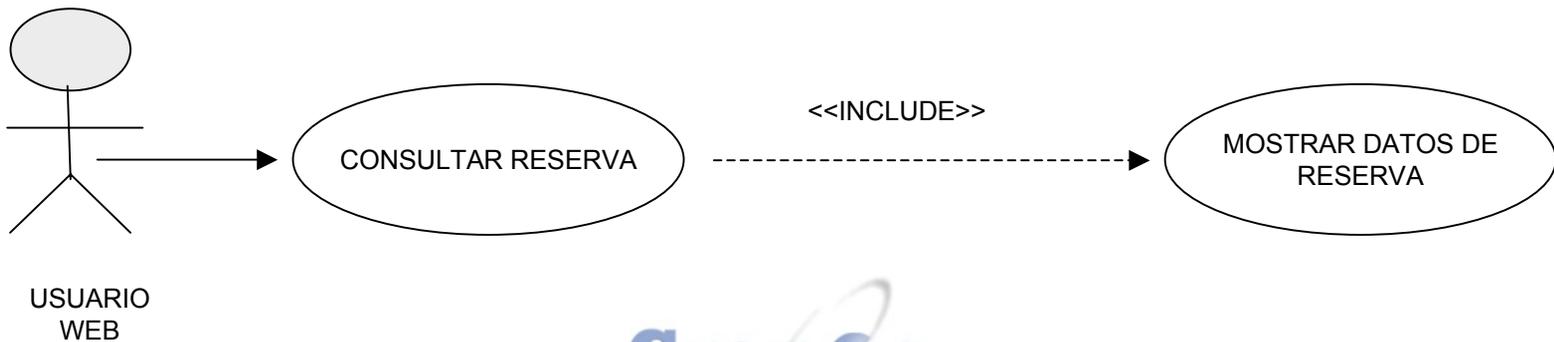
# Tarjeta Descriptiva Caso de Uso 21

<b>Id caso de uso</b>	<b>21</b>
<b>Nombre caso de uso</b>	Crear Reserva de aulas
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Planeacion, Secretarias
<b>Descripción</b>	Permite la reserva de aulas
<b>Precondición</b>	Disponibilidad de las aulas
<b>Postcondiciones</b>	Crear la reserva
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1.Ingresar datos para la reserva(fecha, horas...)</li><li>2.Seleccionar aula de las disponibles</li><li>3.Crear reserva</li></ol>
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	Si no hay aulas disponibles no se puede reservar.
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Alta
<b>Requerimientos especiales</b>	

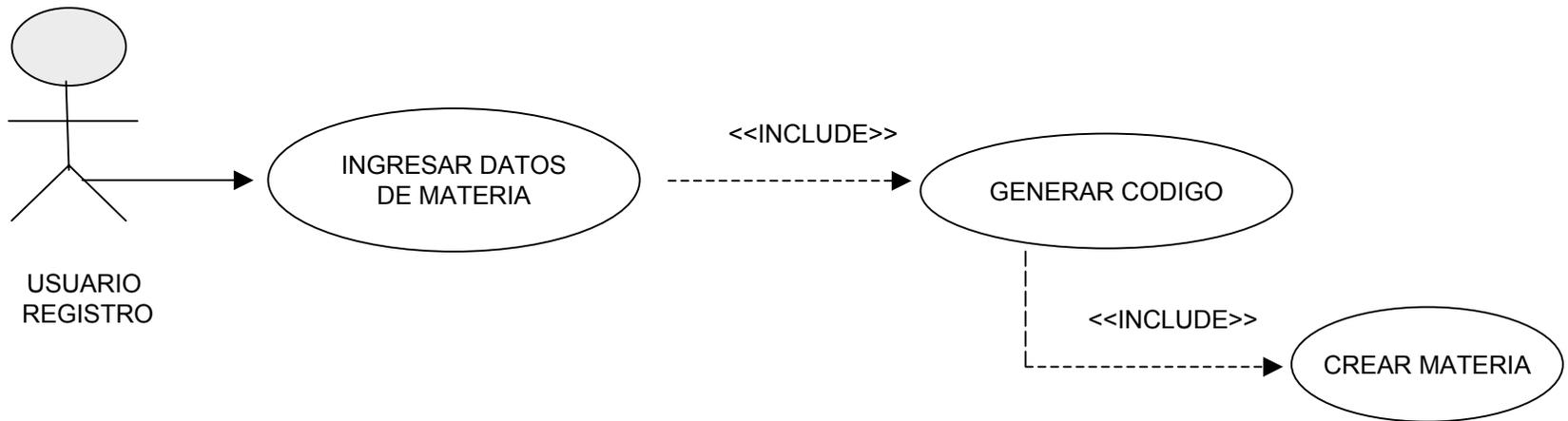
# ID: 22 ELIMINAR RESERVAS



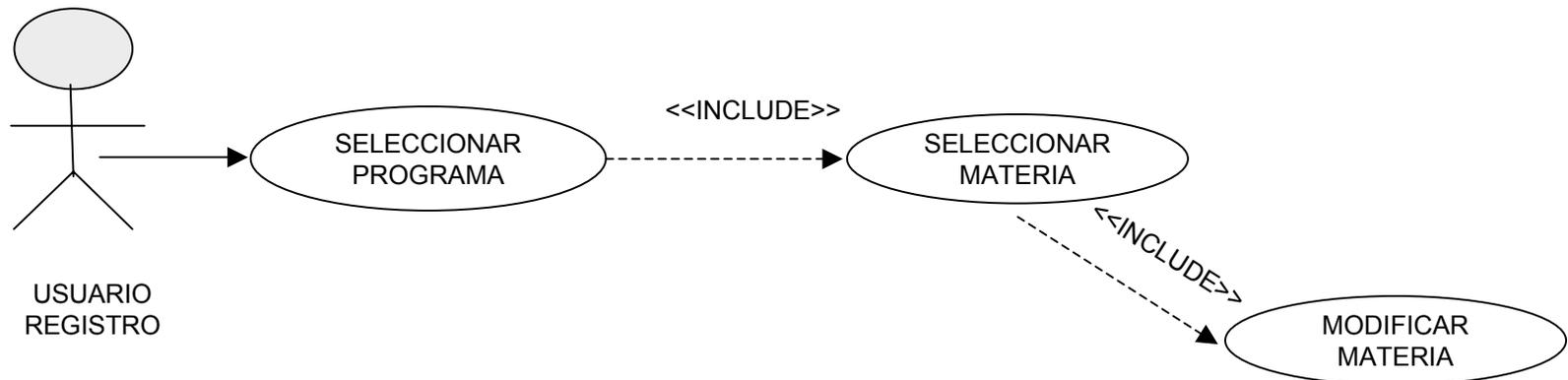
# ID: 23 CONSULTAR RESERVAS



# ID: 24 CREAR MATERIAS



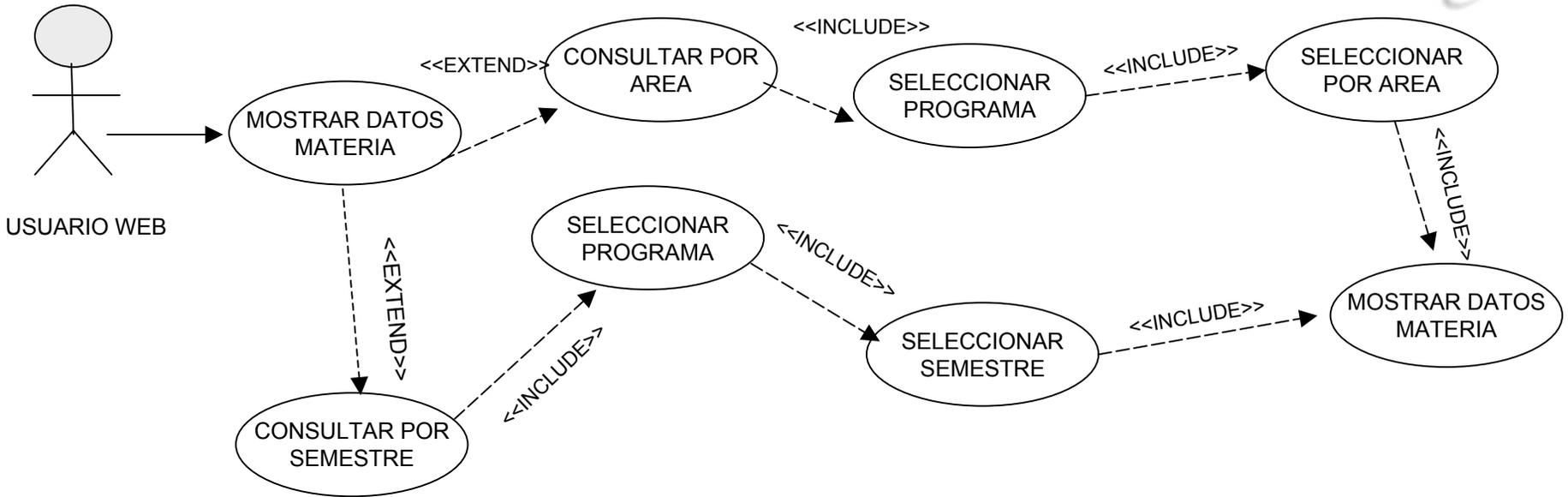
# ID: 25 MODIFICAR MATERIAS



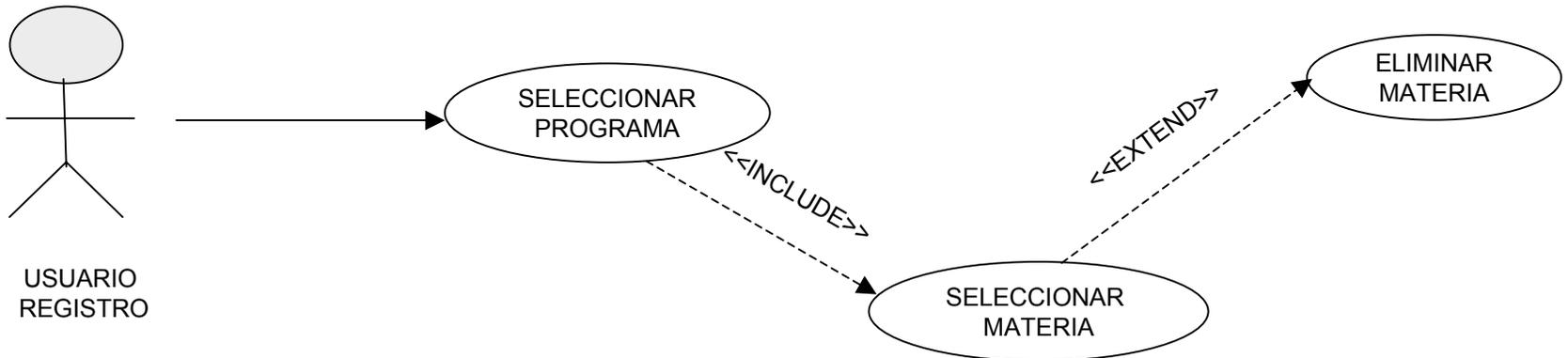
# Tarjeta Descriptiva Caso de Uso 24

<b>Id caso de uso</b>	<b>24</b>
<b>Nombre caso de uso</b>	Crear Materias
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Registro
<b>Descripción</b>	Permite crear materias para las facultades
<b>Precondición</b>	Existencia de facultades y áreas
<b>Postcondiciones</b>	Crear la materia a los programas
<b>Flujo Normal</b>	1.Ingresar datos de la materia 2.Sistema genera código a partir del llenado de los campos. 3. Crear materia
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	Si no selecciona los datos obligatorios no se podrá crear la materia
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	

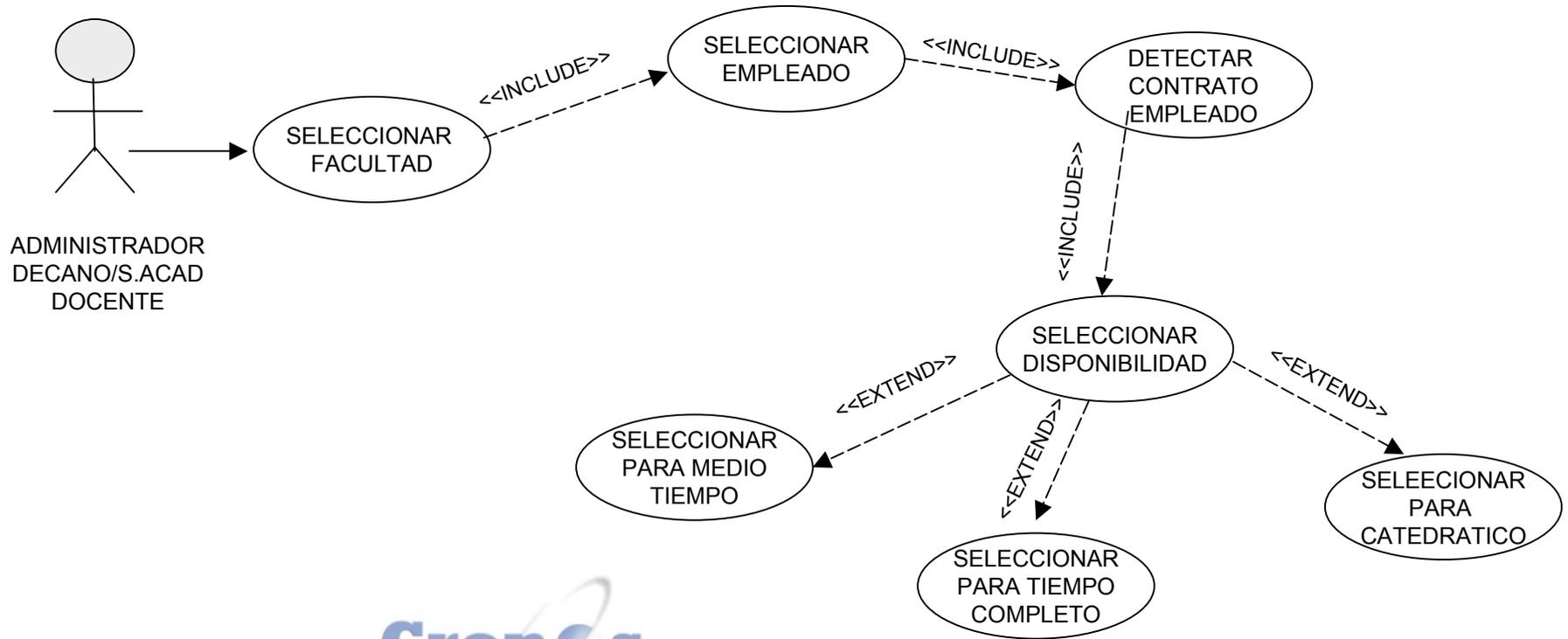
# ID: 26 CONSULTAR MATERIA



# ID: 27 ELIMINAR MATERIA



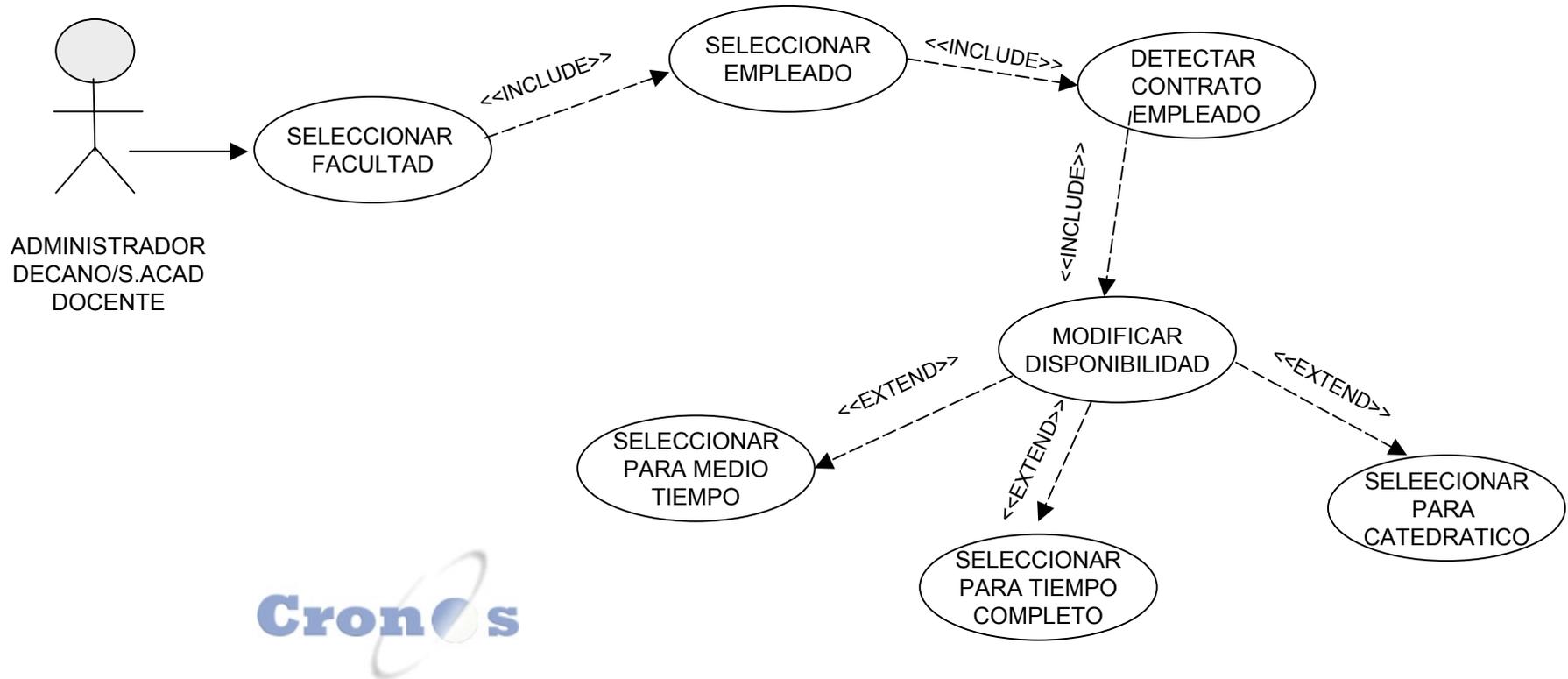
# ID: 28 CREAR DISPONIBILIDAD



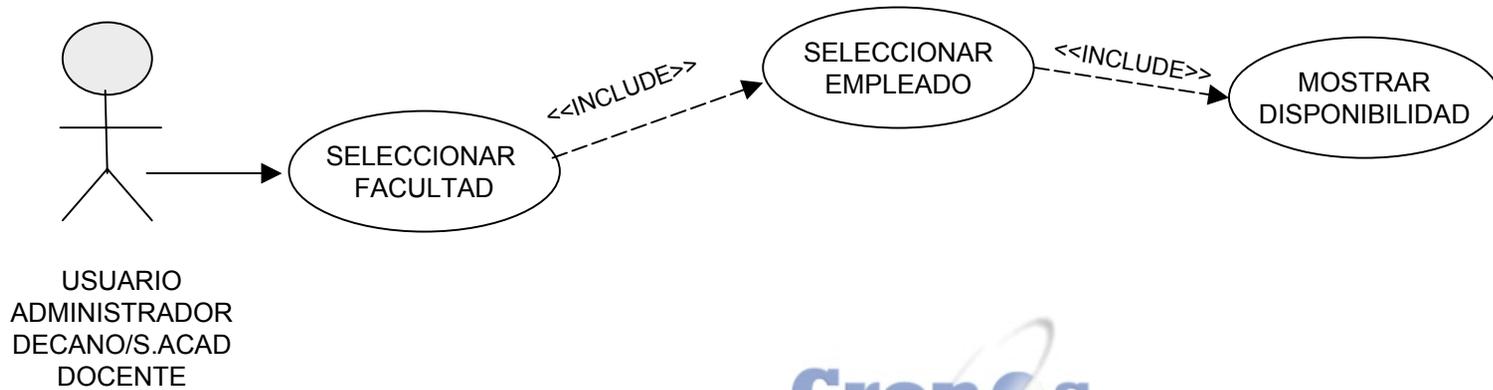
# Tarjeta Descriptiva Caso de Uso 28

<b>Id caso de uso</b>	<b>28</b>
<b>Nombre caso de uso</b>	Crear Disponibilidad
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decanos, Sec académico, Docente
<b>Descripción</b>	Permite asignar la disponibilidad de los docentes a partir de su tipo de contratación laboral
<b>Precondición</b>	Existencia del contrato en el empleado
<b>Postcondiciones</b>	Asignar la disponibilidad
<b>Flujo Normal</b>	1. Seleccionar programa 2. Seleccionar tipo de contrato. 3. Seleccionar el empleado 4. Asignar horario de disponibilidad
<b>Flujos Alternativo</b>	
<b>Excepciones</b>	Si no selecciona la disponibilidad no se creara ninguna y no podrá acceder a ningún tipo de carga.
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Normal
<b>Requerimientos especiales</b>	

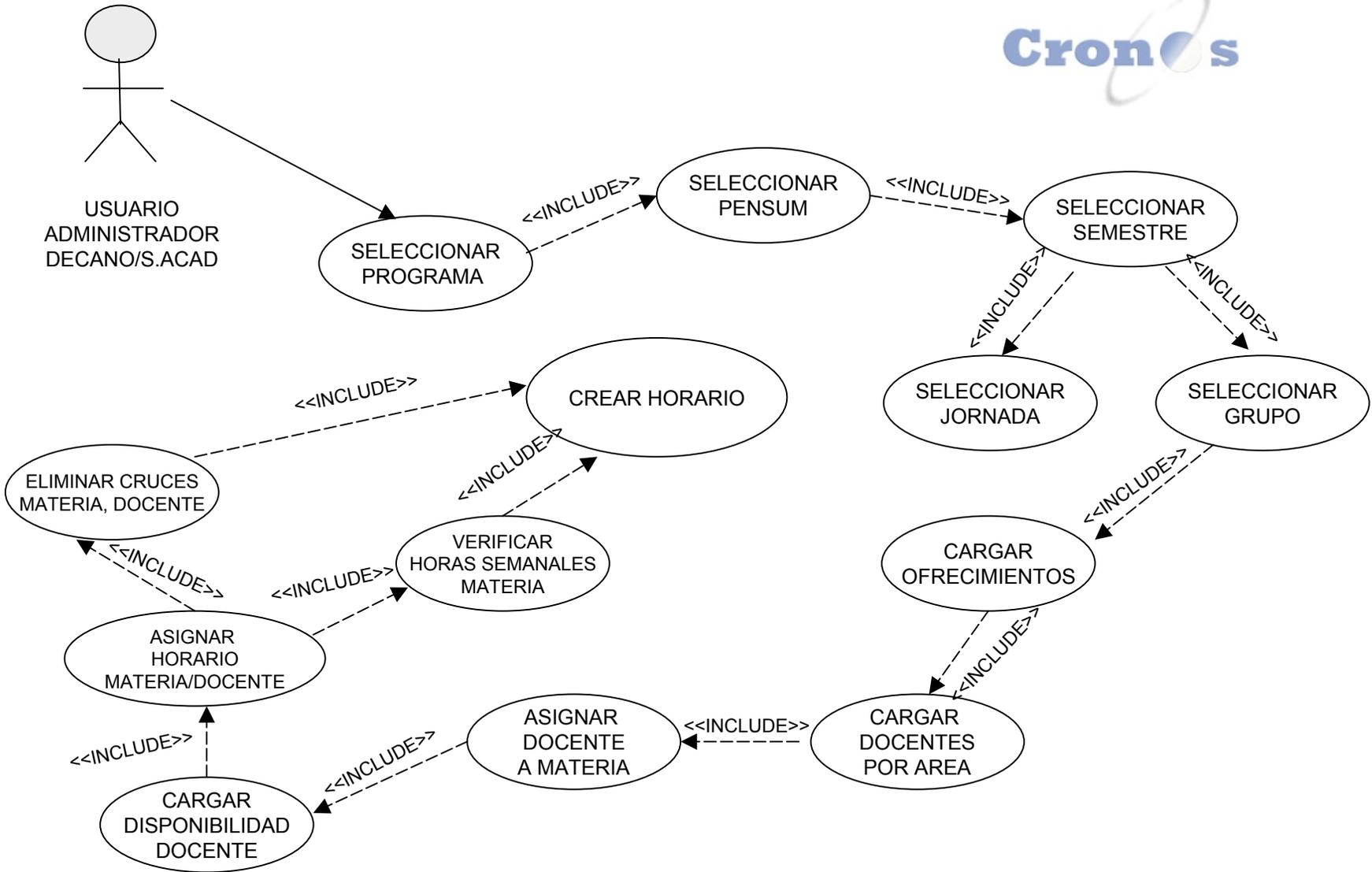
# ID: 29 MODIFICAR DISPONIBILIDAD



# ID: 30 CONSULTAR DISPONIBILIDAD



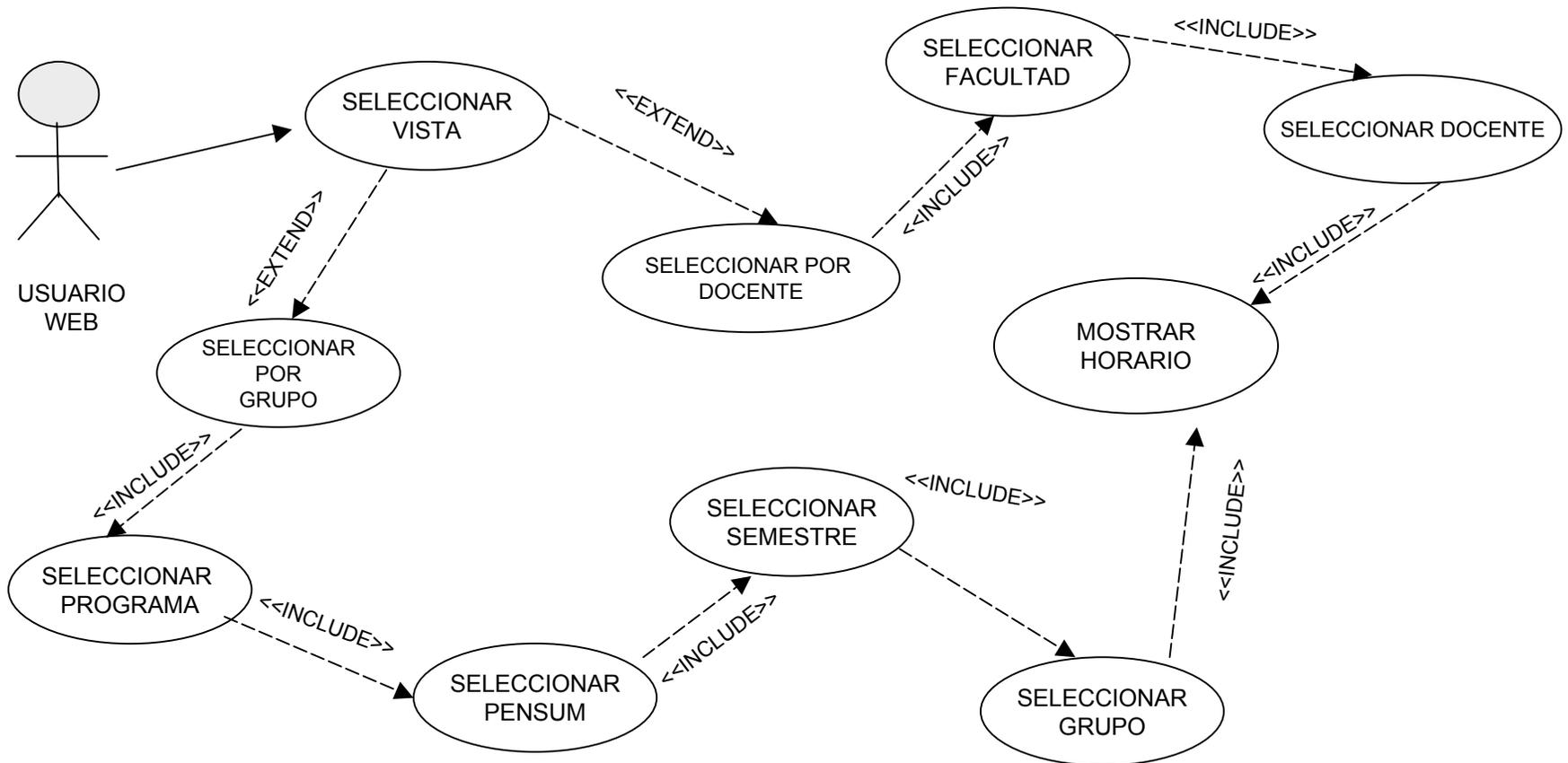
# ID: 31 CREAR HORARIO



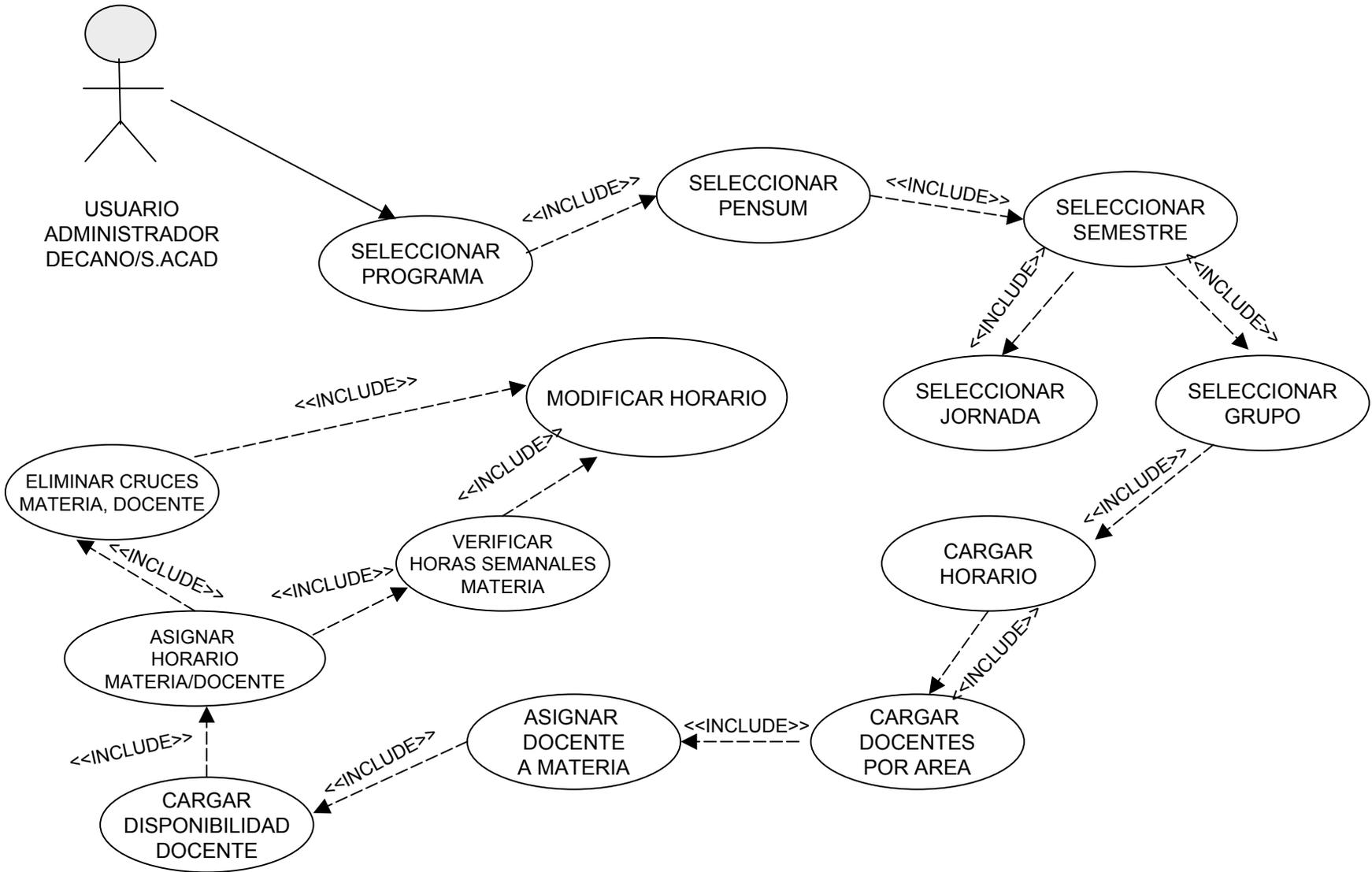
# Tarjeta Descriptiva Caso de Uso 31

<b>Id caso de uso</b>	<b>31</b>
<b>Nombre caso de uso</b>	Crear Horario
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decanos, Sec académico
<b>Descripción</b>	Permite crear los horarios de clase(ofrecimientos)
<b>Precondición</b>	Existencia de ofrecimientos, disponibilidad en docentes y docentes agrupados por áreas, materias con sus horas semanales e inscritas en el pensum que se trabajara.
<b>Postcondiciones</b>	Crear el horario académico
<b>Flujo Normal</b>	<ol style="list-style-type: none"> <li>1. Seleccionar facultad/dependencia</li> <li>2. Seleccionar Pensum.</li> <li>3. Seleccionar el Semestre</li> <li>4. Seleccionar Jornada y grupo</li> <li>5. El sistema carga: ofrecimientos, materias por area, docentes por área, disponibilidad de docentes, horas semanales materia</li> <li>6. Seleccionar ofrecimiento.</li> <li>7. Seleccionar Docente.</li> <li>8: Asignar horario.</li> </ol>
<b>Flujos Alternativo</b>	9. Sistema actualiza disponibilidad del docente, actualiza horas materia, elimina campos de cruce y políticas de horarios.
<b>Excepciones</b>	<ol style="list-style-type: none"> <li>10. Crear Horario</li> </ol> <p>Si no existen los ofrecimientos y docentes no se podrá crear ningún horario.</p>
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Baja
<b>Requerimientos especiales</b>	

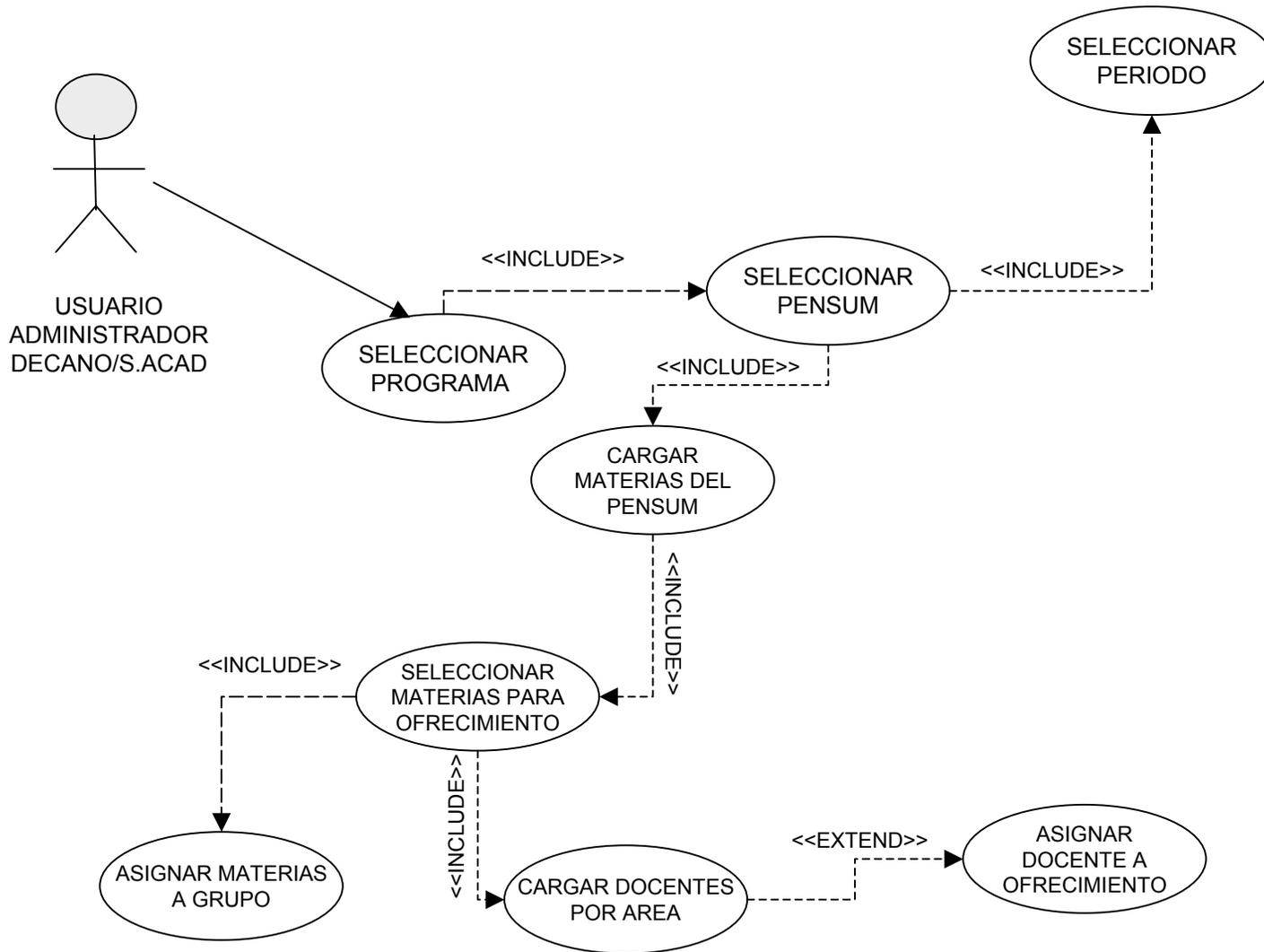
# ID: 32 CONSULTAR HORARIO



# ID: 33 MODIFICAR HORARIO



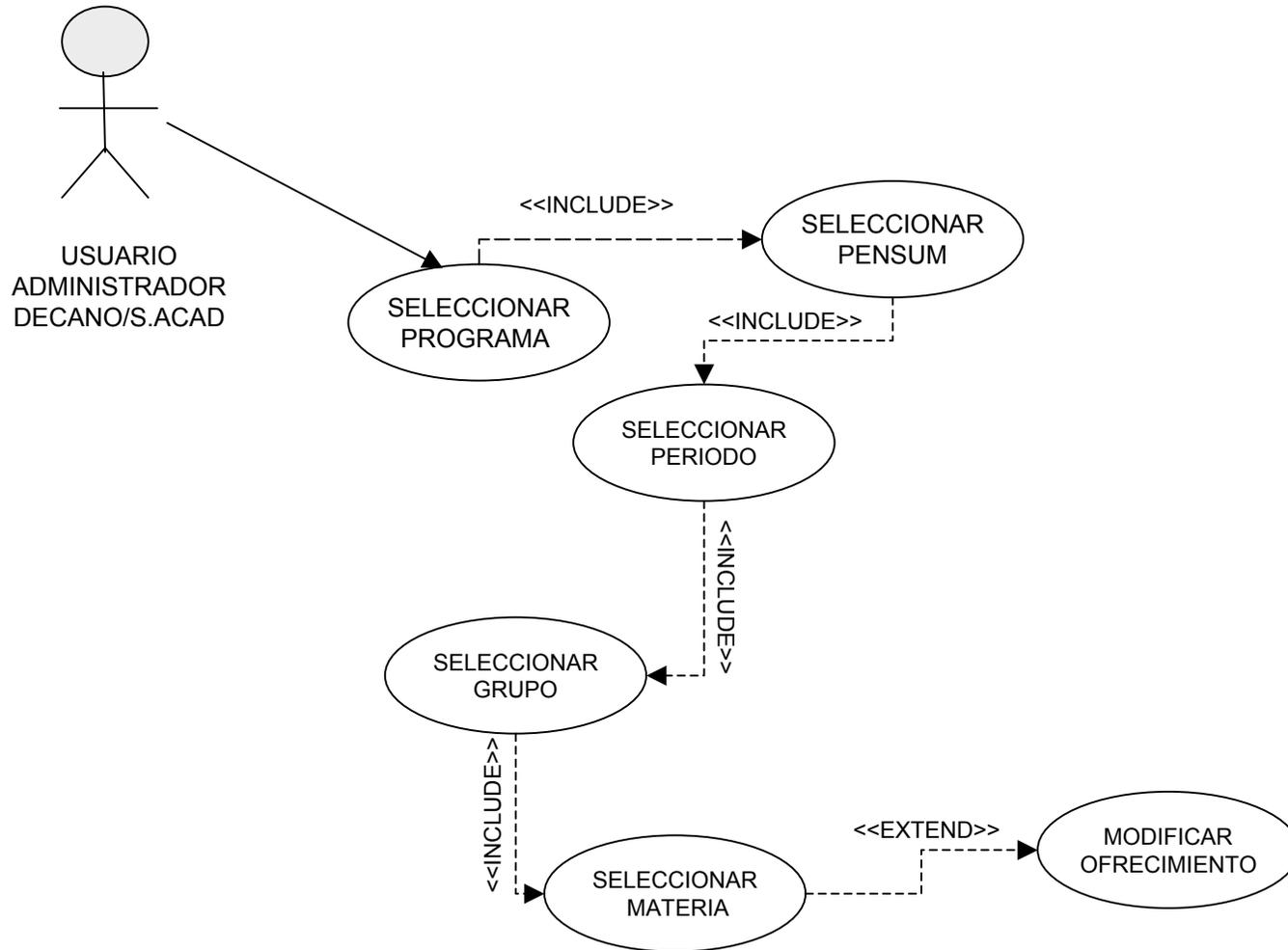
# ID: 34 CREAR OFRECIMIENTO



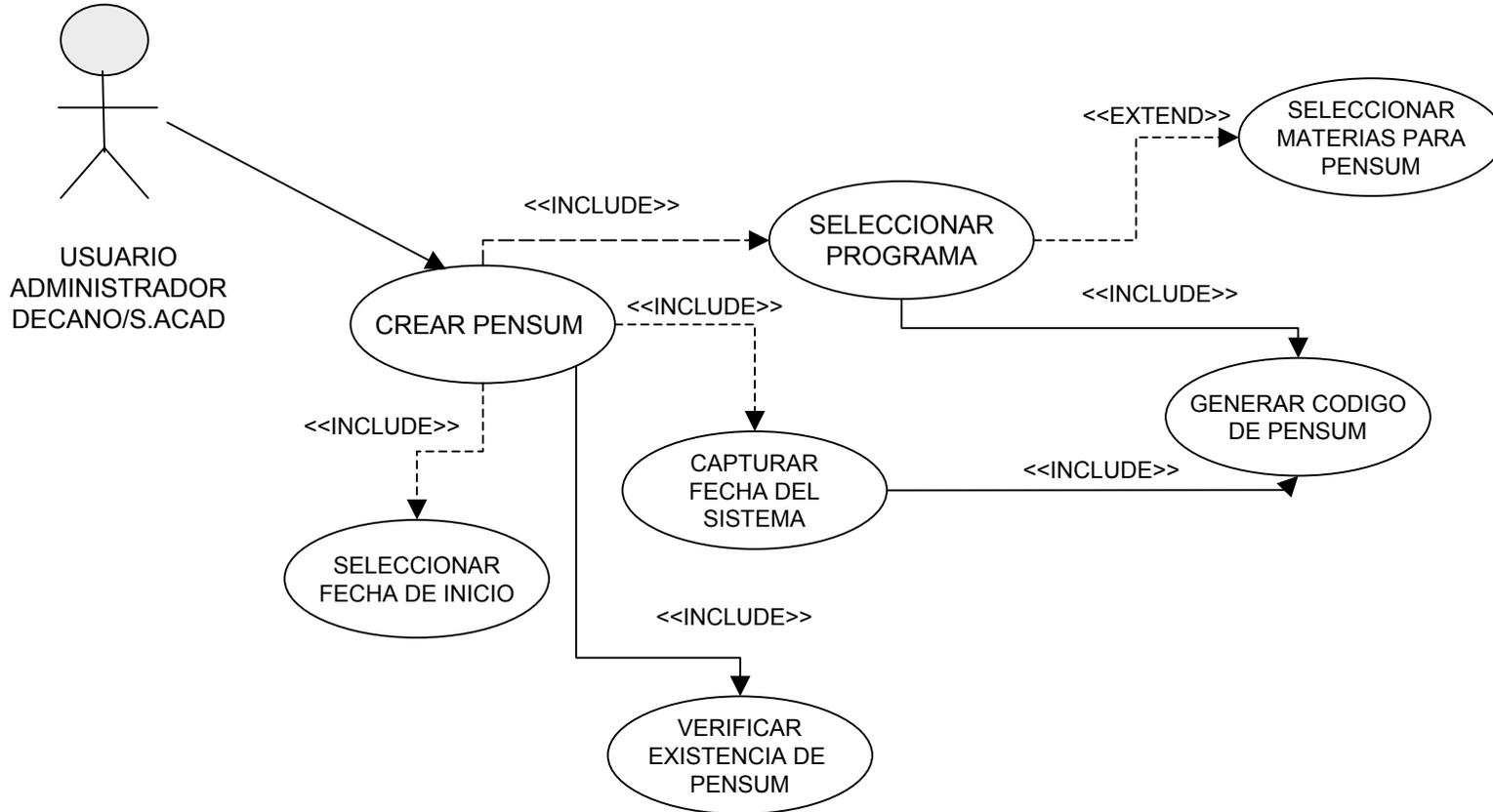
# Tarjeta Descriptiva Caso de Uso 34

<b>Id caso de uso</b>	<b>34</b>
<b>Nombre caso de uso</b>	Crear Ofrecimiento
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decanos, Sec académico
<b>Descripción</b>	Permite crear los ofrecimientos que se usaran durante el semestre académico
<b>Precondición</b>	Existencia de materias en el pensum a utilizar
<b>Postcondiciones</b>	Crear el ofrecimiento
<b>Flujo Normal</b>	1. Seleccionar Programa 2. Seleccionar Pensum. 3. Seleccionar el periodo o semestre 4. Seleccionar las materias a ofrecer 5. Asignar materia a grupo
<b>Flujos Alternativo</b>	6. Asignar docente a materia.
<b>Excepciones</b>	10. Crear Ofrecimiento Si no existen los ofrecimientos y docentes no se podra crear ningun horario.
<b>Incluidos</b>	
<b>Prioridad</b>	Alta
<b>Frecuencia de uso</b>	Baja
<b>Requerimientos especiales</b>	

# ID: 35 MODIFICAR OFRECIMIENTO



# ID: 36 CREAR PENSUM

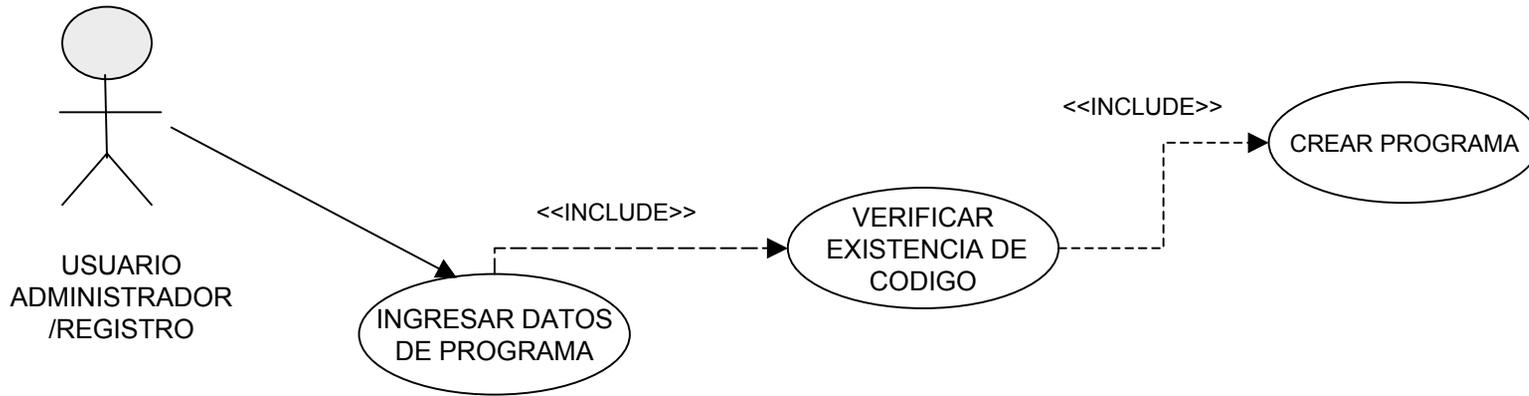


# Tarjeta Descriptiva Caso de Uso 36

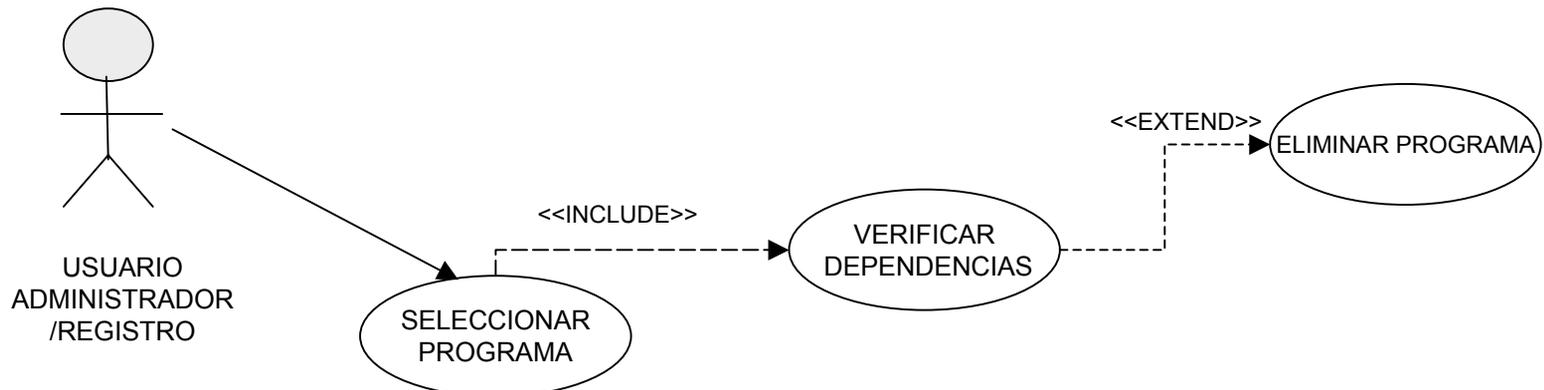
<b>Id caso de uso</b>	<b>36</b>
<b>Nombre caso de uso</b>	Crear Pensum
<b>Creado por</b>	David Agudelo & Juan Carlos Cortes
<b>Fecha de creación</b>	Septiembre 2004
<b>Actores</b>	Administrador, Decanos, Sec académico
<b>Descripción</b>	Permite crear pensum y adicionar materias a este.
<b>Precondición</b>	Existencia de programas académicos
<b>Postcondiciones</b>	Crear el pensum
<b>Flujo Normal</b>	<ol style="list-style-type: none"><li>1. Seleccionar programa</li><li>2. Sistema captura fecha del sistema.</li><li>3. Sistema crea código a partir del programa y la fecha</li><li>4. Se seleccionan las materias que harán parte del pensum</li></ol>
<b>Flujos Alternativo</b>	<ol style="list-style-type: none"><li>5. Se crea el Pensum</li></ol>
<b>Excepciones</b>	Si una persona desea crear mas de un pensum el mismo dia y para el mismo programa, el sistema no lo dejara crear.
<b>Incluidos</b>	
<b>Prioridad</b>	Normal
<b>Frecuencia de uso</b>	Baja
<b>Requerimientos especiales</b>	



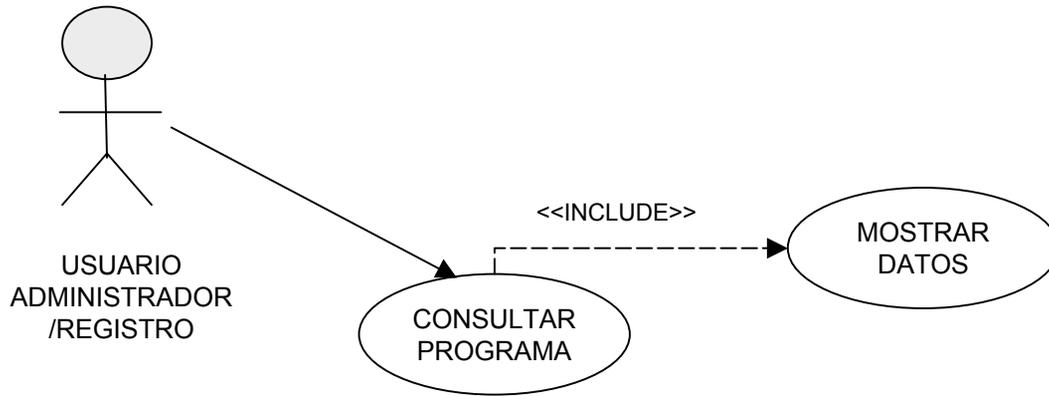
# CREAR PROGRAMA



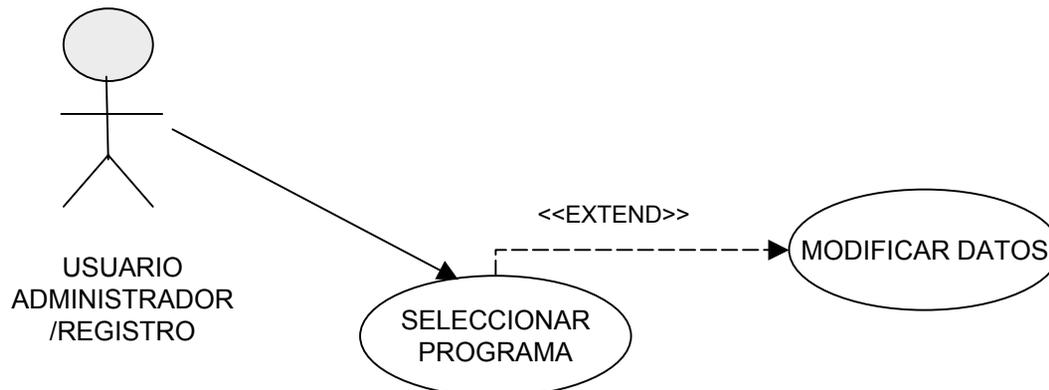
# ELIMINAR PROGRAMA



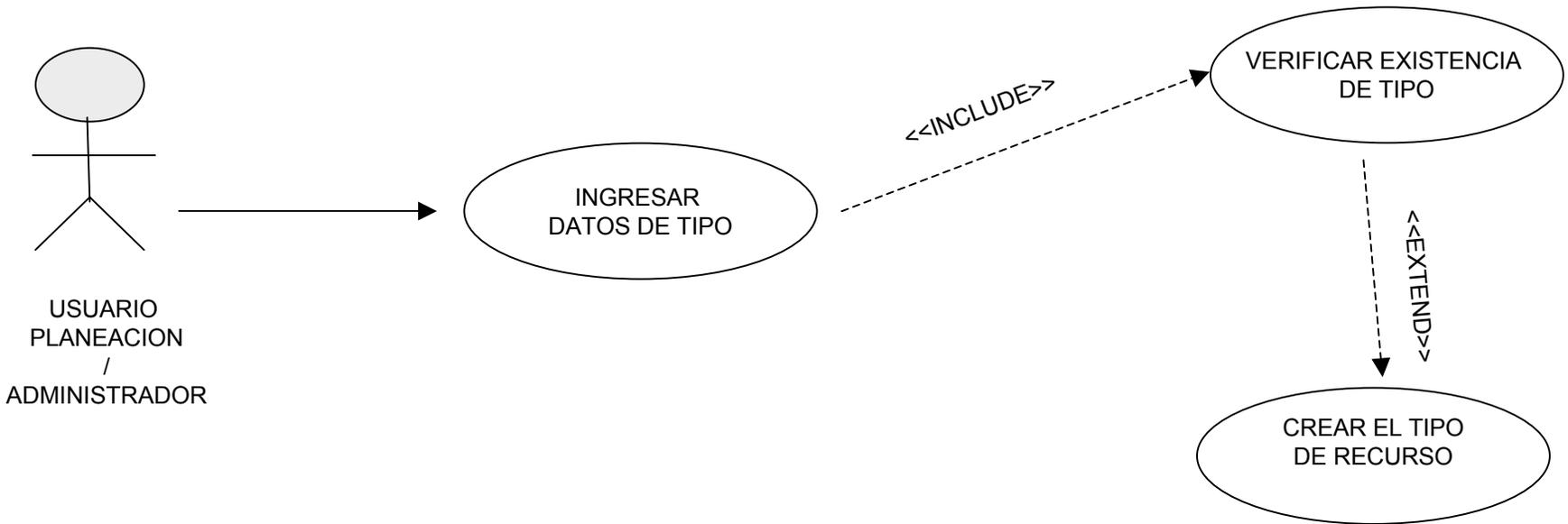
# CONSULTAR PROGRAMA



# MODIFICAR PROGRAMA



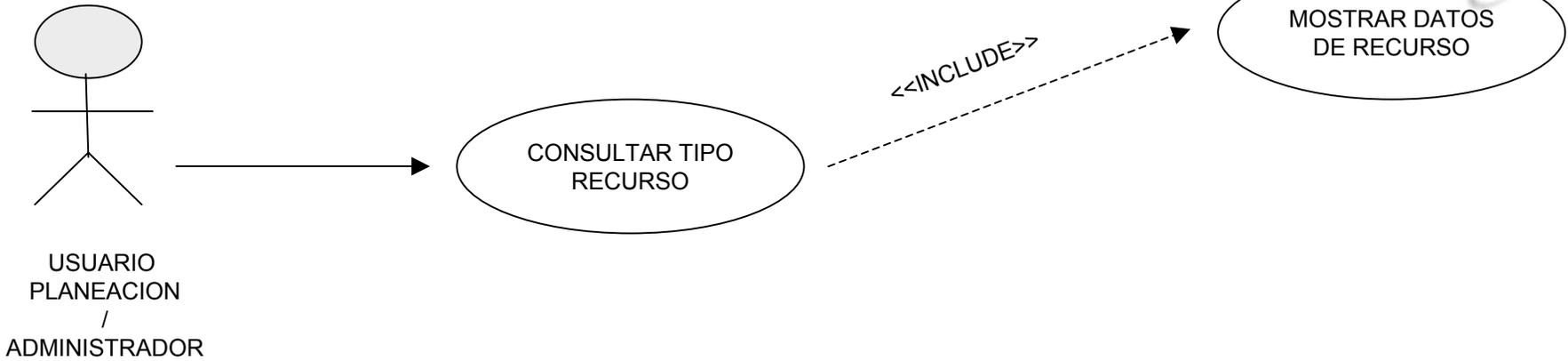
# CREAR TIPO RECURSOS



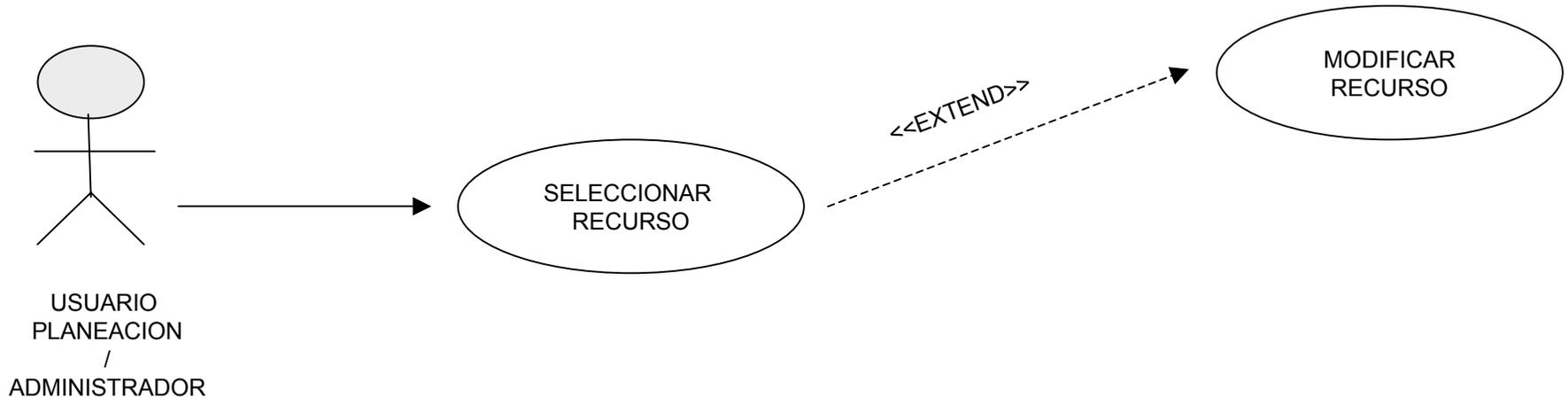
# ELIMINAR TIPO RECURSOS



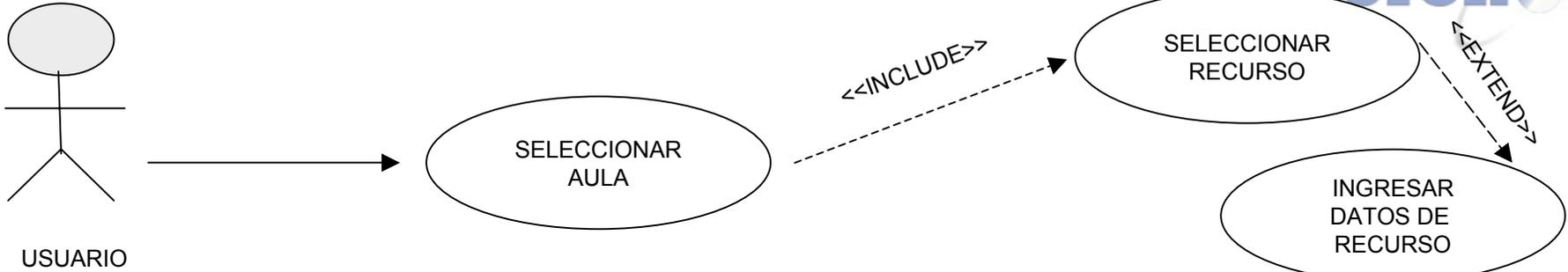
# CONSULTAR TIPO RECURSOS



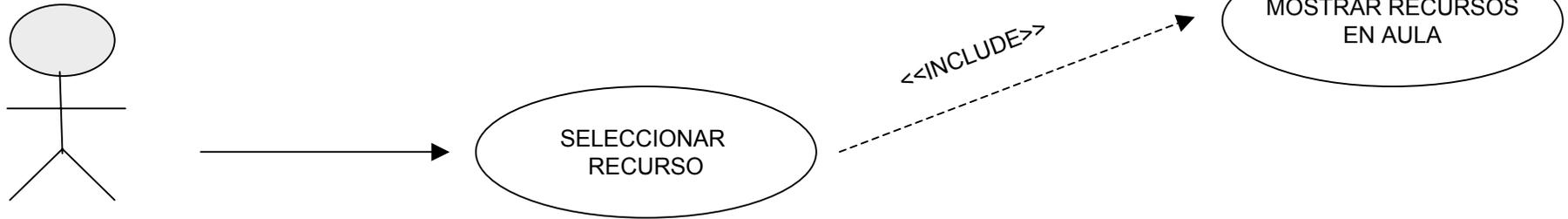
# MODIFICAR TIPO RECURSOS



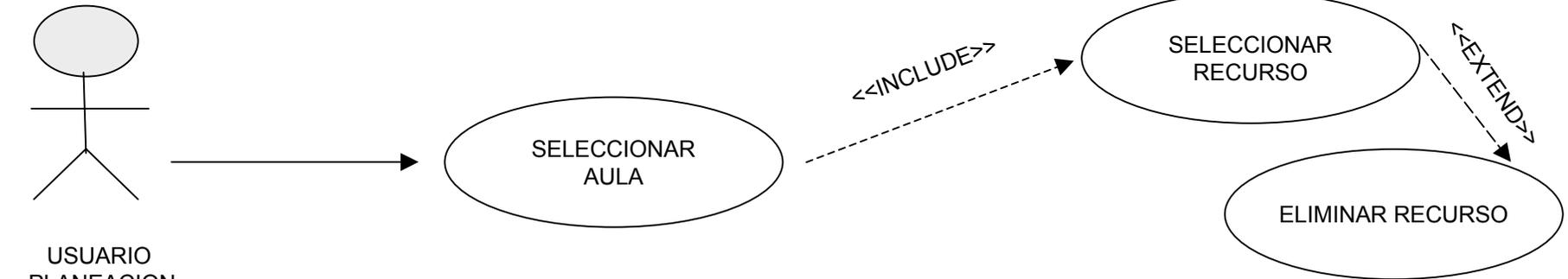
# CREAR RECURSO AULA



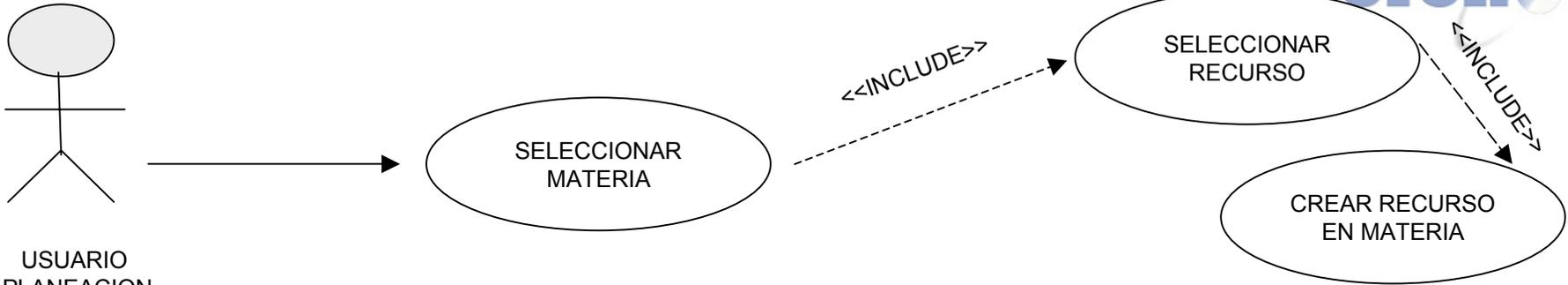
# CONSULTAR RECURSO AULA



# ELIMINAR RECURSO AULA



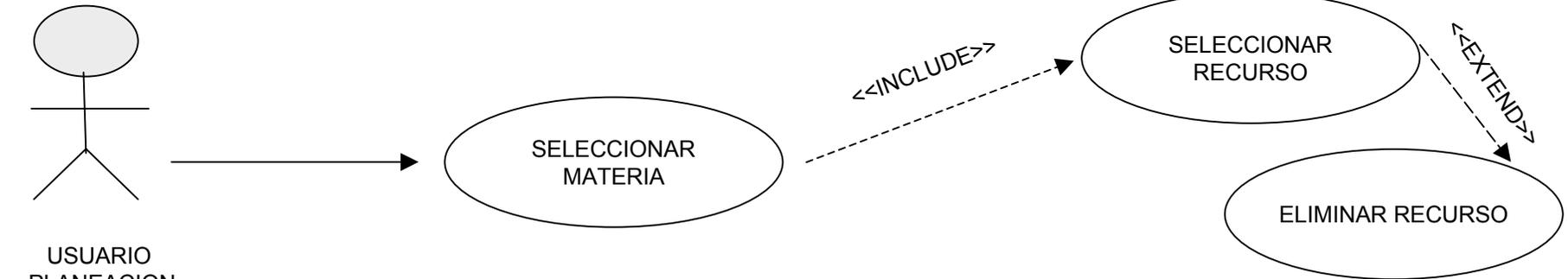
# CREAR RECURSO MATERIA



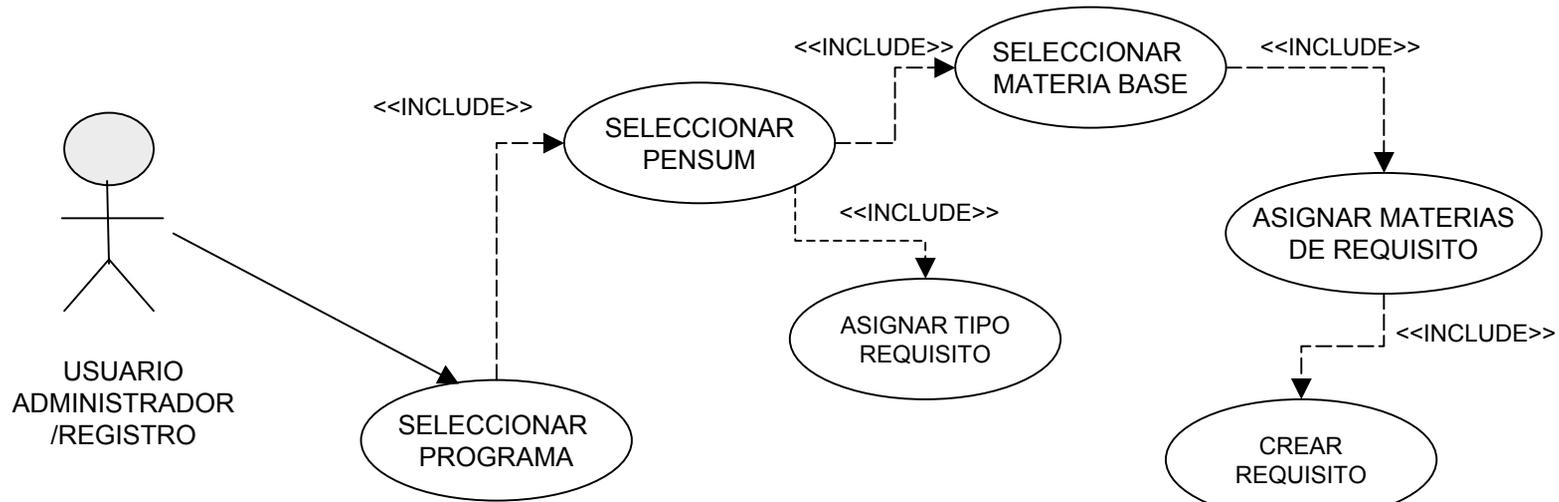
# CONSULTAR RECURSO MATERIA



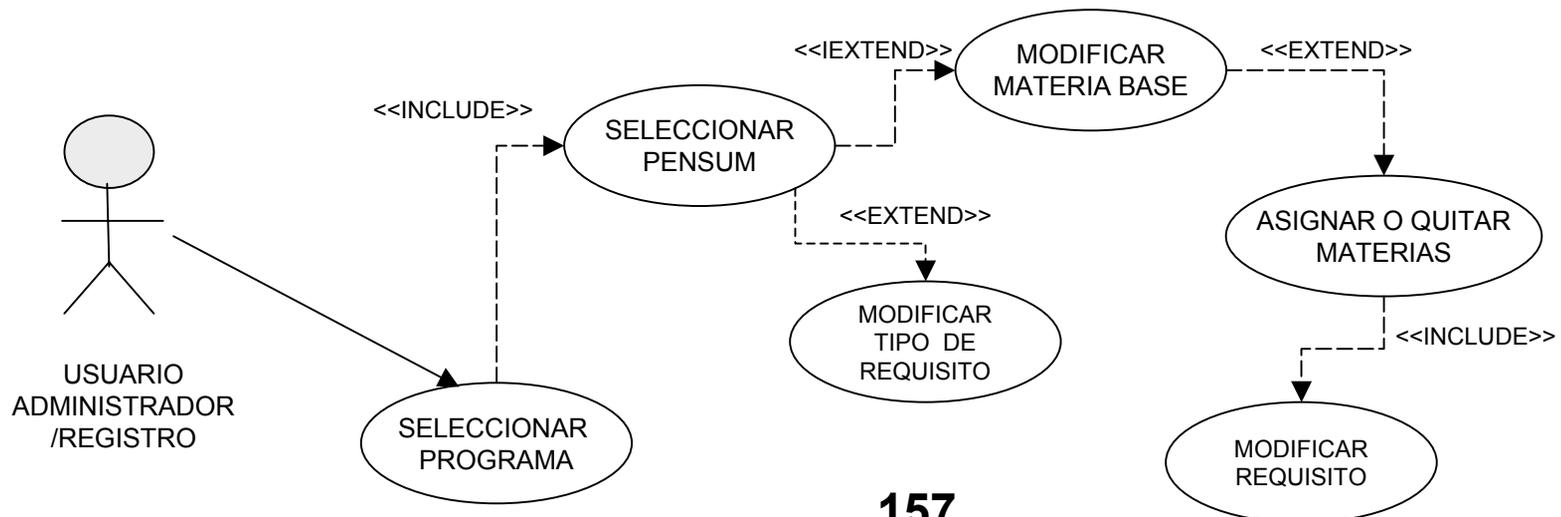
# ELIMINAR RECURSO MATERIA



# CREAR REQUISITO



# MODIFICAR REQUISITO

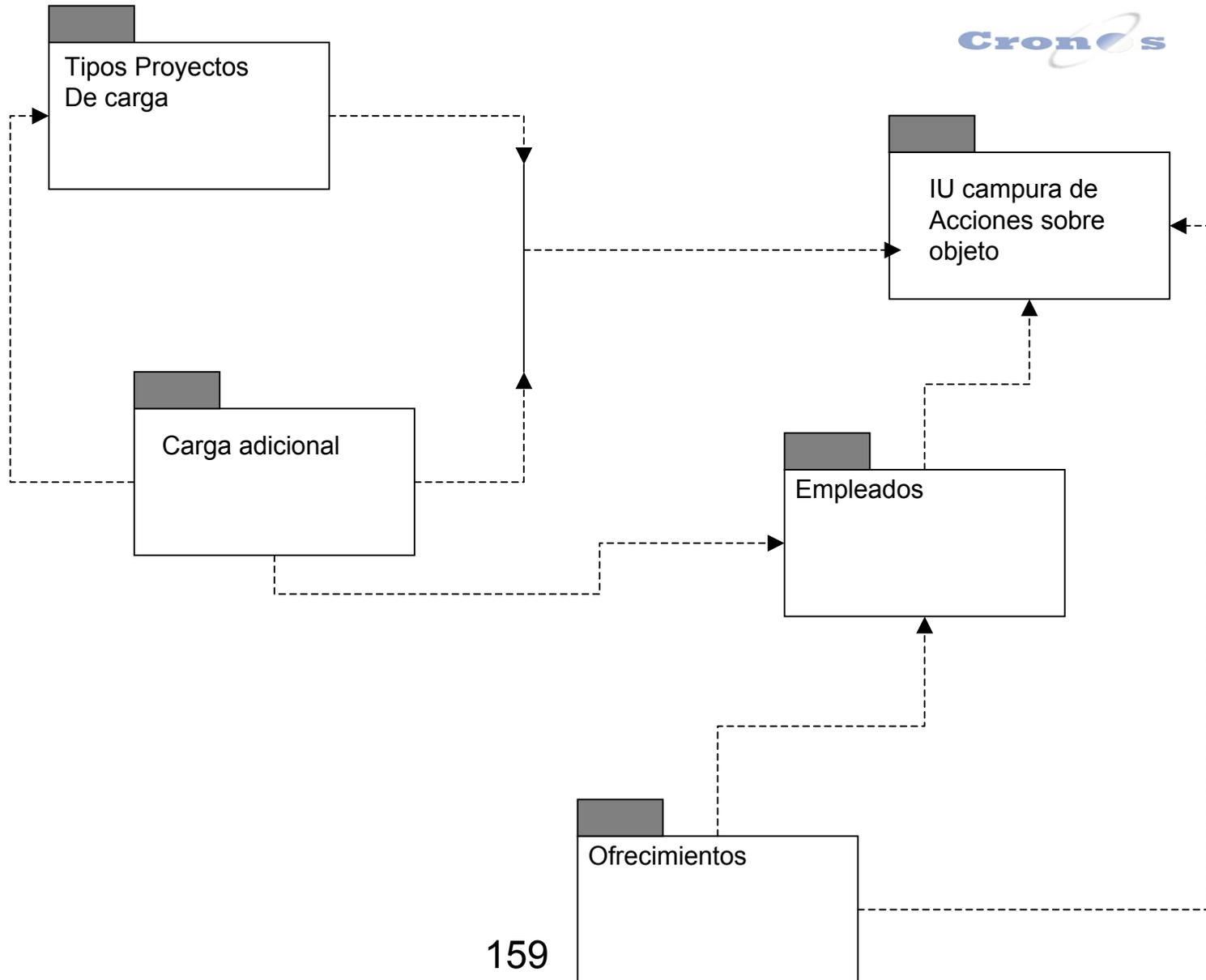


**ANÁLISIS Y DISEÑO**  
**11.4.6 DISEÑO DEL SISTEMA**  
**DIAGRAMA DE PAQUETES**

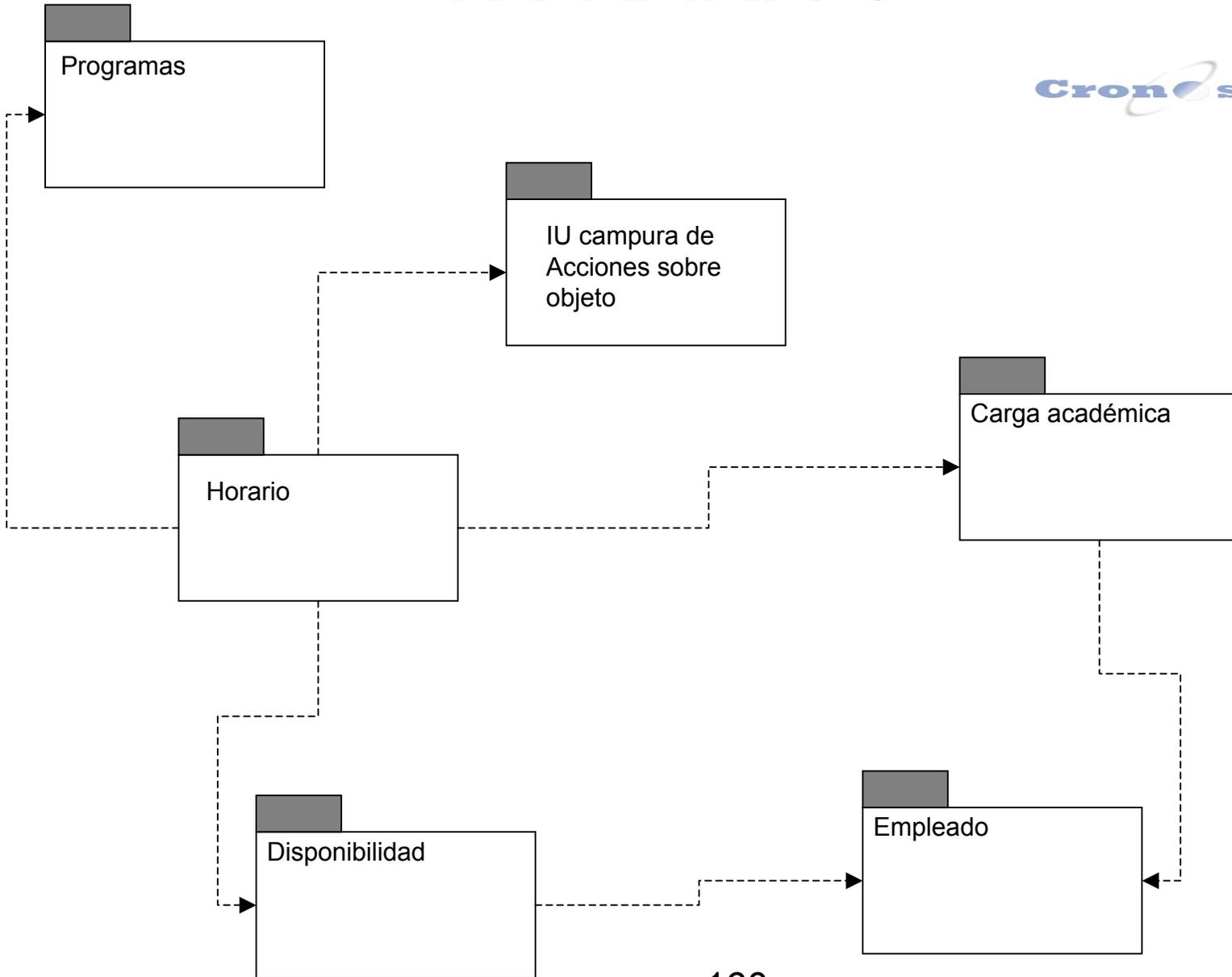
**CRONOS: SISTEMA DE CARGA ACADÉMICA Y PLANEACIÓN DE HORARIOS**



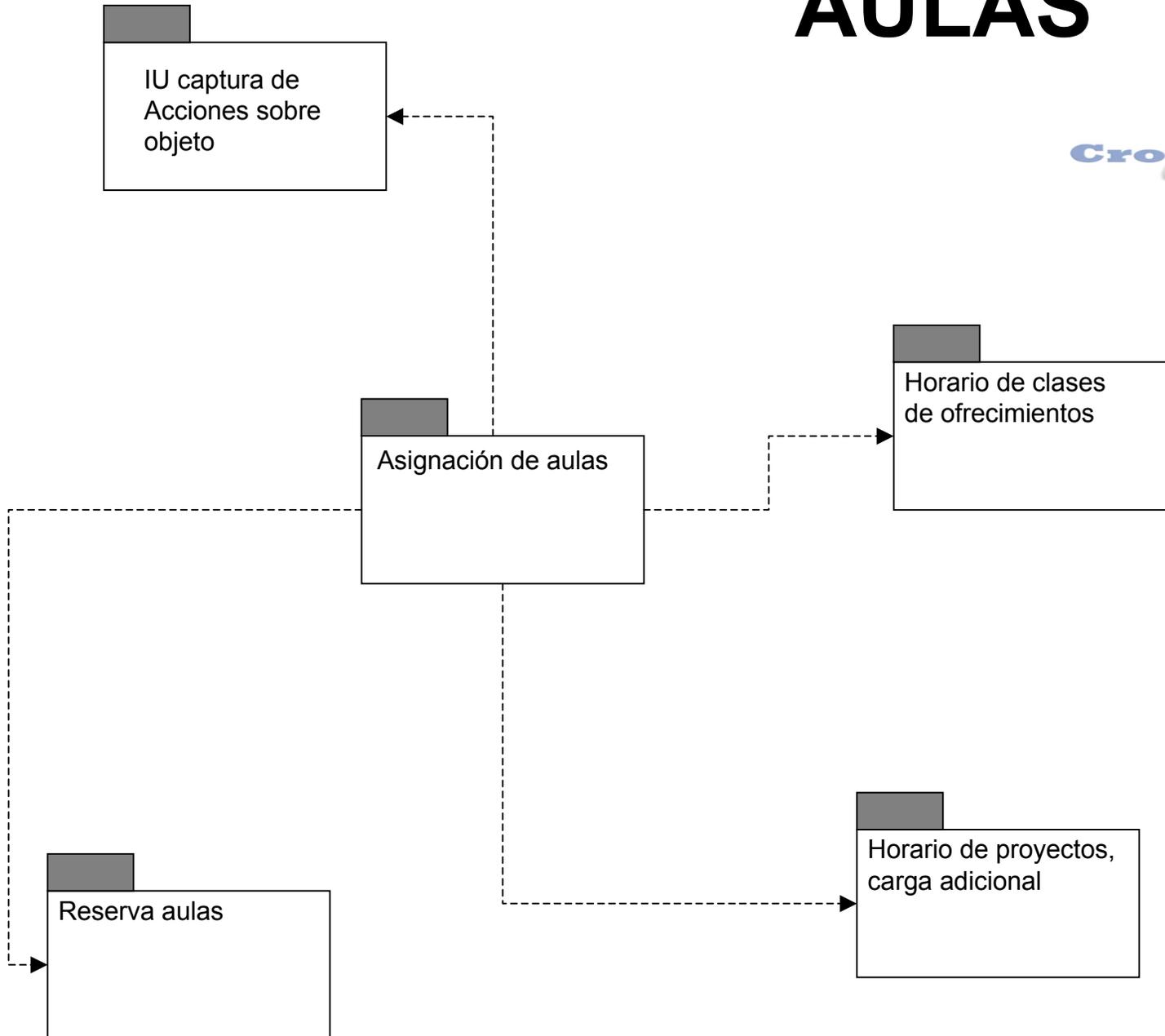
# CARGA ACADÉMICA



# HORARIOS



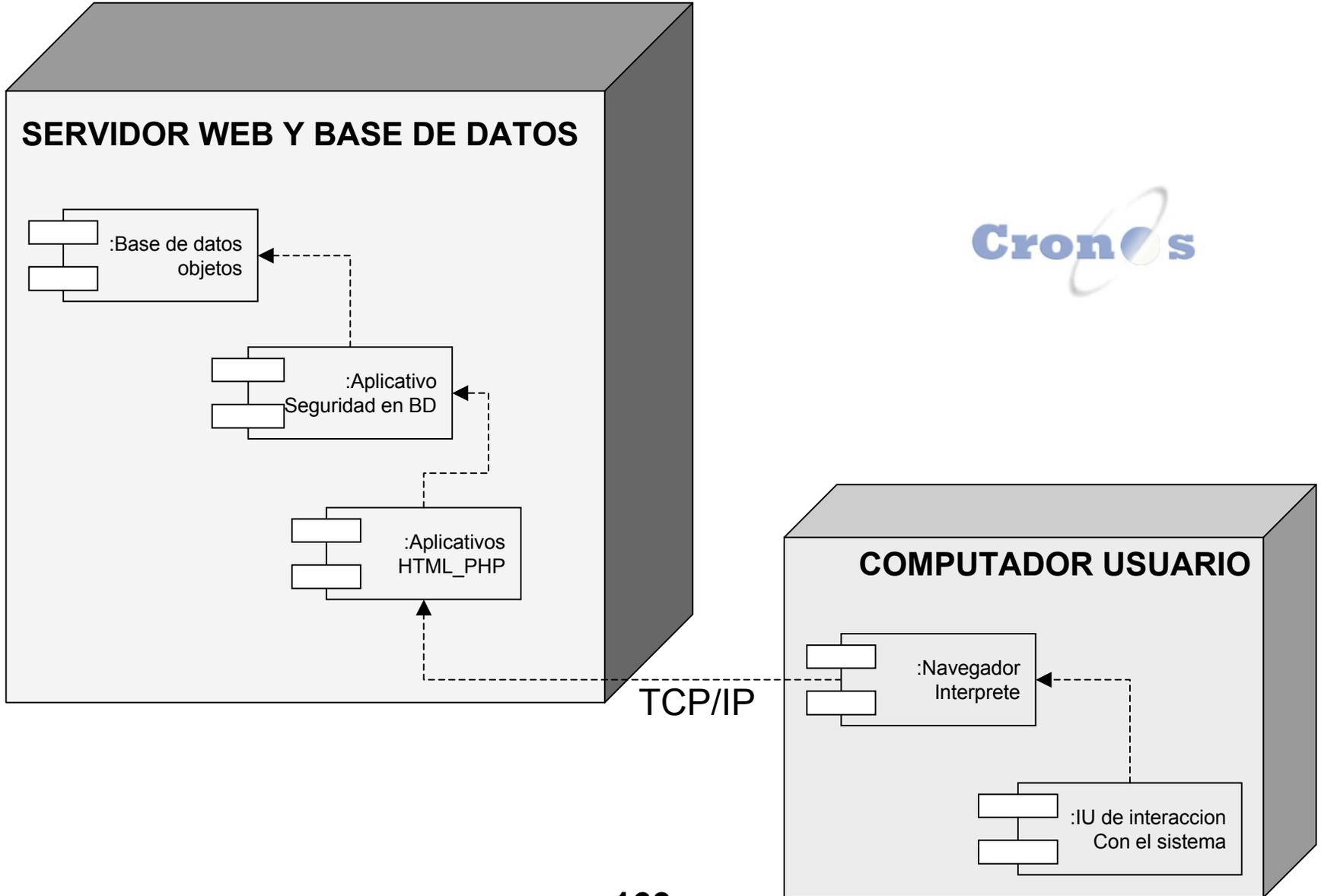
# AULAS



**ANEXO N**  
**ANÁLISIS Y DISEÑO**  
**DIAGRAMA DE DESPLIEGUE**



# CRONOS





**ANEXO P**  
**DESCRIPCIÓN DE PROCESOS EN LA ASIGNACIÓN DE HORARIOS**  
**METODOLOGÍA, TÉCNICAS DE CRUCES Y BLOQUEOS**

## **ANEXO P**

### **DESCRIPCIÓN DE PROCESOS EN LA ASIGNACION DE HORARIOS METODOLOGÍA, TÉCNICAS DE CRUCES Y BLOQUEOS**

Para la asignación de horarios a los ofrecimientos, se tienen 3 procesos generales que intervienen para consulta, inserción y borrado, pero que son analizados independientemente para su clara interpretación. Es importante tener en cuenta para el desarrollo web, se realiza una consulta inicial de información de la base de datos, los datos son llevados a matrices y al finalizar el proceso son almacenados en la base de datos, teniendo en cuenta que en este proceso de asignación se llevan a cabo unos bloqueos que impiden la alteración de los datos fuentes mientras la información se encuentra en memoria volátil. De igual forma se debe tener en cuenta que al realizar modificación en los datos, se realiza un refresco en la página web y en la matrices de información, lo cual implica modificaciones instantáneas.

#### **PROCESOS GENERALES DEL HORARIO**

##### **CONSULTA**

1. Consultar los ofrecimientos por facultad, programa, tipo de periodo (año, semestre, trimestre entre otros) y grupo
2. Consultar los docentes por materia por cada ofrecimiento de la consulta anterior que pueden orientar esta materia según la relación área docente.
3. Cargar en memoria por cada docente que ha sido previamente seleccionado su disponibilidad según día y hora.

##### **MODIFICACIÓN**

###### **Caso 1. Cambio en el Docente:**

Se libera este docente del ofrecimiento  
Se adicionan las horas del docente liberado en su disponibilidad  
Se eliminan los bloqueos a las otras materias que pueden ser orientadas el mismo día en el horario que se acaba de liberar.  
(Un grupo no puede ver más de una materia en igual horario)  
Se adicionan las horas que se acaban de liberar en el total de horas semanales por materia.  
Se asigna el nuevo docente  
Se carga la disponibilidad de horas por días, teniendo en cuenta los bloqueos generados por el horario de otras materias.

###### **Caso 2. Cambio de hora inicial en un día determinado**

Se realiza la liberación de la cantidad de horas que fue asignada anteriormente para esta materia en el día indicado.  
Se devuelven las horas que habían sido asignadas para esta materia a la disponibilidad del docente según día y cantidad de horas

Se devuelve el número de horas asignadas al total de horas disponibles para orientar la materia  
Se libera el bloqueo de horas según el día para las demás materias del grupo.

**Caso 3: Cambio en la cantidad de horas de asignación para una materia**

Se verifica que las horas permitidas para asignación no se crucen con otros

Se verifica que el acumulado de horas por materia no exceda el total de horas por semana de la materia

Se realiza la disminución o aumento en las horas de la disponibilidad del docente, así como la disminución o aumento en el total de horas disponibles para la materia.

Se realiza un bloqueo según la hora inicial y el número de horas para las demás materias que pueden ser orientadas el mismo día, pero no en las mismas horas ya asignadas.

**ELIMINACIÓN**

Corresponde a un caso especial de la modificación, en el cual la hora inicial se deja en vacío y se procede a liberar la disponibilidad del docente, la cantidad de horas totales de la materia y los bloqueos al resto de materias que pueden ser orientados en el mismo día.

Los bloqueos son reconstruidos cada que se cambie la hora inicial, la cantidad de horas y el docente.

ANEXO Q

**CRONOS: SISTEMA DE CARGA ACADÉMICA  
MANUAL DE USUARIO Y DEL PROGRAMADOR**

Por: David Marcelo Agudelo & Juan Carlos Cortes



**TABLA DE CONTENIDOS**

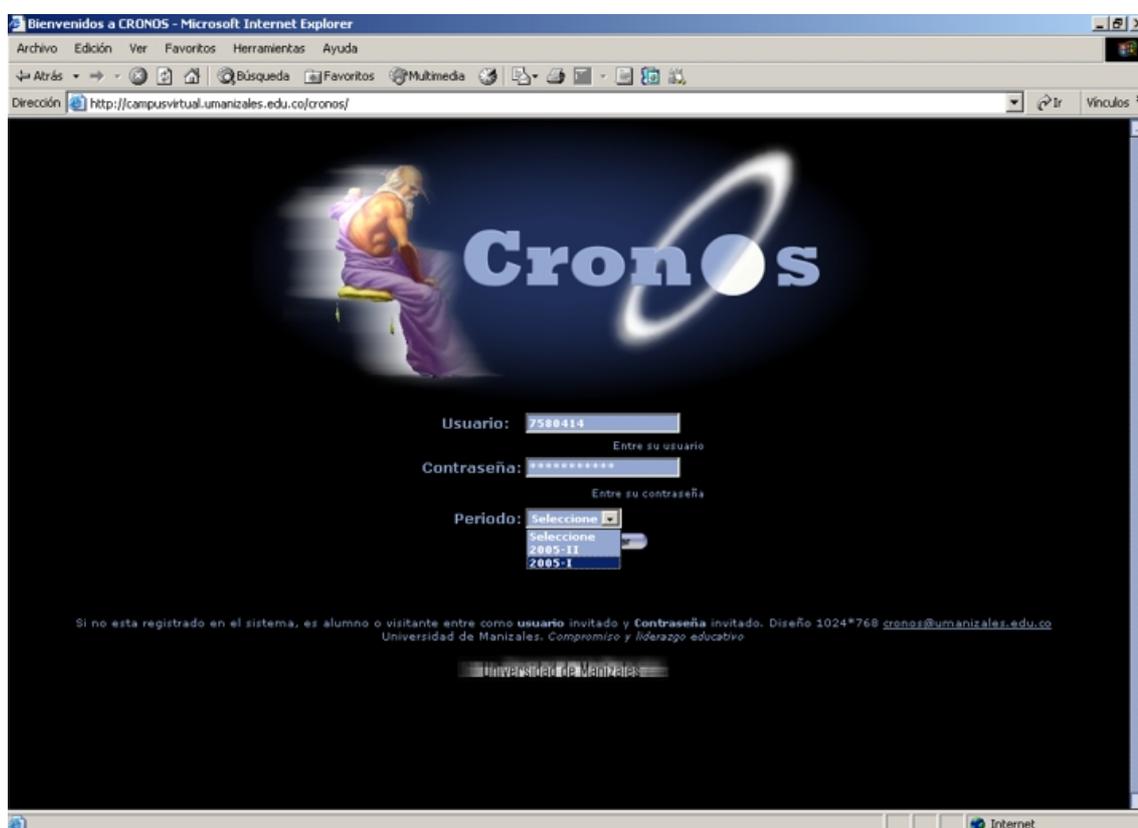
CRONOS:SISTEMA DE CARGA ACADEMICA.....	
MANUAL DE USUARIO .....	168
TABLA DE CONTENIDOS .....	168
Descripción de Interfaces.....	169
Error en Base de Datos.....	169
Descripción de la Interfaz acceso denegado.....	170
Funcionamiento Interfaz de usuarios.....	171
Distribución de la Interfaz en la pantalla.....	172
.....	
Descripción Interfaz de reportes.....	173
Descripción reporte de carga académica.....	173
Descripción de Acciones sobre Menús centrales.....	174
Perfiles de Usuario.....	176
Perfil Administrador.....	176
Perfil Planeación.....	176
Perfil Registro Académico.....	176
Perfil Decano.....	176
Perfil Secretario Académico.....	177
Perfil Secretarias.....	177
Perfil Docente.....	177
Acerca del Desarrollo.....	178
Sistema Operativo.....	178
Motor de Base de datos .....	178
Tipos de archivos.....	178
Recomendaciones.....	178
Manual del Programador.....	179
Manual de Instalación y ejecución de la aplicación.....	180
Descripción de scripts de instalación.....	181
Estructura de Tablas. Tarjetas descriptivas.....	183
Modelo de Objetos.....	197
Diccionario de Clases.....	198
Metodo de Horarios: Técnicas de cruce y bloqueo.....	206

## DESCRIPCIÓN DE INTERFACES

A continuación presentamos la descripción general de las interfaces de CRONOS, como operan sus validaciones, sus menús a partir del usuario activo, y las acciones sobre estas.

### Descripción de la Interfaz de Entrada

La siguiente interfaz (Figura 1) es la primera opción de interacción entre el usuario y el sistema de carga académica Cronos, aquí se deben suministrar el usuario, la clave y el periodo académico al cual desea acceder, todos los datos anteriores son requeridos, en caso de faltar alguno no se podrá acceder al sistema.



(Fig. 1)

### Descripción de la Interfaz de error en Base de Datos

La siguiente interfaz (Figura 2) solamente se activa cuando el sistema detecta alguno de los siguientes errores:

- En la pagina principal de cronos se ingresaron usuario y clave, pero no se selecciono el periodo académico(base de datos) al cual quería acceder.
- El segundo error es que durante el intercambio de información entre el usuario y el sistema se ha perdido la conexión con la base de datos, esto puede suceder si hay una caída de la red.



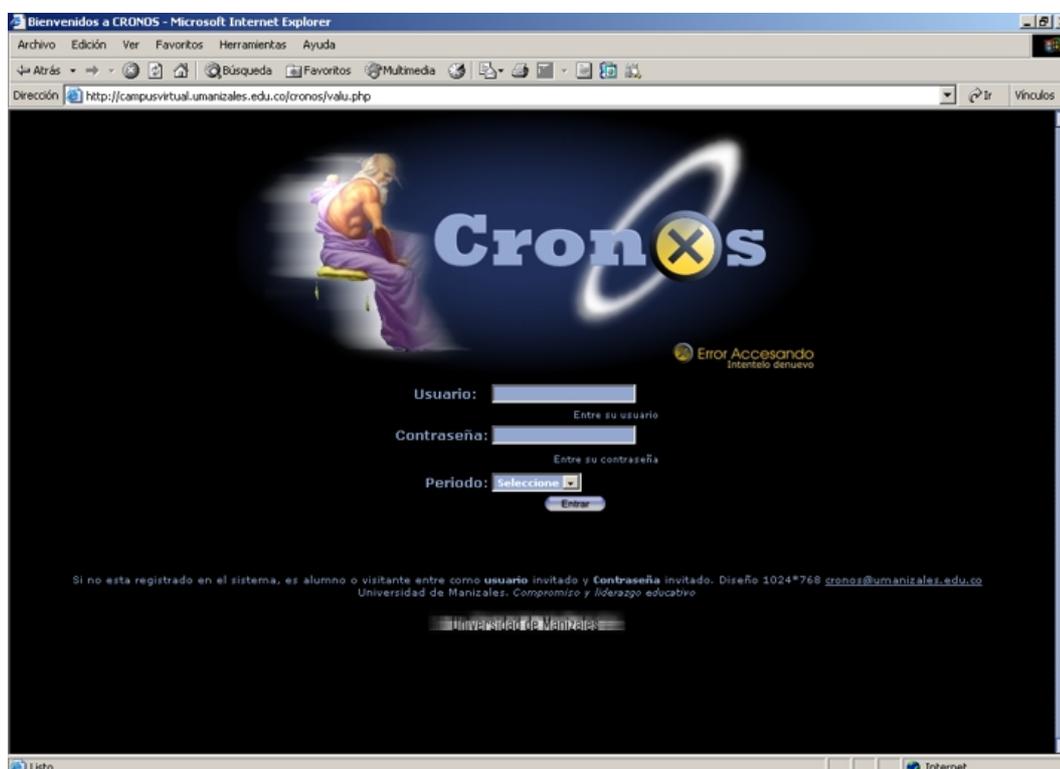
(Fig. 2)

### Descripción de la Interfaz acceso denegado

La siguiente interfaz (Figura 3) se activa en el momento que el script de seguridad detecta alguna de las siguientes inconsistencias:

- Error en el usuario
- Error en la clave
- Inserción de código inyección

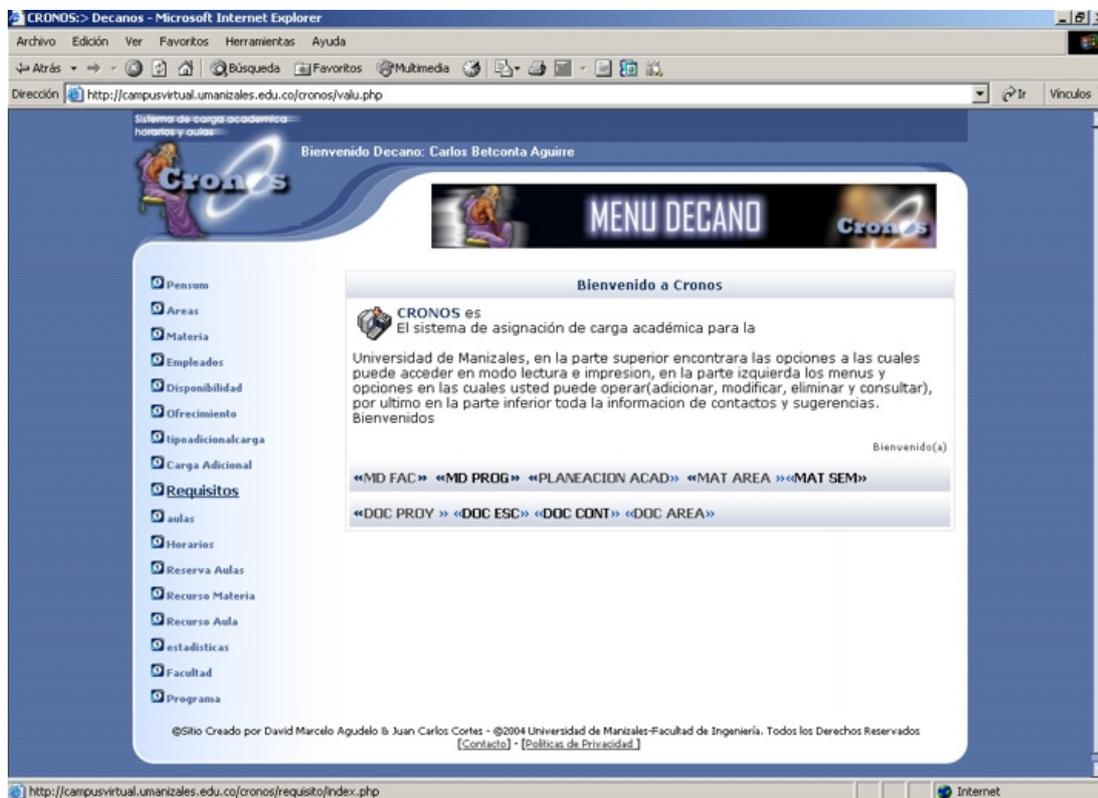
También es activado en el momento en que un usuario teniendo alguna URL interna intenta acceder a esta sin antes haberse registrado (iniciado sesión en la página principal de cronos)



(Fig. 3)

## Funcionamiento Interfaz de usuarios

A continuación se presenta la primera interfaz (Fig. 4) de usuario interna la cual es asignada por el sistema en el momento de verificar el perfil de quien accede.



(Fig. 4)

La interfaz detecta el perfil, muestra una breve presentación del sistema y a continuación el usuario podrá observar el menú de navegación en la parte izquierda para iniciar su interacción.

En la parte inferior de la pantalla encontrara información acerca de políticas de uso del sistema, la versión, copyright y los creadores.

## Distribución de la Interfaz en la pantalla

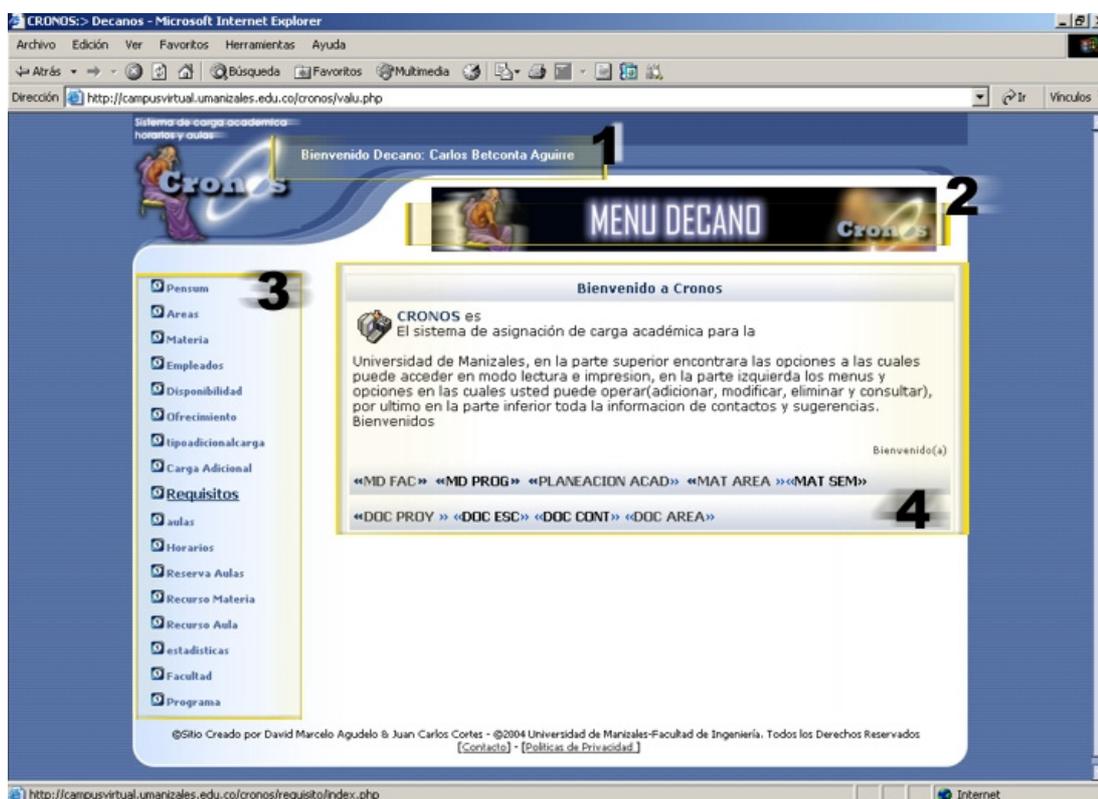
La siguiente imagen (Fig. 5) presenta una descripción de la distribución de la interfaz de los usuarios:

**Numero 1:** Nombre del usuario activo

**Numero 2:** Imagen de titulo, varia de acuerdo a la sección donde esta ubicado.

**Numero 3:** Menú de navegación principal, varia de acuerdo al perfil del usuario

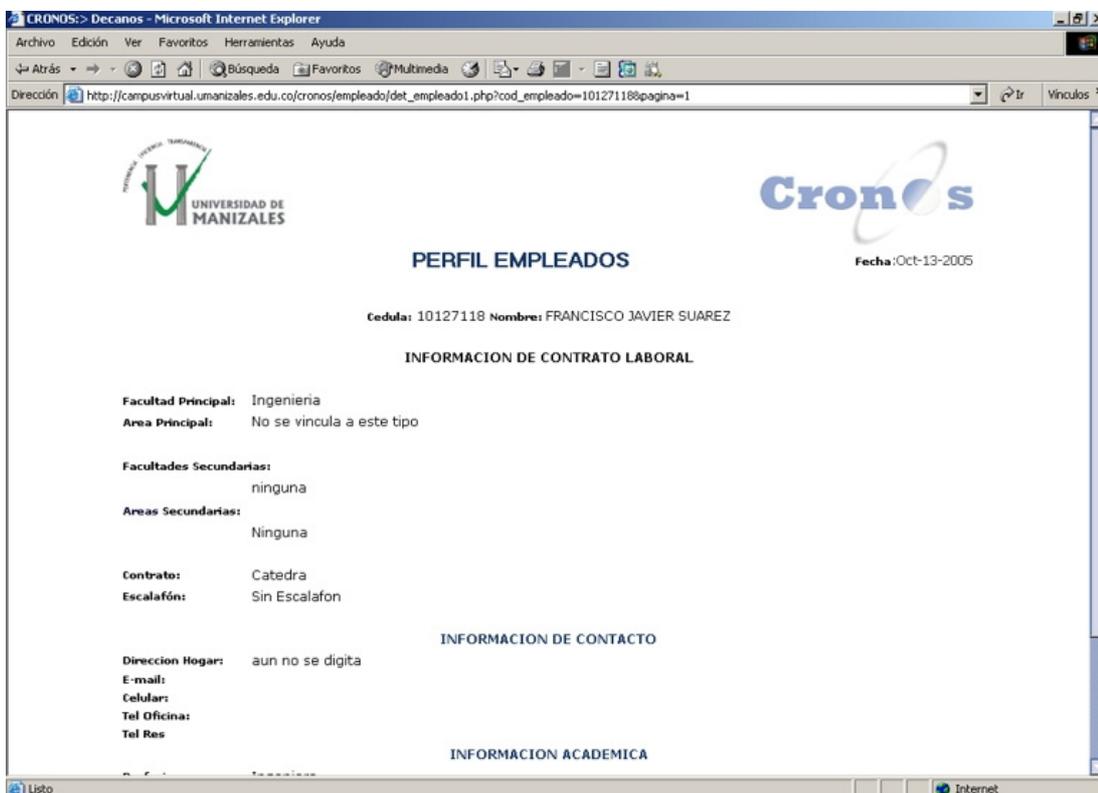
**Numero 4:** Menú de acciones sobre menús(Consultar, Crear, Modificar, Eliminar), este menú también varia y es explicado mas adelante



(Fig. 5)

## Descripción Interfaz de reportes

A continuación se presenta una típica interfaz (Figura 6) de reportes, mas liviana, con los logos de la Empresa(Universidad de Manizales) y del sistema(Cronos), todo bajo un solo lienzo(color blanco) para su impresión.



(Fig. 6)

## Descripción reporte de carga académica

La siguiente interfaz (Figura 7) presenta como queda la distribución de carga cuándo finalmente se ha asignado o como esta quedando. Es una Web igualmente limpia y liviana para imprimir, muestra:

Facultad, tipo de contratación(TC-MT-C), docente, y las horas por cada uno de los tipos de carga que maneja la facultad, al final de las columnas y las filas se muestra un total general por docente y por carga, además de las horas bonificación calculadas cuando supera el tope de horas del contrato. las celdas de color azul claro se activan solamente cuando aun no se ha cumplido el tope de horas de contratación con el propósito de indicar al coordinador de carga que aun faltan horas para su contrato.

CRONOS: Decanos - Microsoft Internet Explorer

Archivo Edición Ver Favoritos Herramientas Ayuda

Atrás Búsqueda Favoritos Multimedia

Dirección [http://campusvirtual.unamizales.edu.co/cronos/reportes/md\\_planeacion1.php](http://campusvirtual.unamizales.edu.co/cronos/reportes/md_planeacion1.php)

UNIVERSIDAD DE MANIZALES

**PLANEACION ACADEMICA** Fecha: Oct-13-2005

Facultad: Ingeniería  
Contrato: Tiempo Completo

Nombre Apellido	Hrs Cat	Hrs Ext	Hrs Inv	Hrs Asp	Hrs Adm	Hrs Cap	Hrs P.E	Hrs Com	Hrs Prep	Hrs Bonif	Tot
CARLOS BETANCOURT CORREA	0	0	0	0	0	0	0	0	0	0	0
DIEGO LOPEZ CARDONA	0	0	0	0	0	0	0	0	0	0	0
GERMAN GALLO GALLON	0	0	0	0	0	0	0	0	0	0	0
GERMAN WILLIAM LONDONO	0	0	0	0	0	0	0	0	0	0	0
JORGE ANDRES QUINTERO	0	0	0	5	0	3	0	0	0	0	8
JORGE HERNAN FRANCO	0	0	0	0	0	0	0	0	0	0	0
JOSE FERNANDO MEJIA CORREA	0	0	0	0	0	0	0	0	0	0	0
NESTOR JAIME CASTANO	0	0	0	0	0	0	0	0	0	0	0
WILSON LOPEZ FRANCO	0	0	0	0	0	0	0	0	0	0	0
<b>TOTALES</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>5</b>	<b>0</b>	<b>3</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>16</b>

« 1 »

Regresar

©Sitio Creado por David Marcelo Agudelo & Juan Carlos Cortes - ©2004 Universidad de Manizales-Facultad de Ingeniería. Todos los Derechos Reservados  
[Contacto] - [Políticas de Privacidad]

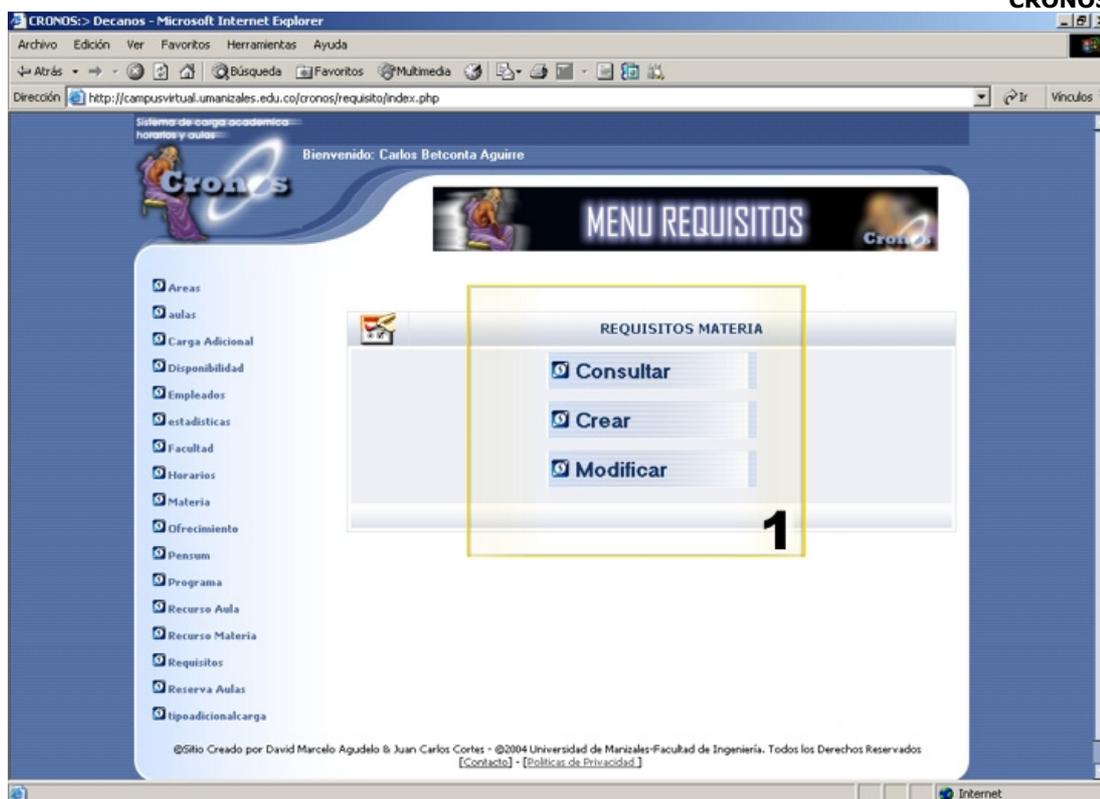
(Fig. 7)

### Descripción de Acciones sobre Menús centrales

Cada ítem de navegación del sistema cronos tiene sus propios menús, estos a su vez cuando son activados despliegan una serie de acciones en el centro de la Web Este submenú de acciones puede variar dependiendo del perfil de usuario y de las mismas opciones que puede mostrar el menú.

Un ejemplo sería que una disponibilidad tiene 3 acciones Consultar , Crear Modificar, mientras que el menú aulas tiene 4 acciones Consultar, Crear, Modificar y eliminar, esta última no aparece en el menú de disponibilidad. Cuando nos referimos a las acciones a partir de los perfiles un ejemplo sería que un docente no puede ni crear, ni eliminar, ni modificar horarios, pero si puede consultarlos.

A continuación presentamos la interfaz de acciones sobre menús (Figura 8), el usuario que está activo es un decano. Y la interfaz es la que se genera al dar clic sobre el menú "Requisito" en la parte izquierda de la pantalla.



(Fig. 8)

### **Perfiles de Usuario**

La actividad laboral de la Universidad de Manizales es basada en varios estamentos tales como Registro académico, El jefe de planeación, Rector, decanos, secretarios de facultad, cada uno con diferentes perfiles y diferentes funciones.

Por ejemplo un decano o secretario académico no puede asignar las aulas a sus horarios, o un docente no puede crear su propio horario de clases sin verificar los de los demás, también las facultades no pueden ser creadas por los decanos. Es por eso que el sistema Cronos en base a su análisis muestra un menú de navegación diferente para cada perfil, con las opciones y acciones en las que se puede interactuar. A continuación presentamos los menús a los que puede tener acceso cada perfil:

1. Perfil Administrador
2. Perfil Planeación
3. Perfil Registro Académico
4. Perfil Decano
5. Perfil Secretario Académico
6. Perfil Secretarias
7. Perfil Docente
8. Perfil Invitado

**Perfil Administrador**

Menús de acceso:

1. pensum
2. area
3. materia
4. empleado
5. carga
6. disponibilidad
7. ofrecimiento
8. requisito
9. aula
10. horario
11. reservas
12. recurso
13. estadísticas
14. facultad
15. programa

Acciones: Todas (Consultar, Crear, Modificar, Eliminar)

**Perfil Planeación**

1. Aulas

Acciones: Todas (Consultar, Crear, Eliminar)

**Perfil Registro Académico**

1. area
2. estadística
3. facultad
4. materia
5. programa

Acciones: Todas (Consultar, Crear, Modificar, Eliminar)

**Perfil Decano**

1. facultad
2. programa
3. area
4. materia

Acciones: Consultar

5. Recurso
6. Pensum
7. Requisito
8. Disponibilidad
9. Empleado
10. Ofrecimiento
11. Horario
12. Carga

13. Reservas
14. Estadísticas
15. Acciones: Todas (Consultar, Crear, Modificar, Eliminar)

### **Perfil Secretario Académico**

1. facultad
  2. programa
  3. area
  4. materia
- Acciones: Consultar*

5. Recurso
  6. Pensum
  7. Requisito
  8. Disponibilidad
  9. Empleado
  10. Ofrecimiento
  11. Horario
  12. Carga
  13. Reservas
  14. Estadísticas
- Acciones: Todas (Consultar, Crear, Modificar, Eliminar)*

### **Perfil Secretarias**

1. Area
  2. Aula
  3. Empleado
  4. Materia
  5. Horario
  6. Ofrecimiento
  7. Pensum
  8. facultad
  9. Programa
  10. Recursos
  11. Requisito
- Acciones: Consultar*

12. Reserva aulas
- Acciones: Todas (Consultar, Crear, Eliminar)*

### **Perfil Docente**

1. Area
2. Aula
3. Empleado
4. Materia
5. Horario
6. Ofrecimiento
7. Pensum

8. facultad
  9. Programa
  10. Recurso
  11. Requisito
  12. Reserva aula
  13. disponibilidad
- Acciones: Consultar*

## **ACERCA DEL DESARROLLO**

### **Sistema Operativo**

El sistema operativo donde opera el sistema es Linux, lo que no significa que no funcione en otros sistemas. El sistema puede ser instalado en SO como Windows

### **Motor de Base de datos**

PostgreSQL 8 , versión de pruebas PostgreSQL 7.3 y 8, aunque el desarrollo incluyo modelos PEAR de PHP que permiten que el sistema funcione transparentemente en otros motores como MySQL y ORACLE

### **Tipos de archivos**

Los archivos que se manejan en esta aplicación son Java Script y PHP y complementos de Ajax

### **Validaciones Internas**

Las validaciones son manejadas bajo estructuras directas de PHP y funciones HTML para permitir su visualización sin problemas en cualquier navegador, ya que en la actualidad no todos los navegadores bajo cualquier sistema operativo soportan funciones JAVASCRIPT, precisamente Firefox V1.5 en Linux

### **Plataformas de visualización y ejecución**

El sistema puede ser visualizado desde cualquier navegador hasta cualquier sistema operativo, la forma ideal de visualizarlo es con configuraciones de pantalla 1024\*768 o superior

Las pruebas se hicieron sobre Netscape, Opera, Safari, Mozilla, Konkeror, Firefox y sistemas operativos Windows, Linux, Sun, Mac OS.

*NOTA: los clientes de mayor uso de la plataforma Cronos son Usuarios de Internet Explorer.*

### **Recomendaciones**

Usar como navegador Internet Explorer Superior a la Versión 5 o Firefox versión 1.5 o Superior  
Configuración de Pantalla 1024\*768.



***MANUAL DEL PROGRAMADOR***  
**CRONOS: Sistema de asignación de carga académica.**  
**Universidad de Manizales**

## MANUAL DE INSTALACIÓN Y EJECUCIÓN DE LA APLICACIÓN

Esta aplicación es basada en un desarrollo de base de datos con interfaces de Web bajo construcción dinámica, por lo tanto no necesita de un archivo ejecutable o de instalación como tal, si no que se basa en la ejecución de scripts de base de datos de extensión .sql.

A continuación se describen cada uno de los scripts en el orden en que deben ejecutarse.

### CONSIDERACIONES PREVIAS A LA EJECUCIÓN DE LOS SCRIPTS.

Atendiendo a los requerimientos lógicos y físicos del sistema nombrados en el manual de usuario y requerimientos de la documentación del proyecto recordamos que antes de ejecutar los scripts se debe tener los siguientes servicios lógicos y físicos instalados:

1. Servicio web Apache(versiones libres o propietarias)
2. Servicio de Base de datos Postgresql 7.3 o superior
3. Computador o servidor bajo las siguientes características:
  - a. Procesador Pentium III o superior
  - b. Memoria 256 o Superior
  - c. Disco Duro de 10 Gigas o Superior

### EJECUCIÓN DE SCRIPTS DE APLICACIÓN

Deberá ejecutar las siguientes líneas de código para crear usuarios de la base de datos y dicha base de datos, a continuación se muestran las líneas que debe escribir indiferentemente del sistema operativo para iniciar el proceso de montaje de la aplicación:

#### CREACIÓN DE USUARIO DE LA BASE DE DATOS

```
CREATE USER nombre usuario with password 'clave usuario';
```

#### CREACIÓN DE LA BASE DE DATOS:

```
Createdb nombre de la base de datos with password clave owner usuario
```

Una vez creados el usuario y la base de datos debe ingresar a ella con el nombre de usuario y la clave directamente desde el prompt que entrega el servicio de base de datos postgres y anteponiendo la palabra reservada psql:

Psql nombreBD usuario clave + enter

#### **Descripción a la línea anterior:**

Psql: palabra reservada de Postgres en línea de comandos eternos

nombreBD: el nombre que le dio a la base de datos

Usuario: usuario que usara la base de datos desde el sistema hacia la web

Clave: clave de acceso a la base de datos.

Ya en el interior de la base de datos se encontrara con un prompt diferente al del sistema operativo, aquí es donde usted deberá correr los scripts con la siguiente línea de comandos:

```
\i nombrescrip.sql
```

NOTA: Recuerde que los scripts deben estar almacenados exactamente en el directorio(windows) o filesystem(Linux) desde donde esta accediendo a la creación y ejecución de la base de datos, de lo contrario encontrara el siguiente error *"no such file or directory"*

### **DESCRIPCIÓN DE SCRIPTS DE INSTALACIÓN**

1. **borrado.sql:**  
este script borra todas las tablas y corre todos los scripts de inserción y permisos
2. **borro.sql:**  
este script contiene las líneas de ejecución para borrar todas las tablas de la base de datos.
3. **cronos.sql:**  
este script contiene las líneas de ejecución para crear todas las tablas de la base de datos.
4. **modoconsulta.sql:**  
este script se ejecuta solo cuando se inicia un nuevo periodo académico y es aplicado al periodo inmediatamente anterior, pues de esta manera si quieren revisar versiones pasadas de la carga académica solo puedan consultar y no modificar.
5. **insertarn.sql:**  
este script contiene los datos obligatorios con los que debe iniciar la base de datos siempre que se vaya a instalar por primera vez tales como tipos de contrato, perfiles, escalafón, horas, números etc.
6. **inser2.sql:**  
este script contiene datos de verificación de tablas, clasificación de días, créditos y niveles para programas.
7. **inseradmin.sql:**  
script de inserciones básicas para el funcionamiento del perfil Administrador.
8. **inserregistro.sql:**  
script de inserciones básicas para el funcionamiento del perfil Registro.
9. **insertplaneacion.sql:**  
script de inserciones básicas para el funcionamiento del perfil planeación.

10. **inserttalento.sql:**  
script de inserciones básicas para el funcionamiento del perfil talento humano.
11. **insertdecano.sql:**  
script de inserciones básicas para el funcionamiento del perfil Decano.
12. **insertsacad.sql:**  
script de inserciones básicas para el funcionamiento del perfil Secretario Académico.
13. **insertempleado.sql:**  
script de inserciones básicas para el funcionamiento del perfil Empleados diferentes a docentes.
14. **insertsecretaria.sql:**  
script de inserciones básicas para el funcionamiento del perfil Secretaria.
15. **insertdocente.sql:**  
script de inserciones básicas para el funcionamiento del perfil Docente.
16. **insertinvitado.sql:**  
script de inserciones básicas para el funcionamiento del perfil invitados y estudiantes.
17. **permisoapache.sql:**  
script que otorga todos los permisos al usuario de base de datos.

## 1. ARCHIVOS WEB

Los archivos web PHP deben quedar almacenados en un directorio o subdirectorio del servidor apache(httpd) y estar en ejecución el servicio web y servicio de base de datos. También debe tener instalado el paquete de PEAR-PHP que viene desde las versiones de PHP 5+ instalado, versiones anteriores no.

## 2. ESTRUCTURA DE TABLAS Y SCRIPTS DE LA BASE DE DATOS

Tabla	tipocarga	
<b>Descripción</b>	Contiene la clasificación de tipos de carga adicional como investigación, comités, preparación de clases entre otros.	
<b>Script</b>	create table tipocarga( cod_tipo varchar(2) primary key, descripción varchar(18));	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_tipo	Consecutivo de tipo de carga
	descripción	Descripción del tipo de carga

Tabla	dia	
<b>Descripción</b>	Contiene los días de la semana	
<b>Script</b>	create table dia( cod_dia varchar(1) primary key, dia varchar (9) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_dia	Consecutivo del día de la semana
	dia	Nombre del día de la semana

Tabla	estado	
<b>Descripción</b>	Contiene los posibles estados de los recursos físicos para las aulas	
<b>Script</b>	create table estado( cod_estado varchar(1) primary key, estado varchar (15) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_estado	Consecutivo para los estados
	estado	Descripción del estado

Tabla	tiporequisito	
<b>Descripción</b>	Contiene la clasificación de las posibles relaciones entre las materias del pensum	
<b>Script</b>	create table tiporequisito( cod_tipor varchar(1) primary key, descripción varchar(20) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_tipor	Consecutivo de tipo de requisito
	descripción	Descripción el requisito

Tabla	horas	
<b>Descripción</b>	Contiene las horas asociadas a un consecutivo para ser asignados.	
<b>Script</b>	create table horas( hora time, horap varchar(5), numero varchar(2));	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	hora	Hora
	horap	Hora formateada para ser mostrada
	numero	Consecutivo de identificación iniciado en 10

Tabla	escalafon	
<b>Descripción</b>	Clasificación del nivel profesional dentro de la Universidad	
<b>Script</b>	create table escalafon( cod_escalafon integer primary key, descripcion varchar(70) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_escalafon	Consecutivo para el escalafón docente
	Descripción	Descripción del escalafón docente

Tabla	contrato	
<b>Descripción</b>	Clasificación del tipo de contrato laboral en la Universidad	
<b>Script</b>	create table contrato( cod_contrato varchar(3) primary key, nombre varchar(70) not null, num_horas numeric(3) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_contrato	Consecutivo para los contratos laborales
	nombre	Descripción del contrato laboral
	num_horas	No. de horas máximo dentro de una clasificación de contrato laboral

Tabla	perfil	
<b>Descripción</b>	Clasificación de usuarios del sistema, que poseen permisos equivalentes y la web asociada para el perfil.	
<b>Script</b>	create table perfil( cod_perfil varchar(12) primary key, descripcion varchar(40) not null, url_home varchar(40) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_perfil	Consecutivo para los perfiles
	Descripción	Descripción del perfil
	url_home	Direcciones de las páginas web a las cuales puede acceder un determinado perfil

Tabla	tipodependencia	
<b>Descripción</b>	Clasificación de las entidades aptas para generar programas dentro de la universidad	
<b>Script</b>	create table tipodependencia ( codigo varchar(1) primary key, descripcion varchar(12) not null);	
Estructura	Campo	Descripción
	codigo	Consecutivo del tipo de dependencia
	Descripción	Descripción del tipo de dependencia

Tabla	zona	
<b>Descripción</b>	Clasificación de las zonas de la universidad para agrupar las aulas.	
<b>Script</b>	create table zona( cod_zona varchar(2) primary key, descripcion varchar(50) not null);	
Estructura	Campo	Descripción
	cod_zona	Consecutivo de las zonas
	Descripción	Descripción de las zona

Tabla	facultad	
<b>Descripción</b>	Sección de una universidad que engloba estudios de una misma rama.	
<b>Script</b>	create table facultad( tipo varchar(1) references tipodependencia(codigo), cod_fac varchar(2) primary key, nom_fac varchar(40) not null, zona varchar (2) references zona(cod_zona));	
Estructura	Campo	Descripción
	tipo	Si es facultad o dependencia
	cod_fac	Código único para la facultad o dependencia
	nom_fac	Nombre de la dependencia o facultad
	zona	Código de la zona donde se encuentra ubicada la facultad o dependencia

Tabla	tipoprograma	
<b>Descripción</b>	Clasificación de los tipos de programa para la Universidad	
<b>Script</b>	create table tipoprograma( cod_tipop varchar(2) primary key, descripcion varchar(18));	
Estructura	Campo	Descripción
	cod_tipop	Consecutivo para los tipos de programa
	Descripción	Descripción de los tipos de programa

Tabla	periodicidad	
<b>Descripción</b>	Clasificación de los tipos de periodicidad.	
<b>Script</b>	create table periodicidad( cod_periodo varchar(1) primary key, descripcion varchar(18));	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_periodo	Consecutivo para la periodicidad
	Descripción	Descripción de la periodicidad

Tabla	jornada	
<b>Descripción</b>	Clasificación del día	
<b>Script</b>	create table jornada( cod_jornada varchar(1) primary key, descripcion varchar(18));	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_jornada	Consecutivo para la jornada
	Descripción	Descripción de la jornada

Tabla	programa	
<b>Descripción</b>	Es el conjunto de actividades orientadas a la formación profesional en una determinada área del saber y a la obtención de un título profesional.	
<b>Script</b>	create table programa( cod_fac_p varchar(2) references facultad(cod_fac), tipo_programa varchar(2) references tipoprograma(cod_tipop), nom_prog varchar(60) not null, cod_prog varchar(2) primary key, periodicidad varchar(10) references periodicidad(cod_periodo), cantidad varchar(2) not null, reg_icfes varchar(15), jornada varchar(6) not null, hora_inicio_clases time);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_fac_p	Código de la Facultad
	tipo_programa	Código del tipo de programa
	nom_prog	Nombre del Programa
	cod_prog	Código del Programa
	periodicidad	Tipo de Periodicidad
	cantidad	Cantidad de niveles según la periodicidad
	reg_icfes	Registro del icfes para el programa
	jornada	Jornada del progra
hora_inicio_clases	Hora inicio de clases	

Tabla	empleado	
<b>Descripción</b>	Personas que tienen acceso al sistema	
<b>Script</b>	<pre> create table empleado(     cod_empleado varchar(12) primary key,     nom_empleado varchar(40) not null,     profesion varchar(20) not null,     cod_contrato varchar(2) references contrato(cod_contrato),     escalafon integer references escalafon(cod_escalafon),     direccion varchar(40),     email varchar(30),     celular varchar(10),     tel_oficina varchar(10),     exten varchar(4),     tel_residencia varchar(10),     clave varchar(8) not null,     cod_perfil varchar(1) references perfil(cod_perfil) not null,     fecha date,     estado varchar(6),     pregrado varchar(40),     postgrado1 varchar(40),     postgrado2 varchar(40));                     </pre>	
<b>Estructura</b>	Campo	Descripción
	cod_empleado	Cédula
	nom_empleado	Nombre completo
	profesion	Profesión
	cod_contrato	Código del contrato asignado
	escalafon	Escalafón
	direccion	Dirección
	email	Correo electrónico
	celular	Celular
	tel_oficina	Teléfono de la oficina
	exten	Extensión de la oficina
	tel_residencia	Teléfono de residencia
	clave	Clave para el acceso a la información
	cod_perfil	Tipo de perfil al que se encuentra vinculado
	fecha	Fecha de ingreso
	estado	Estado
	pregrado	Pregrado
	postgrado1	Postrado y/o especialización
	Postgrado2	Postrado y/o especialización

Tabla	<b>empleadofacultad</b>	
<b>Descripción</b>	Empleados asignados a una facultad	
<b>Script</b>	<pre>create table empleadofacultad(     cod_facultad varchar(2) references facultad(cod_fac),     cod_empleado varchar(12) references     empleado(cod_empleado),     tipo varchar(1),     constraint pk_facemple primary key     (cod_facultad,cod_empleado));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_facultad	Código de la facultad
	cod_empleado	Cédula del empleado
	tipo	Tipo de relación entre el empleado y la facultad (P)rimaria o (S)ecundaria

Tabla	<b>area</b>	
<b>Descripción</b>	Agrupación de materias de un campo específico de conocimiento.	
<b>Script</b>	<pre>create table area(     cod_fac varchar(2) references facultad(cod_fac),     cod_area numeric(2),     descripcion varchar(40) not null,     constraint pk_area primary key(cod_fac,cod_area));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_fac	Código de la facultad a la que pertenece el área
	cod_area	Código del area
	Descripción	Descripción del area

Tabla	<b>empleadoarea</b>	
<b>Descripción</b>	Empleados asignados en un área de conocimiento	
<b>Script</b>	<pre>create table empleadoarea(     cod_facultad varchar(2),     cod_area numeric(2),     cod_empleado varchar(12) references empleado     (cod_empleado),     tipo varchar(1),     constraint pk_emplearea primary     key(cod_facultad,cod_area,cod_empleado));  alter table empleadoarea add constraint fk_area foreign key(cod_facultad,cod_area) references area(cod_fac,cod_area);</pre>	

<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_facultad	Facultad a la que pertenece el área
	cod_area	Código del área
	cod_empleado	Cédula del empleado
	tipo	Tipo de relación entre el empleado y el área (P)rimaria o (S)ecundaria

<b>Tabla</b>	<b>pensum</b>	
<b>Descripción</b>	Plan estudiantil creado para los programas de cada facultad y está compuesto por un conjunto de asignaturas adecuadamente sistematizadas que el estudiante debe cursar y aprobar en el transcurso de su carrera.	
<b>Script</b>	<pre>create table pensum(     cod_prog_pe varchar(2) references     programa(cod_prog),     cod_pensum varchar(20) primary key,     descripcion varchar(80) not null,     fecha date);</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_prog_pe	Código del programa
	cod_pensum	Código del pensum
	Descripción	Descripción del pensum
	fecha	Fecha de creación del pensum

<b>Tabla</b>	<b>Materia</b>	
<b>Descripción</b>	Asignaturas para los pensum asociadas a un área	
<b>Script</b>	<pre>create table materia(     cod_facultad varchar(2),     cod_programa varchar(2) references     programa(cod_prog),     cod_materia varchar(12) primary key,     nom_mat varchar(20) not null,     num_creditos varchar(1) references creditos(numero),     no_horaste numeric(2),     no_horaspr numeric(2),     semestre varchar(2),     cod_area numeric(2));  alter table materia add constraint fk_materia foreign key(cod_facultad,cod_area) references area(cod_fac,cod_area);</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_facultad	Código de la facultad
	cod_programa	Código del programa
	cod_materia	Código de la materia

	nom_mat	Nombre de la materia
	num_creditos	Numero de créditos mínimos para aprobar la materia
	no_horaste	No. de horas teóricas
	no_horaspr	No. de horas prácticas
	semestre	Nivel en el cual se encuentra la materia
	cod_area	Código del area a la cual corresponde la materia

Tabla	materiapensum	
<b>Descripción</b>	Materias asociadas a un determinado pensum	
<b>Script</b>	<pre>create table materiapensum(   cod_mat_mp varchar(12) references   materia(cod_materia),   cod_pensum_mp varchar(20)references pensum   (cod_pensum),   num_grupos varchar(1),   constraint pk_mape primary key   (cod_mat_mp,cod_pensum_mp));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_mat_mp	Código de la materia
	cod_pensum_mp	Código del pensum
	num_grupos	No. de grupos que se ofrecerán de esta materia

Tabla	requisito	
<b>Descripción</b>	Relación de requisito entre las materias de un pensum	
<b>Script</b>	<pre>create table requisito(   codigomateriabase varchar(12) ,   codigomaterianecesita varchar(12) ,   codigopensum varchar(20),   tipo varchar(12) not null,   constraint pk_requisito primary key(codigomateriabase,   codigomaterianecesita,codigopensum));  alter table requisito add constraint fk_re_matpen foreign key(codigomateriabase,codigopensum) references materiapensum (cod_mat_mp,cod_pensum_mp);</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	codigomateriabase	Materia a analizar
	codigomaterianecesita	Materia a relacionar con la materia base
	codigopensum	Pensum en el cual se creará la relación entre materias
	tipo	Tipo de requisito

Tabla	<b>tipoadicionalcarga</b>	
<b>Descripción</b>	Contiene todos los proyectos y cargas adicionales con sus respectivos requerimientos.	
<b>Script</b>	<pre>create table tipoadicionalcarga(   cod_concepto_ti numeric(2) primary key,   nom_concepto varchar(40) not null,   Descripción varchar(70),   num_horas numeric(3),   num_docentes numeric(2),   facultad varchar(2) references facultad(cod_fac));</pre>	
Estructura	Campo	Descripción
	cod_concepto_ti	Código del tipo de carga adicional
	nom_concepto	Nombre del concepto de carga adicional
	Descripción	Descripción de la carga adicional
	num_horas	No. de horas totales para la carga adicional
	num_docentes	No. de docentes de la carga adicional
	facultad	Código de la facultad a la que pertenece la carga adicional

Tabla	<b>adicionalcarga</b>	
<b>Descripción</b>	Contiene la asignación de la carga adicional a un empleado.	
<b>Script</b>	<pre>create table adicionalcarga(   cod_concepto numeric(2) references   tipoadicionalcarga(cod_concepto_ti),   cod_docente varchar(12) not null,   num_horas numeric(3) not null);</pre>	
Estructura	Campo	Descripción
	cod_concepto	Código del tipo de carga adicional
	cod_docente	Cédula del empleado
	num_horas	No. de horas asignadas al empleado para la carga adicional

Tabla	<b>ofrecimiento</b>	
<b>Descripción</b>	Contiene las materias con el grupo a ser ofrecido a los estudiantes	
<b>Script</b>	<pre>create table ofrecimiento(   cod_materia varchar(12),   cod_pensum varchar(12),   grupo numeric(1) not null,   num_alumnos numeric(2) not null,   aprobados numeric(2),   reprobados numeric(2),   jornada varchar(1) not null,   estado varchar(9),   fecha_inicializacion date,</pre>	

	<pre> fecha_finalizacion date, constraint pk_ofrecimiento primary key (cod_materia,cod_pensum, grupo));  alter table ofrecimiento add constraint fk_ofre_matpen foreign key(cod_materia,cod_pensum) references materiapensum (cod_mat_mp,cod_pensum_mp); </pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_materia	Código de la materia a ofrecer
	cod_pensum	Código del pensum
	grupo	Grupo a ofrecer
	num_alumnos	Numero de alumnos que se prevén para el ofrecimiento
	aprobados	Alumnos que aprobaron al finalizar el periodo
	reprobados	Alumnos que reprobaron al finalizar el periodo
	jornada	Jornada del ofrecimiento
	estado	Estado del ofrecimiento
	fecha_inicializacion	Fecha de inicio
	fecha_finalizacion	Fecha de fin

<b>Tabla</b>	<b>docenteofrecimiento</b>	
<b>Descripción</b>	Ofrecimientos asignados a un docente	
<b>Script</b>	<pre> create table docenteofrecimiento( cod_docente varchar(12) not null references empleado(cod_empleado), cod_pensum varchar(12) not null, cod_materia varchar(12) not null, grupo numeric(1));  alter table docenteofrecimiento add constraint fk_docofre foreign key (cod_materia,cod_pensum,grupo) references ofrecimiento(cod_materia,cod_pensum,grupo); </pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_docente	Cédula del docente a quien se asignará el ofrecimiento
	cod_pensum	Código del pensum a ofrecer
	cod_materia	Código de la materia a ofrecer
	grupo	Grupo a ofrecer

<b>Tabla</b>	<b>recurso</b>	
<b>Descripción</b>	Contiene los posibles recursos para un aula o materia	
<b>Script</b>	<pre> create table recurso( cod_recurso numeric(2) primary key, descripcion varchar(40) not null </pre>	

	);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_recurso	Consecutivo del recurso
	descripcion	Descripción del recurso

<b>Tabla</b>	<b>recursomateria</b>	
<b>Descripción</b>	Asignación de recursos par a las materias	
<b>Script</b>	create table recursomateria( codigomateria varchar(12), recurso numeric(2), constraint pk_recmat primary key(codigomateria,recurso) );	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	codigomateria	Código de la materia
	recurso	Código del recurso

<b>Tabla</b>	<b>tipoaula</b>	
<b>Descripción</b>	Clasificación de las aulas	
<b>Script</b>	create table tipoaula( cod_tipoaula numeric(2) primary key, descripcion varchar(40) not null);	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_tipoaula	Consecutivo del tipo de aula
	descripcion	Descripción del tipo de aula

<b>Tabla</b>	<b>aula</b>	
<b>Descripción</b>	Aulas y laboratorios	
<b>Script</b>	create table aula( cod_aula varchar(7) primary key, capacidad varchar(3) not null, tipo numeric(2) references tipoaula(cod_tipoaula), zona varchar(2)references zona(cod_zona));	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_aula	Código del Aula
	capacidad	Capacidad del aula
	tipo	Tipo de aula
	zona	Zona donde se encuentra ubicada el aula

<b>Tabla</b>	<b>recursoaula</b>	
<b>Descripción</b>	Asignación de recursos a las aulas	
<b>Script</b>	create table recursoaula( cod_aula varchar(7) references aula(cod_aula), cod_recurso numeric(2) references recurso(cod_recurso), descripcion varchar(40) not null,	

	estado varchar(1) references estado(cod_estado), constraint pk_recaula primary key(cod_aula,cod_recurso));	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_aula	Código del aula
	cod_recurso	Recurso asociado al aula
	descripcion	Descripción del recurso
	estado	Estado del recurso

<b>Tabla</b>	<b>horario</b>	
<b>Descripción</b>	Asignación de horario para los ofrecimientos	
<b>Script</b>	<pre>create table horario(   cod_materia varchar(12),   cod_pensum varchar(12),   grupo numeric(1),   hora_inicio time,   hora_final time,   dia numeric(1) references dia(cod_dia),   cod_aula varchar(7),   constraint pk_horario primary   key(cod_materia,cod_pensum,   grupo,dia,hora_inicio));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_materia	Código de la materia
	cod_pensum	Código del pensum
	grupo	Grupo
	hora_inicio	Hora de inicio de la clase
	hora_final	Hora de finalización de la clase
	dia	Día de la clase
	cod_aula	Código del aula para la clase

<b>Tabla</b>	<b>horarioproxy</b>	
<b>Descripción</b>	Asignación de horario para las cargas adicionales	
<b>Script</b>	<pre>create table horarioproxy(   cod_docente varchar(12) not null references   empleado(cod_empleado),   adicional numeric(2) references   tipoadicionalcarga(cod_concepto_ti),   hora_inicio time,   hora_final time,   dia numeric(1) references dia(cod_dia),   cod_aula varchar(7));  alter table horario add constraint fk_horario foreign key(cod_materia,cod_pensum,grupo) references ofrecimiento(cod_materia,cod_pensum,grupo);</pre>	

<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_docente	Código del docente
	adicional	Código del tipo de carga adicional
	hora_inicio	Hora de inicio
	hora_final	Hora de finalización
	dia	Día
	cod_aula	Aula si se es necesaria

<b>Tabla</b>	<b>reserva_aula</b>	
<b>Descripción</b>	Fecha y horas en las cuales se requiere un aula para una actividad determinada, asociada a un empleado registrado.	
<b>Script</b>	<pre>create table reserva_aula(     hora_inicio time not null,     hora_final time not null,     fecha_prestamo date not null,     cod_aula varchar(7) not null references aula(cod_aula),     fecha_reserva date,     usuario varchar(12) references     empleado(cod_empleado));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	hora_inicio	Hora de inicio de la reserva
	hora_final	Hora de finalización de la reserva
	fecha_prestamo	Fecha en la que se realiza la reserva
	cod_aula	Aula a reservar
	fecha_reserva	Fecha en que se utilizará el aula
	usuario	Usuario que realiza la reserva

<b>Tabla</b>	<b>tabla</b>	
<b>Descripción</b>	Tablas que posee el sistema y dirección de su respectiva página web	
<b>Script</b>	<pre>create table tabla(     cod_tabla varchar(20) primary key,     descripcion varchar(40) not null,     url_tabla varchar(70));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_tabla	Tabla del sistema
	<u>descripcion</u>	Descripción de la tabla
	url_tabla	Dirección web asociada a la tabla

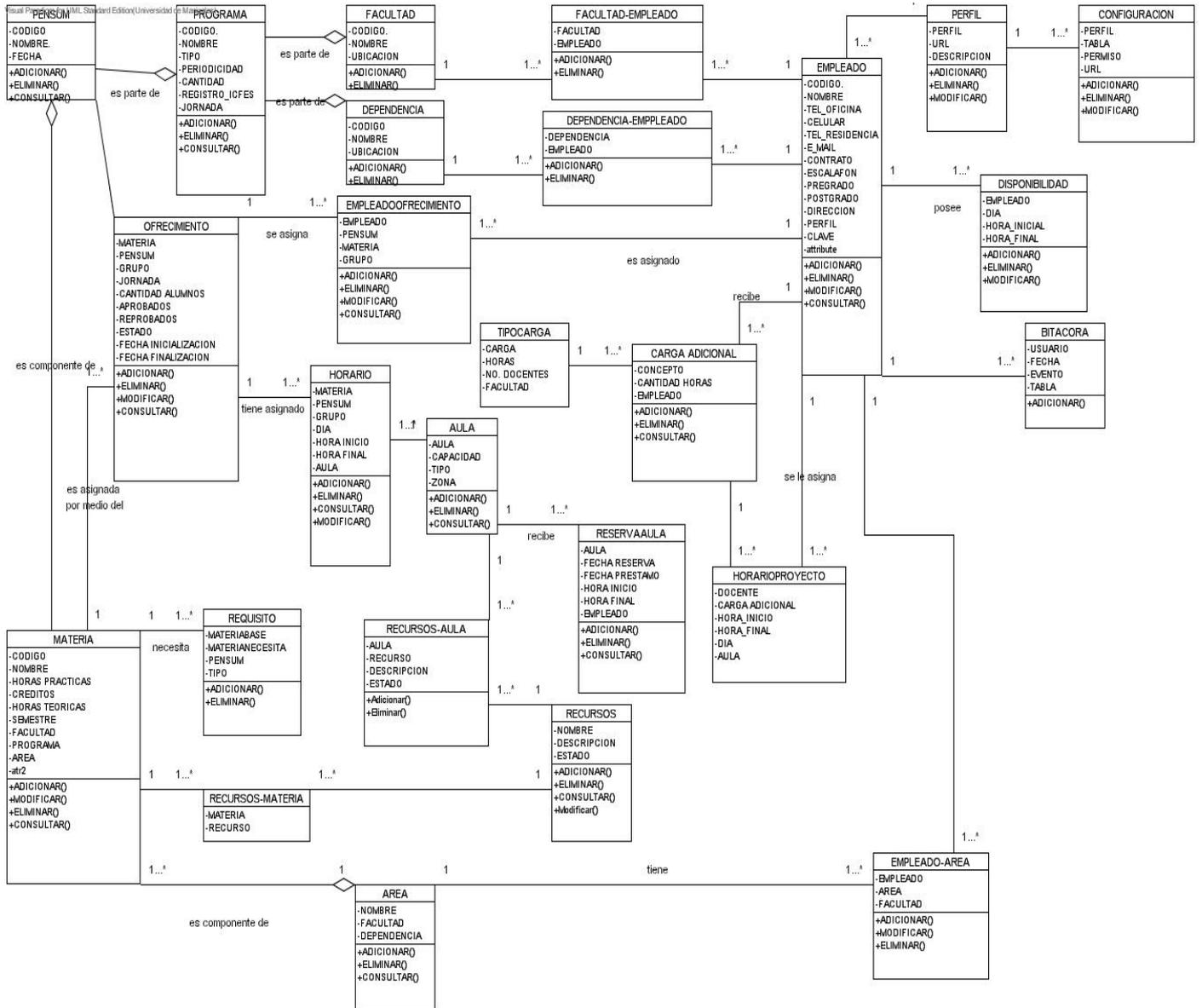
<b>Tabla</b>	<b>permiso</b>	
<b>Descripción</b>	Clasificación de los tipos de permiso	
<b>Script</b>	<pre>create table permiso(     cod_permiso varchar(1) primary key,     descripcion varchar(10) not null);</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_permiso	Consecutivo para los permisos
	descripcion	Descripción de los permisos

<b>Tabla</b>	<b>bitacora</b>	
<b>Descripción</b>	Contiene las acciones que realiza un usuario en el sistema	
<b>Script</b>	<pre>create table bitacora(     cod_usuario varchar(12) references     empleado(cod_empleado),     fecha timestamp,     accion varchar(10) not null,     tabla varchar(20) not null);</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_usuario	Usuario del sistema
	fecha	Fecha de la acción
	accion	Tipo de acción registrada
	tabla	Tabla afectada

<b>Tabla</b>	<b>disponibilidad</b>	
<b>Descripción</b>	Contiene los días y horas en los cuales un docente se puede dedicar a una actividad para la universidad.	
<b>Script</b>	<pre>create table disponibilidad(     cod_empleado varchar(12) references     empleado(cod_empleado),     dia numeric(1) references dia(cod_dia),     hora_inicial time,     hora_final time,     primary key(cod_empleado,dia,hora_inicial,hora_final));</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_empleado	Código del empleado
	dia	Día que esta disponible
	hora_inicial	Hora inicial del bloque de disponibilidad
	hora_final	Hora final del bloque de disponibilidad

<b>Tabla</b>	<b>opciontabla</b>	
<b>Descripción</b>	Contiene la relación entre el perfil, la tabla, los permisos y la web asociada	
<b>Script</b>	<pre>create table opciontabla (     cod_perfil varchar(12) references perfil(cod_perfil),     tabla varchar(20) references tabla(cod_tabla),     permiso varchar(1) references permiso (cod_permiso),     url_opcion varchar(90) not null);</pre>	
<b>Estructura</b>	<b>Campo</b>	<b>Descripción</b>
	cod_perfil	Código del perfil
	tabla	Código de la tabla
	permiso	Código del permiso
	url_opcion	Dirección Web asociada

MODELO DE OBJETOS



## DICCIONARIO DE CLASES Y ATRIBUTOS

Clase	Pensum	
<b>Descripción</b>	Plan estudiantil creado para los programas de cada facultad y está compuesto por un conjunto de asignaturas adecuadamente sistematizadas que el estudiante debe cursar y aprobar en el transcurso de su carrera.	
<b>Atributos</b>	codigo varchar(20) nombre varchar(80) fecha date	Código del pensum Nombre del pensum Fecha de creación
<b>Métodos</b>	Adicionar() Modificar() Consultar()	

Clase	programa	
<b>Descripción</b>	Programa Académico es el conjunto de actividades orientadas a la formación profesional en una determinada área del saber y a la obtención de un título profesional.	
<b>Atributos</b>	tipo varchar(2) nombre varchar(60) codigo varchar(2) periodicidad varchar(10) cantidad varchar(2) registro_icfes varchar(15) jornada varchar(6) hora_inicio_clases time	Tipo de programa académico Nombre del programa Código del programa Periodicidad del programa No. de niveles del programa Registro Icfes para el programa Jornada del programa Hora de inicio de clases para el programa
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

Clase	facultad	
<b>Descripción</b>	Sección de una universidad que engloba estudios de una misma rama.	
<b>Atributos</b>	codigo varchar(2) nombre varchar(40) ubicacion varchar (2)	Código de la facultad Nombre de la facultad Ubicación de la facultad
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>dependencia</b>	
<b>Descripción</b>	Sección de una universidad que engloba estudios de una misma rama.	
<b>Atributos</b>	codigo varchar(2) nombre varchar(40) ubicacion varchar (2)	Código de la Dependencia Nombre de la Dependencia Ubicación de la Dependencia
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadofacultad</b>	
<b>Descripción</b>	Docentes y empleados asociados a una facultad de la Universidad.	
<b>Atributos</b>	facultad varchar(2) empleado varchar(12)	Código facultad Cédula del empleado
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadodependencia</b>	
<b>Descripción</b>	Docentes y empleados asociados a una dependencia de la Universidad.	
<b>Atributos</b>	dependencia varchar(2) empleado varchar(12)	Código Dependencia Cédula del empleado
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleado</b>	
<b>Descripción</b>	Personas que hacen parte del sistema, con su respectiva información personal.	
<b>Atributos</b>	codigo varchar(12) nommbre varchar(40) tel_oficina varchar(10) celular varchar(10) tel_residencia varchar(10) email varchar(30) contrato varchar(2) escalafon integer pregrado varchar(40) postgrado varchar(40) direccion varchar(40)	Cédula del empleado Nombre Teléfono de la oficina Celular Teléfono de la residencia Correo Electrónico Tipo de contrato laboral Tipo de escalafón docente Pregrado Postgrado o Especialización Postgrado o Especialización Tipo de perfil asociado

	perfil varchar(1)	
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	bitacora	
<b>Descripción</b>	Información histórica de control de acciones ejecutadas sobre las tablas del sistema por cualquier usuario.	
<b>Atributos</b>	usuario varchar(12) fecha timestamp evento varchar(10) tabla varchar(20)	Cédula del empleado Fecha de la acción Acción realizada Tabla del sistema afectada
<b>Métodos</b>	Adicionar() Consultar()	

<b>Clase</b>	<b>contrato</b>	
<b>Descripción</b>	Tipos de contrato que se pueden presentar para un docente según el no. de horas.	
<b>Atributos</b>	contrato varchar(3) nombre varchar(70) horas numeric(3)	Tipo del contrato Nombre del tipo de contrato No. de horas mínimas para el tipo de contrato
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>cargaadicional</b>	
<b>Descripción</b>	Asignación de carga académica adicional a un docente, en una cantidad de horas determinada.	
<b>Atributos</b>	concepto numeric(2) empleado varchar(12) cantidad horas numeric(3)	Código tipo de carga adicional Cédula del empleado Cantidad de horas para el proyecto
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>tipocarga</b>	
<b>Descripción</b>	Corresponde a las cargas académicas adicionales que no hacen parte de la orientación de clases como son los proyectos, procesos administrativos, investigación, planeación, entre otros.	
<b>Atributos</b>	carga varchar(40)	Carga adicional

	no docentes numeric(2) facultad varchar(2)	No de docente para la carga adicional Facultad donde se asignará la carga adicional
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>perfil</b>	
<b>Descripción</b>	Corresponde a los grupos identificados de usuarios que acceden al sistema con unos permisos comunes dentro del perfil.	
<b>Atributos</b>	perfil varchar(12) Descripción varchar(40) url varchar(40)	Código del perfil Descripción de perfil Url home asociada al perfil
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadoarea</b>	
<b>Descripción</b>	Docentes que poseen habilidades para un área del conocimiento que agrupa varias materias.	
<b>Atributos</b>	empleado varchar(12) area numeric(2) facultad varchar(2)	Cédula del empleado Area a la que se vincula el empleado Facultad donde se realiza la vinculación
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	
<b>Clase</b>	<b>area</b>	
<b>Descripción</b>	Agrupación de materias según una determinada área del conocimiento.	
<b>Atributos</b>	nombre varchar(40) facultad varchar(2) dependencia varchar(2)	Nombre el área Facultad del área (si es facultad) Dependencia del área (si es dependencia)
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>materia</b>
<b>Descripción</b>	Asignatura que hace parte de un pensum dentro de un

	programa y asociado a la facultad.	
<b>Atributos</b>	cod_materia varchar(12) nombre varchar(20) horas practicas numeric(2) credits varchar(1)  horas teoricas numeric(2) semestre varchar(2) facultad varchar(2), programa varchar(2) references programa(cod_prog), area numeric(2)	Código de la materia Nombre de la materia No. de horas prácticas No. mínimo de créditos para aprobar materia No. de horas teóricas Semestre en que se dicta la materia Facultad asociada a la materia Programa asociada a la materia  Area asociada a la materia
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>ofrecimiento</b>	
<b>Descripción</b>	Es la materia con un grupo y pensum asociados, para ser orientada a los estudiantes.	
<b>Atributos</b>	materia varchar(12) pensum varchar(12) grupo numeric(1) jornada varchar(1) cantidad alumnos numeric(2) aprobados numeric(2)  reprobados numeric(2)  estado varchar(9) fecha_inicializacion date fecha_finalizacion date	Código de la materia Código del pensum Grupo del ofrecimiento Jornada No. de alumnos  Alumnos que aprueban al finalizar el semestre Alumnos que reprueban al finalizar el semestre Estado del ofrecimiento (Activo Inactivo) Fecha de inicio de clases Fecha de finalización de clases
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>horario</b>	
<b>Descripción</b>	Corresponde al ofrecimiento que se orientará a los estudiantes, con una hora inicial y hora final en un día determinado de la semana y con aula asociada.	
<b>Atributos</b>	materia varchar pensum varchar	Código de la materia Código del pensum

	grupo numeric dia numeric(1) hora_inicio time hora_final time aula varchar(7)	Grupo de ofrecimiento Dia de la semana Hora Inicio Hora Final Aula para el horario
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>requisito</b>	
<b>Descripción</b>	Relación existente entre las materias de un pensum, la cual puede ser: Prerrequisito: una asignatura es el prerrequisito de una segunda cuando para poder formalizar la matrícula de la segunda es necesario tener aprobada la primera. Correquisito: una asignatura es correquisito de una segunda cuando para poder aprobar la segunda es necesario matricularse también de la primera o tenerla aprobada	
<b>Atributos</b>	materiabase varchar(12) , materianecesita varchar(12) pensum varchar(20), tipo varchar(12) not null	Código Materia base Código Materia analizada Código del pensum Tipo de requisito
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>empleadooofrecimiento</b>	
<b>Descripción</b>	Relación en la cual el docente orientará un determinado ofrecimiento.	
<b>Atributos</b>	docente varchar(12) pensum varchar(12) materia varchar(12) grupo numeric(1)	Cédula del docente Código del pensum Código de la materia No. de grupo
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>disponibilidad</b>	
<b>Descripción</b>	Horas se la semana por día, en el cual el docente se encuentra disponible para la orientación de una materia y/o para la asignación de una carga adicional.	
<b>Atributos</b>	empleado varchar(12)	Cédula del empleado

	dia numeric(1) hora_inicial time hora_final time	Código del día Hora inicio de bloque disponible Hora final de bloque disponible
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>aula</b>	
<b>Descripción</b>	Lugar registrado por Planeación para la orientación de una materia.	
<b>Atributos</b>	aula varchar(7) capacidad varchar(3) tipo numeric(2) zona varchar(2)	Código del aula Capacidad del aula Tipo de aula Zona del aula
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>recursoaula</b>	
<b>Descripción</b>	Recurso físico que se encuentra dentro del aula y su estado asociado.	
<b>Atributos</b>	aula varchar(7) recurso numeric(2) Descripción varchar(40) estado varchar(1)	Código del aula Recurso del recurso Descripción del recurso Estado del recurso
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>recursomateria</b>	
<b>Descripción</b>	Recursos requeridos por la materia para su adecuada orientación.	
<b>Atributos</b>	materia varchar(12) recurso numeric(2)	Código de la materia Código del recurso asociado a la materia
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>horarioproyecto</b>	
<b>Descripción</b>	Días y horas asociadas a un proyecto o carga adicional por docente registrado.	
<b>Atributos</b>	docente varchar(12) adicional numeric(2) hora_inicio time hora_final time día numeric(1) cod_aula varchar(7)	Cédula del docente Código del proyecto Hora inicio asociado al día Hora final asociado al día Día de asignación del proyecto Código del aula para la carga adicional (si es necesario)
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

<b>Clase</b>	<b>reservaaula</b>	
<b>Descripción</b>	Fecha y horas en las cuales se requiere un aula para una actividad determinada, asociada a un empleado registrado.	
<b>Atributos</b>	cod_aula varchar(7) fecha_reserva date fecha_prestamo date hora_inicio time hora_final time empleado varchar(12)	Código del aula Fecha en que se utilizará el aula Fecha en que se reserva el aula Hora inicio de la reserva Hora final de la reserva Cédula del empleado que reserva
<b>Métodos</b>	Adicionar() Modificar() Eliminar() Consultar()	

## DESCRIPCIÓN DE PROCESOS EN LA ASIGNACIÓN DE HORARIOS METODOLOGÍA, TÉCNICAS DE CRUCES Y BLOQUEOS

Para la asignación de horarios a los ofrecimientos, se tienen 3 procesos generales que intervienen para consulta, inserción y borrado, pero que son analizados independientemente para su clara interpretación. Es importante tener en cuenta para el desarrollo web, se realiza una consulta inicial de información de la base de datos, los datos son llevados a matrices y al finalizar el proceso son almacenados en la base de datos, teniendo en cuenta que en este proceso de asignación se llevan a cabo unos bloqueos que impiden la alteración de los datos fuentes mientras la información se encuentra en memoria volátil. De igual forma se debe tener en cuenta que al realizar modificación en los datos, se realiza un refresco en la página web y en la matrices de información, lo cual implica modificaciones instantáneas.

### PROCESOS GENERALES DEL HORARIO

#### CONSULTA

1. Consultar los ofrecimientos por facultad, programa, tipo de periodo (año, semestre, trimestre entre otros) y grupo
2. Consultar los docentes por materia por cada ofrecimiento de la consulta anterior que pueden orientar esta materia según la relación área docente.
3. Cargar en memoria por cada docente que ha sido previamente seleccionado su disponibilidad según día y hora.

#### MODIFICACIÓN

##### **Caso 1. Cambio en el Docente:**

Se libera este docente del ofrecimiento  
Se adicionan las horas del docente liberado en su disponibilidad  
Se eliminan los bloqueos a las otras materias que pueden ser orientadas el mismo día en el horario que se acaba de liberar.  
(Un grupo no puede ver más de una materia en igual horario)  
Se adicionan las horas que se acaban de liberar en el total de horas semanales por materia.  
Se asigna el nuevo docente  
Se carga la disponibilidad de horas por días, teniendo en cuenta los bloqueos generados por el horario de otras materias.

##### **Caso 2. Cambio de hora inicial en un día determinado**

Se realiza la liberación de la cantidad de horas que fue asignada anteriormente para esta materia en el día indicado.  
Se devuelven las horas que habían sido asignadas para esta materia a la disponibilidad del docente según día y cantidad de horas  
Se devuelve el número de horas asignadas al total de horas disponibles para orientar la materia

Se libera el bloqueo de horas según el día para las demás materias del grupo.

**Caso 3: Cambio en la cantidad de horas de asignación para una materia**

Se verifica que las horas permitidas para asignación no se crucen con otros

Se verifica que el acumulado de horas por materia no exceda el total de horas por semana de la materia

Se realiza la disminución o aumento en las horas de la disponibilidad del docente, así como la disminución o aumento en el total de horas disponibles para la materia.

Se realiza un bloqueo según la hora inicial y el número de horas para las demás materias que pueden ser orientadas el mismo día, pero no en las mismas horas ya asignadas.

**ELIMINACIÓN**

Corresponde a un caso especial de la modificación, en el cual la hora inicial se deja en vacío y se procede a liberar la disponibilidad del docente, la cantidad de horas totales de la materia y los bloqueos al resto de materias que pueden ser orientados en el mismo día.

Los bloqueos son reconstruidos cada que se cambie la hora inicial, la cantidad de horas y el docente.

## ANEXO R

### 11.4.8 MODELO DE ENTREVISTA PARA EL ACERCAMIENTO AL MODELO DE CARGA ACADÉMICA EN LA UNIVERSIDAD DE MANIZALES.

Desarrollado por David Agudelo & Juan Carlos Cortes  
Proceso de Análisis y diseño para el desarrollo de Cronos, Sistema de Carga Académica para la Universidad  
de Manizales

Nombre: \_\_\_\_\_

Cargo: \_\_\_\_\_

Facultad \_\_\_ Dependencia \_\_\_

Fecha Entrevista (dd/mm/aaaa) : \_\_\_ / \_\_\_ / \_\_\_\_\_

#### PREGUNTAS PARA DEPENDENCIAS:

##### 1- Carácter de la dependencia:

---

---

---

---

---

##### 2-Tiene algún vinculo con la asignación de carga académica de cada facultad?

En caso de ser afirmativo: Cuales son?

---

---

---

##### Descripción de cada uno de ellos

---

---

---

---

---

**3- En los procesos de creación de horarios y agendas de cada facultad de que manera participa en ellos?**

---

---

---

**4- Esta dependencia Se vincula a los procesos de asignación, préstamo o distribución de aulas para los diferentes usos académicos de la universidad, cuales son y de que manera se vinculan a estos procesos?**

---

---

---

---

---

**5- Esta dependencia Se vincula a los procesos de asignación, préstamo o distribución de recursos académicos para los diferentes usos académicos de la universidad, cuales son y de que manera se vinculan a estos procesos?**

---

---

---

---

---

**6-Solo si respondió de una manera afirmativa la pregunta 4 :  
Bajo que criterios y parámetros puede una facultad u otra dependencia acceder a estos recursos mencionados en la pregunta 4**

---

---

---

---

**7-Solo si respondió de una manera afirmativa la pregunta 5 :  
Bajo que criterios y parámetros puede una facultad u otra dependencia acceder a  
estos recursos mencionados en la pregunta 5**

---

---

---

---

**8- En los procesos académicos que usted se vincula con las diferentes facultades(carga académica y recursos), que tipo de organización, control y planeación maneja para su administración.**

- a) Manual
- b) Hojas de Calculo
- c) Procesadores de Texto
- d) Software específico de administración
- e) Todas las anteriores
- f) Otra diferente, Descripción

---

---

---

---

**Espacio para acotaciones propias del entrevistado como parámetros de tiempos, entregas o casos especiales.**

---

---

---

---

**9- Puede describirnos según la respuesta de la pregunta ocho(8) el proceso que usted lleva y la descripción de la forma de trabajo con la cual usted participa en estos procesos académicos antes mencionados.**

*NOTA: Solo se llena la descripción de los procesos que hace parte.  
Las respuestas pueden ser complementadas desde el punto de vista de los desarrolladores del proyecto*

Pregunta 2

---

---

---

---

Pregunta 3

---

---

---

---

Pregunta 4

---

---

---

---

---

Pregunta 5

---

---

---

---

---

**10- Existe algún otro rol o cargo diferente al suyo en esta dependencia, cual es el nivel sobre usted y que papel y permisos maneja dentro de los procesos de esta dependencia.**

**Descripción de Rol Uno**

---

---

---

---

---

.  
.  
.

**Descripción de Rol N**

---

---

---

---

---

**11-Que sugerencias tendría usted para el desarrollo de un sistema que integre sus necesidades, o pueda mejorar los procesos que usted actualmente lleva?**

---

---

---

---

---

**OBSERVACIONES GENERALES POR PARTE DE LOS ANALISTAS**

---

---

---

---

---

---

---

---

---

---

## **PREGUNTAS PARA FACULTADES:**

**1- Como se lleva actualmente la asignación de carga académica:**

- a) Manual
- b) Hojas de Calculo
- c) Procesadores de Texto
- d) Software Especifico

**2- Cual es el tiempo de duración Total o aproximado si es el caso, para asignar la carga académica?**

---

**3-Hay algún momento en que el proceso de asignación se vuelva complejo y dispendioso, Cual?**

---

---

**Razón:**

---

---

---

---

**4-Existe algún rol o usuario diferente a usted que pueda acompañar o desarrollar solo el proceso de asignación de carga?, cuales son?**

---

---

---

---

---

**5- La Universidad cuenta con tres tipos bases de contratación: Tiempo Completo(TC), Medio Tiempo(MT), Catedrático(C) y otros como docente especial(DE) y foráneos(F), como es el proceso de asignación de la carga académica para cada uno de estos tipos:**

**Para TC:**

---

---

---

---

**Para MT:**

---

---

---

---

**Para C:**

---

---

---

---

**Para DE:**

---

---

---

---

**Para F:**

---

---

---

---

**6- Como se maneja la disponibilidad de cada docente o empleado según su carga a la hora de asignar las cargas académicas?**

---

---

---

**7 Además de las clases(Ofrecimientos), existe algún otro tipo de carga académica adicional a esta?, cuales son?**

---

---

---

---

**8- Los proyectos nuevos que van existiendo en una facultad se clasifican en base a las cargas académicas anteriores?**

---

---

---

---

**9- Cuales son los parámetros y características de una carga académica adicional?**

---

---

---

---

---

**10- Además de las clases(ofrecimientos) que se le asignan a los empleados, estos pueden participar en algún otro tipo de carga académica diferente a este.?**

---

---

---

**11-Con respecto a los horarios, existe algún modelo o sistema que usted use de los mencionados a continuación a la hora de planearlos y crearlos?**

- a) Manual
- b) Hojas de Calculo
- c) Procesadores de Texto
- d) Software Especifico
- e) otro: \_\_\_\_\_

**12-Cual es el proceso y como puede describirnos el modelo usado?**

---

---

---

---

---

**13- En la actualidad que es lo mas difícil de hacer en un horario, y que tiempo puede tardar en concluir el proceso de creación?**

---

---

---

**14- Si existiera un nuevo sistema que le ayude a mejorar alguno o todos los procesos que usted lleva actualmente, que modificaciones y/o mejoras le haría?**

---

---

---

---

**15-Ustedes llevan horarios de clase para los docentes y los ofrecimientos, ¿también llevan horarios del docente en cargas adicionales?**

**16- En los procesos académicos que usted se vincula con las diferentes facultades y dependencias respecto a solicitud de recursos y reserva de aulas como lo hace?**

- A)Consultas Telefónicas
- B)Consulta Software centralizado
- C)Oficina de Planeación
- D)Audiovisuales
- E) Otro, Cual?

---

---

---

**17- Maneja algun tipo de reporte especifico o general para uso propio, o para alguna dependencia o facultad especifica que se lo solicite?**

**Descripción Uno**

**Uso Uno:**

---

**Descripcion Uno:**

---

---

---

---

**Descripción N**

**Uso N:**

---

**Descripcion N:**

---

---

---

---

**18- Respecto a los pensums y materias, Cual es la forma de creación y modificación de Un pensum y una materia, además de su facultad existe otra dependencia que intervenga en el proceso?**

---

---

---

---

---

**19 Si se desarrollara un único sistema integrado, que reúna los procesos y necesidades de una asignación de carga académica, de la creación de horarios y de reserva de recursos y aulas, en que lo mejoraría con respecto al que usted lleva actualmente y respecto a los que usted usa de otras dependencias?**

---

---

---

**20- Existe algún otro rol o cargo diferente al suyo en esta facultad, cual es el nivel sobre usted y que papel y permisos maneja dentro de los procesos de esta facultad**

**Descripción de Rol Uno**

---

---

---

---

---

.  
. .  
.

**Descripción de Rol N**

---

---

---

---

---

**21- Manejan algún tipo de aula especial en su facultad, que no dependa dela administración central?**

**OBSERVACIONES GENERALES POR PARTE DE LOS ANALISTAS**

---

---

---

---

---

---

---

---

---

---

**ANEXO S**  
**SISTEMAS ACTUALES Y ENTREVISTAS A LOS MODELOS**  
**TRADICIONALES DE CARGA ACADÉMICA Y PLANEACIÓN DE LA**  
**UNIVERSIDAD DE MANIZALES**

Se presenta a continuación los primeros acercamientos a las formas tradicionales en que se maneja la asignación de carga académica, horarios y aulas de las Facultades y dependencias de la Universidad de Manizales. También algunas sugerencias y aportes para una futura solución informática por parte de los entrevistados y desarrolladores del proyecto CRONOS.

## **SISTEMA DE LA OFICINA DE PLANEACIÓN**

**Plataforma: Windows**

**Desarrollo: FoxPro**

Este sistema maneja programas, profesores, materias, ofrecimientos aulas. Horarios en forma manual, semi-automática y automática, pero de las cuales solo servía la primera "manual". Este sistema al tratar de hacer una creación de horarios semi-automática presentaba fallas hasta el punto en que ya no respondía, pero como conclusión general ninguno de los tipos de asignación de horarios era utilizado por los administradores del sistema.

El sistema de planeación permitía la administración de aulas con atributos específicos como ubicación, día y horas semanales de disponibilidad, capacidad y consulta. Reportes de horarios por edificio, profesor y programa.

### **ENTREVISTAS A DECANOS, SECRETARIOS ACADÉMICOS, SECRETARIAS, DOCENTES Y ADMINISTRATIVOS**

Se hicieron entrevistas a cada Facultad y dependencias de la Universidad que forman parte del problema específico de esta tesis, de las cuales se rescatan algunas charlas y preguntas para detectar las necesidades primarias de un nuevo sistema y detectar también las formas actuales en que se lleva la planeación y desarrollo de una carga académica

### **COMUNICACIÓN SOCIAL ABRIL DE 2004**

Asignación de carga académica: Se basa en la experiencia, conocimiento y prioridad del docente que orientara algún ofrecimiento o tarea especial(carga adicional).también basado en el reglamento y tipo de vinculación: Tiempo Completo(TC), Medio Tiempo (MT), Cátedra(C), docente especial(DE).

Dificultades: las mayores dificultades según expuestas por el decano se presentan en no haber un sistema que satisfaga las necesidades, no este al alcance de todos. También menciona que además de este problema de aplicación o técnico el sistema de carga se hace complejo a la hora de tratar con empleados de tipo Catedrático por su disponibilidad y contrato en la Universidad.

Componentes de la carga académica: Los componentes son basados en ofrecimientos y tareas externas que se dividen en actividades académicas, proyectos, cátedra e investigación

Horarios: básicamente el horario se trabaja en base de la complejidad de disponibilidad de docentes catedráticos, también se necesita un sistema de clasificación de aulas como estudio de televisión, radio, clases normales. Además de tener aulas privadas y específicas por Facultad

## **FACULTAD DE CONTADURÍA**

Asignación de carga académica: es basada en los tipos de vinculación TC, MT, C etc, solo que se asignan las clases dependiendo el semestre y la disponibilidad dando un poco de comodidad a los catedráticos:

**MT:** solo dan clases en las noches

**TC:** Día y noche, no hay algún tipo de conciliación para su disponibilidad

**C:** algunos de día y otros de noche, pero la mayoría accede es en horas de la noche.

La carga académica se basa en la visualización de los horarios de semestres inmediatamente anteriores.

Se asigna un tipo de vinculación, se asignan las horas semanales por ofrecimiento y las horas que le sobran en el contrato son asignadas a otras actividades

Horarios: se desarrolla de una manera muy radical desde el punto de vista de un contrato de cátedra. No hay maneras claras y visibles de informes que permitan dar una alternativa al catedrático mas haya del ofrecimiento que se le hace (la hora de la clase que fue asignada a la materia que va a orientar y ya).

Si hay algún cambio extraordinario en el horario toca hacer un re-planteamiento a lápiz para su posterior digitalización de formatos de hojas de calculo como Excel de Microsoft, lo cual se vuelve muy dispendioso para evitar cruces de horarios y asignación de catedráticos.

**NOTA ESPECIAL:** el programa de contaduría cambia de horario diurno a nocturno en los últimos 3 semestres(8, 9, 10). Lo que implica un diseño especial de planeación para estos últimos semestres en base a horarios, ofrecimientos y vinculación. Esto también no se refleja en el sistema de planeación (software descrito en los primeros párrafos de este documento) lo cual devela que no hay un sistema que atienda efectivamente a este tipo de casos a la hora de asignar aulas a los horarios de la Universidad, afectando a cada Facultad y dependencia que necesite de aulas para sus labores normales.

**¿Manejan Aulas especiales?:** Se manejan dos aulas(315 y 316) informáticas exclusiva de la Facultad pero asignadas y administradas por la Facultad de Contaduría y la Facultad de Ingeniería.

La Facultad de derecho solo realiza una vez al año la asignación de carga académica, lo que hace que por ejemplo las reservas de aulas de clase para esta Facultad queden registradas durante todo el año igual que la disponibilidad de los docentes.

## **FACULTAD DE PSICOLOGÍA**

La característica especial que diferencia a esta Facultad del resto de las otras es que durante los primeros 6 semestres no existen pre-requisitos en las materias, lo que significaría especial para este proyecto seria el momento de crear los grupos y los horarios de clase.

## **FACULTAD DE MEDICINA.**

Se trabajaría normalmente como el común de las Facultades asignando las cargas a partir de sus contratos(MT, TC, CAT). La diferencia en esta radica en que los estudiantes de sexto semestre en adelante no tiene que venir a clases al interior de la Universidad o alguna de sus cedes, pues se encuentran en las practicas y laboratorios clínicos y de salud de la ciudad. Lo cual beneficiaria el sistema de asignación de aulas para ofrecimientos de una manera tal que entraría en un modelo de validación y control suave en el momento de hacer los horarios con sus respectivas aulas. Es de recordar que hasta nuestros días 2006 esta Facultad es la de mayor numero de estudiantes lo cual generaría un descanso al sistema en su momento de asignar aulas a las diferentes Facultades y programas.

## **FACULTAD DE INGENIERÍA**

La Facultad de Ingeniería junto con la Facultad de Contaduría son las Facultades mas organizadas y presentan un mejor modelo de reportes(maestros de detalle) manejados por la Universidad. Los modelos y formatos de estas Facultades son tomados de ejemplo en otras no tan bien alimentadas como Psicología, educación y economía.

En resumen el sistema de asignación de carga académica se basa en su tipo de contratación TC, MT, CAT para la distribución de ofrecimientos(entendida por este las clases) y cargas adicionales. Para su posterior asignación de aulas y horarios, caracterizados por diferentes roles de usuario que puedan acceder a determinados niveles de la aplicación para su administración, manipulación y control.

También es necesario resaltar que el desarrollo de un buen sistema y de los modelos actuales de asignación de carga de las diferentes Facultades se ve limitada por los contratos de tipo catedrático, también es necesario entre otros de los muchos aspectos generar reportes claros y concretos sobre las labores académicas de los docentes diferentes a los ofrecimientos, un sistema de

administración de proyectos, aulas y recursos, como valores agregados a este proyecto de grado.

Una vez iniciado la segunda parte de análisis y diseño del sistema se espera lograr llegar a un sistema único, centralizado y estándar que permita de una manera amable y clara para el usuario la asignación de carga académica, horarios y aulas de la Universidad de Manizales. Que pueda ser accedida desde cualquier PC dentro o fuera de la Universidad vía Web.

## REFERENCIAS DE PRUEBAS DE CALIDAD DE SOFTWARE POR PARTE DE USUARIOS UNA VEZ FINALIZADO EL PROYECTO

Las pruebas al desarrollo del sistema se han venido trabajando desde el segundo semestre de 2005 hasta la fecha siendo acompañadas por Usuarios que entrarían a usar el sistema y juegan un rol importante. Algunos de ellos son el Dr Henry Duque Secretario Académico de la facultad de Contaduría, El Ingeniero Carlos Betancourt Correa, Ex Secretario Académico de La Facultad de Ingeniería y ahora Docente, El profesor Fernando López de La Universidad Cooperativa de Colombia participando como Usuario Invitado, y algunos estudiantes de diferentes facultades quienes durante el transcurso del desarrollo del proyecto, nos han venido presentando algunas recomendaciones las cuales se han tenido en cuenta y se han desarrollado a cabalidad.

***También debe comentarse que las pruebas que se hicieron con los usuarios antes y posteriormente citados en este documento fueron pruebas de acuerdo al perfil que cumplen como: Invitados, Decanos, Secretarios Académicos, Docentes, Estudiantes.***

Finalizando El segundo semestre de 2005 se recibieron las recomendaciones por parte del Jurado calificador del proyecto(Ing Wilson Lopez, Ing Diego Samir, Ing José Fernando Mejía) donde presentaban inquietudes y propusieron algunos cambios que fueron abordados y solucionados a cabalidad por parte de los tesisistas, estos cambios fueron:

- Desarrollar un sistema que pueda involucrar diferentes gestores de bases de datos diferentes al del proyecto(PostgreSQL) tales como ORACLE y MYSQL
- Incorporar un modelo que permita hacer backups desde la web
- Modificar los sistemas de validación de campos de petición de datos en la web
- Que la aplicación pueda ser visualizada en cualquier navegador y Sistema operativo.
- Presentar un catalogo de fuentes que permita abstraer en un documento la función específica de cada web(función de la web)

En resumen General, además de las Recomendaciones se ha logrado tener un visto bueno favorable sobre la aplicación y que refleja comodidad en el diseño de la Interfaz con la cual interactúa el usuario y otras bondades operativas que presentan novedad y mayor utilidad sobre los sistemas descritos en los Antecedentes del proyecto y los sistemas de la Universidad.

Los últimos usuarios que participaron en una crítica en el año 2006 en el mes de Mayo fueron El Dr Richard Millán de la Facultad de Comunicación social y periodismo quien mostró especial agrado por la organización, control y reserva de aulas de clase, en especial sobre su clasificación, ya que este es encargado de uno de los centros mas importantes de la facultad como lo es el estudio de Televisión.

El segundo y ultimo usuario que presento el visto bueno del proyecto y quien lo observo a cabalidad fue el Decano de la Facultad de Ingeniería Carlos Alberto Cortes.

## **ANEXO T CATALOGO DE FUENTES**

### **DOCUMENTACIÓN DE PAGINAS WEB**

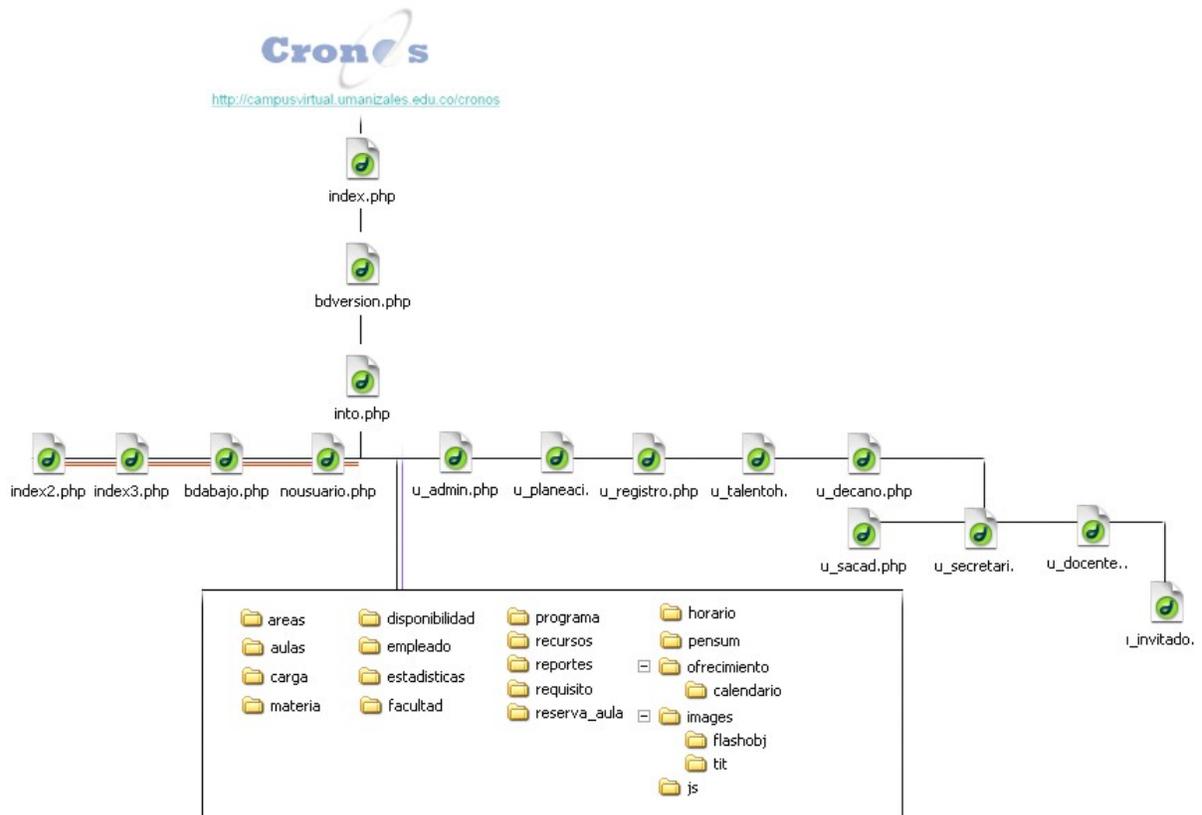
*SISTEMA CRONOS. ASIGNACIÓN DE CARGA ACADÉMICA Y PLANEACIÓN  
DE HORARIOS*



## ÍNDICE

<b>Mapa del Sitio .....</b>	<b>230</b>
<b>modelo de documentación e indentación de fuentes .....</b>	<b>231</b>
<b>Archivos de raíz del sitio.....</b>	<b>232</b>
<b>Interfaces de entrada inicial IU-Homes .....</b>	<b>233</b>
<b>Descripción de contenedores del sitio .....</b>	<b>235</b>
<b>Secuencia de entrada al sistema cronos .....</b>	<b>237</b>
<b>Como se construye una web cronos, bajo construcción dinámica? .....</b>	<b>238</b>
<b>Descripción de subcontenedores y archivos .....</b>	<b>239</b>

El sistema de planeación cronos, todo su modelo web se presenta bajo el concepto de construcción dinámica lo que quiere decir que sus acciones, su interfaz, sus menús de navegación y operación son construidos directamente desde la base de datos.



## MODELO DE DOCUMENTACIÓN E IDENTACIÓN DE FUENTES

```
<?php
    session_start();
    if({$username !='YZ'} and {$usertype !='7'})
    {

        /*
        |-----|
        | FOR: David Marcelo Agudelo & Juan Carlos Cortes |
        | Sistema de planeación de carga académica CRONOS 2004-2006. |
        | Universidad de Manizales. Facultad de Ingeniería |
        |-----|
        | DESCRIPCION DE LA WEB |
        | Web que permite crear las áreas a los diferentes programas, existe un caso |
        | especial en el momento de la creación del área: si el sistema reconoce que |
        | se trata de la facultad de contaduría, crea una casilla de verificación |
        | especial llamada hcomponente que le permitirá identificar si es área o |
        | componente, de acuerdo a la resolución 3459 de diciembre 30 de 2003 la cual |
        | inscribe las materias dentro de áreas o componentes para este tipo de |
        | facultades. |
        |-----|
        */

    }

    ?>
    <?

function listfacultad()
( //include ("../config/conexion.php");

    $sql="select cod_fac, nom_fac from facultad where cod_fac <>'99'and cod_fac <> '98'";
    $result2=Conectarse($sql,$versionbd);

    if (DB::isError ($result2))
    {
        //es cuando la tabla no existe
        die ("Falla de Consulta: " . $result2->getMessage () . "\n");
    }

    echo "<select class='combo' id='select_0' name='nfacultad' onChange='cargaContenido()'>";
    echo "<option value=''>Seleccione</option>";
    while ($row =< $result2->fetchRow ())
    {
        echo "<option value='".$row[0]."'>".$row[1]."</option>";
    }
    echo "</select>";
    $result2->free ();
}
?>

<HTML>
<HEAD>
<TITLE>CRONOS:&gt; Decanos</TITLE><!-- Creation date: 10/19/2002 -->
<META http-equiv=Content-Type content="text/html; charset=iso-8859-1">
<META content="MSHTML 6.00.2600.1106 name=GENERATOR<LINK href="../images/pref.css" type=text/css rel=StyleSheet
</HEAD>
```

## ARCHIVOS DE LA RAÍZ DEL SITIO

### **Index.php**

Web que redirecciona a otra url para que esta ultima se abra en pantalla completa e iniciar proceso de seguridad en la aplicación.

### **Indexe.php**

Web que se presenta en tamaño completo de pantalla, aquí el usuario ingresa su login(usuario), su clave, y la base de datos(periodo académico) al cual desea acceder, también acá se registran las variables globales para la ejecución normal de la aplicación web.

### **Index2.php**

Web que le informa al usuario que su primer acceso tuvo errores de digitación en su clave o en el usuario, y le permite volver a reintentar.

### **Index3.php**

Web que le informa al usuario que esta intentando acceder de una forma ilegal, el contenido presente se muestra de una manera bloqueada.

### **Enter.php**

Esta web se encarga de verificar los datos de entrada por parte del usuario y redireccionar a otra web dependiendo el caso: Acceso no valido por error de digitación(index2.php), Acceso denegado en caso de posible inyección de código maligno pruebas a través de MD5 (index3.php) o acceso exitoso(urlppal), en esta ultima se determina el perfil de usuario y se redirecciona a la web respectiva.

### **Bdabajo.php**

Web que informa al usuario que la conexión con el servidor de base de datos se ha perdido, se activa en el momento que la web o el usuario ejecuta alguna consulta.

### **Nousuario.php**

Web que se activa en el momento que un usuario desea acceder a una web que no es permitida para el perfil que maneja, o que conoce la url directa de una web y no a registrado su entrada en el sistema(también puede ser un posible ataque).

### **Políticas.php**

Web de presentación del sistema cronos, el cual muestra ayudas, descripciones generales, patente y avisos legales para su uso.

## **INTERFACES DE ENTRADA INICIAL DE USUARIOS Y PERFILES, TAMBIÉN LLAMADAS HOME DE USUARIO**

**Ubicación:** /

### **U\_admin.php**

Web principal de usuario administrador o súper usuario, desde esta web se tiene acceso a todos los subsistemas y acciones de cronos. Ver manual del usuario en la opción perfiles de usuario, perfil administrador.

### **decano.php**

Web principal de usuario decano, desde esta web se tiene acceso a todos los subsistemas y acciones de cronos que se encuentran bajo el perfil de decano. Ver manual del usuario en la opción perfiles de usuario, perfil decano.

### **U\_docente.php**

Web principal de usuario docente, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil de docente. Ver manual del usuario en la opción perfiles de usuario, perfil docente.

### **U\_invitado.php**

Web principal de usuario invitado, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil de invitado, el cual puede ser un visitante externo a la Universidad o estudiantes de la misma. Ver manual del usuario en la opción perfiles de usuario, perfil invitado.

### **U\_planeacion.php**

Web principal de usuario de planeación académica, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil de planeación. Ver manual del usuario en la opción perfiles de usuario, perfil planeación.

### **U\_registro.php**

Web principal de usuario registro académico, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil de registro. Ver manual del usuario en la opción perfiles de usuario, perfil registro.

### **U\_sacad.php**

Web principal de usuario secretario académico, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil

de usacad. Ver manual del usuario en la opción perfiles de usuario, perfil secretario académico.

### **U\_secretaria.php**

Web principal de usuario secretaria, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil de secretaria. Ver manual del usuario en la opción perfiles de usuario, perfil secretaria.

### **U\_talentoh.php**

Web principal de usuario talento humano, desde esta web se tiene acceso a algunos de los subsistemas y acciones de cronos que se encuentran bajo el perfil de talentoh. Ver manual del usuario en la opción perfiles de usuario, perfil talento humano.

## DESCRIPCIÓN DE CONTENEDORES DE ARCHIVOS

### Ubicación de contenedores /

Los contenedores en el sistema cronos son las carpetas que contienen y clasifican las webs de acuerdo al objeto sobre el cual se hará referencia, estos son: Aulas, Área, Calendario, Carga, config, disponibilidad, empleado, estadísticas, facultad, horario, images, js, materia, ofrecimiento, pensum, programa, recurso, reportes, requisito, reservaula.

El orden de la descripción de archivos dentro del contenedor se presenta de manera secuencial tal cual como un usuario entraría a hacer alguna acción sobre el objeto.

### CONTENEDOR CALENDARIO

Este subcontenedor es llamado para cargar fechas del calendario en un formulario. Los contenedores ofrecimiento, pensum, reserva aula lo tienen, varia es en el nombre de sus campos y sus funciones del formulario específico donde se desea trabajar por ello el contenedor calendario se encuentra en algunos sub contenedores. A continuación presentamos la descripción general de este contenedor:

#### **Calendario.php**

Web que a través de códigos de programación script arma y crea un calendario.

#### **Index.php**

Esta web muestra el calendario en un navegador.

#### **Javascripts.js**

Archivo javascript que determina el tamaño de la ventana donde se abre la web del calendario, además de llevar el campo de formulario php donde se desea que quede insertada la fecha.

#### **Estilo.css**

Archivo de estilo de cascada que le da una presentación especial al calendario.

## **CONTENEDOR CONFIG**

**Numero de Archivos:**9

**Peso:** 22.3 Kb

Contiene los archivos de configuración, funcionamiento, control y seguridad general del sistema.

### **Conexión.php**

Este archivo determina el tipo de base de datos(ORACLE-PostgreSQL, MySQL, Acces)mediante la utilización del paquete de PEAR, en el cual se desea que trabaje el sistema, parámetros de conexión, ubicación de servicios de almacenamiento de archivos, y consultas a la BD. Esta web lleva dos parámetros esenciales SQL y VERSIONBD las cuales respectivamente son las consultas que desea hacer el usuario y la base de datos(semestre académico)sobre la cual quiere trabajar

### **Back.php**

Esta web solo puede ser accedida por el administrador del sistema. La web permite crear el backup de la base de datos del sistema directamente desde un navegador web. El archivo generado para la descarga se describe a continuación: Bkcronos+fecha(AAAMMDD).sql (ej: bkcronos20060528), si un usuario que no es administrador así este registrado en el sistema desea acceder al backup el sistema lo enviara a una interfaz de error(index3.php).

### **Nuevo\_periodo.php**

Crea un nuevo periodo académico generando una nueva base de datos(nuevo periodo académica) a partir de la actual(periodo en curso) y colocando el periodo actual en modo consulta.

**Nombreusuarioh.php:** muestra el nombre del usuario que ha ingresado al sistema pero solo para las interfaces principales o homes de usuario.

**Nombreusuario.php** muestra el nombre de usuario dentro de las webs que existen en los contenedores.

### **Codi.php**

Es otro modelo de seguridad además del MD5 utilizado en las webs de acceso al sistema. la contraseña de acceso a la base de datos de cronos en esta web es decodificada ya que siempre trabaja bajo una protección de 64Bits(base64decode) siempre que es solicitada en los parámetros de conexión.

### **Subcontenedor mnu /config/mnu**

Este subcontenedor presenta los archivos dinámicos de creación de menús de usuario y acciones acorde al perfil y al objeto al cual se desea acceder.

### **Mnu\_raiz.php**

Esta web crea dinámicamente el menú principal de navegación(ubicado en la parte izquierda de la interfaz del usuario) de usuario y puede variar de acuerdo al perfil del usuario que ha ingresado al sistema.

### **Mnu\_subc.php**

Este archivo crea dinámicamente el submenú principal, es decir los menús de las webs(ubicado en la parte izquierda de la interfaz del usuario) que se encuentran en los contenedores, el menú varía de acuerdo al perfil del usuario.

### **Mnu\_interno.php**

Este archivo presenta las acciones dinámicamente que el usuario desea hacer sobre el objeto que esta consultando, estas acciones varían dependiendo del perfil del usuario que esta accediendo.

Una vez presentado la configuración y operación general del sistema y descrito sus archivos, se muestra a continuación la forma de acceso al sistema por parte de cualquier usuario y posteriormente la descripción de los contenedores secundarios que clasifican al sistemas de acuerdo al objeto sobre el cual se desea operar.

## **SECUENCIA DE ENTRADA POR PARTE DEL USUARIO WEB**

Archivos de ejecución consecutiva para el acceso a CRONOS:

1. Index.php
2. Indexe.php
3. Enter.php
4. Connexion.php
5. Codi.php
6. urlppal

- La descripción de los archivos anteriores la puede encontrar en “Archivos de raíz de sitio Pág. 2”

## COMO SE CONSTRUYE UNA WEB O INTERFAZ DE USUARIO CRONOS

Una web o IU Cronos se forma en base a objetos php incrustados. Una Web de Cronos se divide y construye en 3 partes descritas a continuación

**Parte Uno:** Es la parte superior de forma horizontal donde reposa el objeto nombreusuario.php y la imagen que identifica en que lugar se encuentra del sistema.

**Parte Dos:** Es la parte Donde reposa el menú de navegación de usuario el cual se ubica en la parte izquierda del navegador en forma vertical.

**Parte tres:** Esta parte se ubica en todo el centro de la web entre el objeto incrustado nombredeusuario.php y mnu\_raiz.php,es aca donde se mostraran las acciones que usted podrá hacer sobre el sistema y objetos determinados según su ubicación en el sistema.

## DESCRIPCIÓN DE SUBCONTENEDORES

Raíz /contenedor

### CONTENEDOR ÁREA

**Numero de Archivos:**13

**Peso:** 91.6 Kb

Este contenedor clasifica todas las acciones(Eliminar, Crear, Modificar, Consultar) que un usuario puede hacer o solicitar sobre el objeto área siempre y cuando el perfil se lo permita:

#### CREAR

##### **crear\_area.php**

Web que permite crear las áreas a los diferentes programas, existe un caso especial en el momento de la creación del área: si el sistema reconoce que se trata de la facultad de contaduría, creara una casilla de verificación especial llamada hcomponente que le permitirá identificar si es área o componente, de acuerdo a la resolución 3459 de diciembre 30 de 2003 la cual inscribe las materias dentro de áreas o componentes para este tipo de facultades.

##### **Crea\_area1.php**

Web que se encargara de insertar en la BD los datos del área que se crea en crea\_area1.php

#### MODIFICAR

##### **mod\_area.php**

Web que pide los datos del área que se desea modificar o actualizar.

##### **Mod\_area1.php**

Web que carga los datos del área seleccionada en mod\_area.php para poder ser modificados y/o actualizados.

##### **Mod\_area2.php**

Web que recibe los datos modificados por el usuario y realiza la acción de UPDATE en la tabla de área.

#### ELIMINAR

##### **Borra\_area.php**

Web que pide los datos del área que se desea eliminar

##### **Borra\_area2.php**

Web que elimina el área seleccionada en borra\_area.php

## **CONSULTAR**

### **Con\_area.php**

Web que muestra las áreas asociadas a una facultad, si el sistema detecta que es administrador este deberá seleccionar la facultad, en caso de no ser administrador, automáticamente el sistema detecta la facultad a la que pertenece el usuario y listara las áreas.

En la parte inferior de la web existen dos tipos de consulta(Materias por área y Docentes por área)los cuales lo redireccionaran al contenedor reportes.

## **CONTENEDOR AULAS**

**Numero de Archivos:**20

**Peso:** 195Kb

Este contenedor clasifica todas las acciones(Eliminar, Crear, Modificar, Consultar) que un usuario puede hacer o solicitar sobre el objeto tipos de aulas y aulas siempre y cuando el perfil se lo permita:

pero para poder crear las aulas primero se deben tener(y crear) los tipos de aulas que existen en la Universidad como lo son Laboratorios, salas de sistemas, estudio de TV, y otras que puedan existir en el futuro.

## **CREAR**

### **Crea\_tipoaula.php**

Web que crea los tipos de aulas que pueden existir en la Universidad.

### **Crea\_tipoaula1.php**

Web que inserta los datos asignados para el nuevo tipo de aula.

## **ELIMINAR**

### **Borra\_tipoaula.php**

Web que pide el tipo de aula que se desea eliminar.

### **Borra\_tipoaula2.php**

Web que elimina el tipo de aula que el usuario selecciono en borra\_aula.php

## **MODIFICAR**

### **Mod\_tipoaula.php**

Web que pide el tipo de aula a modificar.

### **Mod\_tipoaula1.php**

Web que trae y muestra los datos del tipo de aula que se desea modificar.

### **Mod\_tipoaula2.php**

Web que modifica y/o actualiza el tipo de aula seleccionado.

## **CONSULTAR**

### **Con\_tipoaula.php**

Muestra la descripción de los parámetros del tipo de aula.

## **CREAR**

### **Crea\_aula.php**

Web que crea un aula a partir de su tipo (Clasificación). Los tipos pueden ser laboratorios, salas, aulas de cátedra o cualquier tipo de aula que se clasifique en la realidad para ser ingresado como nuevo en el sistema, de esta manera por ejemplo se puede tener un aula de tipo laboratorio, o un aula de tipo estudio de radio.

### **Crea\_aula1.php**

Web que recibe e inserta los datos a la BD ingresados por el usuario en crea\_aula.php

## **MODIFICAR**

### **Mod\_aula.php**

Web que pide los datos del aula que se desea modificar o actualizar.

### **Mod\_aula1.php**

Web que carga los datos del aula seleccionada en mod\_aula.php para poder ser modificados y/o actualizados.

### **Mod\_aula2.php**

Web que recibe los datos modificados por el usuario y realiza la acción de UPDATE en la tabla de aula.

## **BORRAR**

### **Borra\_aula.php**

Web que pide los datos del área que se desea eliminar

### **Borra\_aula2.php**

Web que elimina el área seleccionada en borra\_area.php

## **CONSULTAR**

### **Con\_aula.php**

Web que muestra los datos descriptivos de las aulas creadas o que existen en el sistema.

## **CONTENEDOR CARGA**

**Numero de Archivos:23**

**Peso: 255Kb**

Es acá donde se crean las cargas académicas para su posterior asignación en los académicos.

Este contenedor presenta todos los archivos relacionados con la clasificación(tipos de carga) de las cargas académicas es decir, los proyectos y/o tareas que se crean a partir de una clasificación general en que se reglamenta en cada facultad, centro de Investigación o dependencia de la Universidad para su posterior asignación a docentes y administrativos. Esta clasificación actualmente se divide en: Producción Escrita, asesorias a practicas académicas, Administración. Capacitación, por nombrar algunos.

### **OPERACIÓN EN TIPOS DE CARGAS(Crear, Consultar, Modificar, Eliminar)**

#### **CREAR**

##### **Crea\_cargaadicional.php**

Crea un tipo de carga y la clasifica en un tipo que puede ser Investigación, Producción escrita, asesorias, comites etc.

##### **Crea\_cargaadicional1.php**

Recibe e inserta en la base de datos los datos que vienen de crea\_cargaadicional.php

#### **CONSULTAR**

##### **Con\_cargaadicional.php**

Consulta y muestra las cargas académicas que tiene cada facultad creada

#### **ELIMINAR**

##### **Borra\_cargaadicional.php**

Selecciona una carga académica para eliminar

##### **Borra\_cargaadicional2.php**

Elimina la carga academica que selecciono el usuario

#### **MODIFICAR**

##### **Mod\_cargaadicional.php**

Selecciona una carga académica para modificar

##### **Mod\_cargaadicional1.php**

Muestra los datos de la carga académica seleccionada para que el usuario determine que desea eliminar.

### **Mod\_cargaadicional2.php**

Recibe y actualiza los datos de la carga que se modifíco.

## **OPERACIÓN DE LOS TIPOS DE CARGAS SOBRE LOS ACADÉMICOS**

### **CREAR**

#### **Crea\_adicionaldocente.php**

Asigna en un docente algún tipo de carga académica existente, luego de seleccionar la carga despliega un horario en el cual el usuario elige cuando desea trabajarle a la carga a partir de su disponibilidad.

#### **Crea\_adicionaldocente1.php**

Asigna la cantidad de horas para el proyecto según el docente y el horario en el cual se trabajará en el proyecto teniendo en cuenta la disponibilidad del docente.

#### **Crea\_adicionaldocente2.php**

Almacena los datos referentes al docente, proyecto, no de horas del proyecto y horario asignado, realizando el descuento de horas de la disponibilidad del docente.

### **MODIFICAR**

#### **Mod\_adicionaldocente.php**

Selecciona el docente y el proyecto que desea modificar, de los que tiene asignado.

#### **Mod\_adicionaldocente1.php**

Teniendo en cuenta el docente y proyecto, puede cambiar la cantidad de horas asignadas y el horario para el proyecto teniendo en cuenta la disponibilidad del docente.

### **CONSULTAR**

#### **Con\_adicionaldocente.php**

Muestra las cargas académicas adicionales de cada docente.

### **BORRAR**

#### **Borra\_adicionaldocente.php**

Solicita la cedula del docente que se le desea eliminar la carga

#### **Validadocenteelim.php**

Web que determina si el docente tiene alguna carga académica adicional

**Borra\_adicionaldocente1.php**

Muestra las cargas académicas que tiene el docente para seleccionar alguna y borrarla.

**Borra\_adicioanldocente2**

Elimina en el sistema la carga académica seleccionada.

**CONTENEDOR DISPONIBILIDAD**

**Numero de Archivos:**13

**Peso:** 176Kb

**CREAR****Crea\_disponibilidad.php**

Selecciona el docente al cual se le ingresará la disponibilidad de horario semana.

**Crea\_disponibilidad1.php**

Según el tipo de contrato del docente, el formulario de ingreso se optimiza para el ingreso de información, si es medio tiempo o tiempo completo posee un seleccionador, si es catedrático posee casillas de selección para ingresar los datos.

**Crea\_disponibilidad2.php**

Almacena la disponibilidad del docente en la base de datos.

**MODIFICAR****Mod\_disponibilidad.php**

Selecciona el docente al cual se requiere cambiar la disponibilidad.

**Mod\_disponibilidad1.php**

Según el tipo de contrato del docente, el formulario de modificación se optimiza para el ingreso de información, si es medio tiempo o tiempo completo posee un seleccionador, si es catedrático posee casillas de selección para ingresar los datos.

**Mod\_disponibilidad2.php**

Almacena los cambios en la disponibilidad del docente en la base de datos.

**CONSULTAR****Con\_disponibilidad.php**

En este archivo se debe seleccionar el docente sobre el cual se desea hacer la consulta de la disponibilidad académica

**Con\_disponibilidad1.php**

Muestra la disponibilidad del docente

## **BORRAR**

No aplica para disponibilidad

## **CONTENEDOR EMPLEADO**

**Numero de Archivos:**25

**Peso:** 269Kb

## **CREAR**

### **Crea\_empleado.php**

Selecciona el tipo de empleado a crear(Administrativo o Docente)

### **Docente.php**

Carga una web que solicita los datos para crear el empleado tipo docente.

### **Docente1.php**

Inserta los datos principales del empleado y automáticamente crea un subformulario que se despliega si el usuario selecciono si pertenece a otra facultad u otra área o ambas mostrando otras areas y facultades diferentes a la principal.

### **Docente\_fa.php**

Esta web se activa solo si el usuario selecciono que el docente pertenecía a un area o facultad secundaria o ambas, su acción es la de insertar los datos de las areas y facultades secundarias en la base de datos

### **Otros.php**

Carga una web que solicita los datos para crear el empleado tipo Administrativo.

### **Crea\_otros1.php**

Inserta los datos del empleado Administrativo.

## **MODIFICAR**

### **Mod\_empleado.php**

Solicita la cedula del empleado a modificar.

### **Validaempleado.php**

Verifica la existencia de la cedula digitada y en caso de ser exitoso selecciona el perfil del empleado(Administrativo o Docente) encontrado para enviarlo a la interfaz de modificación respectiva.

### **Mod\_empleado1.php**

Interfaz de modificación de docentes. Carga los datos del empleado docente para que el usuario determine que desea modificar.

### **Moddocente1.php**

Inserta los datos del empleado pero si desea modificar o agregar una facultad o área secundaria activara otra web para terminar su proceso de actualización (docente\_modfa.php)

### **Docente\_modfa.php**

Web que muestra las areas y facultades secundarias que el docente desea

### **Mod\_empleado1p.php**

Interfaz de modificación de Administrativo. Carga los datos del empleado administrativo para que el usuario determine que desea modificar.

**Mod\_empleado2p.php**

Actualiza los datos que el usuario modifico en el empleado Administrativo.

**CONSULTAR**

**Con\_empleado.php**

Muestra una descripción general de los usuarios de la Universidad discriminados por facultad, en la parte inferior de la web muestra algunas opciones para hacer consultas mas especificas, las cuales se describirán mas adelante

**Det\_empleado.php**

Solicita al usuario que ingrese la cedula del empleado a modificar.

**Cedulano.php**

Esta Web solo se activa si la cedula no existe o fue mal digitada.

**Det\_empleado1.php**

Muestra un reporte del usuario seleccionado en una web blanca lista para ser impresa si es necesario.

Existen otras busquedas especializadas de empleados las cuales son accedidas desde la web con\_empleado pero que no residen en este contenedor, estas son:

**Docentes por facultad secundaria** Para su descripción consulte el contenedor reportes

**Docentes por Área** Para su descripción consulte el contenedor reportes

**Carga académica en empleados** Para su descripción consulte el contenedor carga

**Docentes por Contratación** Para su descripción consulte el contenedor reportes

**Docentes por Escalafón** Para su descripción consulte el contenedor reportes

**BORRAR**

**Borra\_empleado.php**

Solicita la cedula del empleado a modificar

**validaempleadoelim.php**

Verifica que el empleado exista, en caso de ser exitoso verifica el tipo de empleado(Administrativo o Docente) para mandar a interfaz de eliminación.

**borra\_empleado1.php**

Web de eliminación de docente, en la cual se le indica al usuario que se va a eliminar a un empleado.

**borra\_empleado2.php**

Verifica que el empleado no tenga datos que hagan al sistema depender de el, en caso de ser negativo, no se podrá eliminar al usuario.

**borra\_empleado1p.php**

administrativos falta arreglar

Web de eliminación de Administrativo, en la cual se le indica al usuario que se va a eliminar al usuario.

**borra\_empleado2p.php** falta arreglar

Verifica que el empleado no tenga datos que hagan al sistema depender de el, en caso de ser negativo, no se podrá eliminar al usuario.

**CONTENEDOR ESTADÍSTICAS**

**Numero de Archivos:**4

**Peso:** 70Kb

**Index.php**

Presenta las estadísticas mas comunes que permiten apoyar la toma de desiciones para planeacion académica y apoyos para docentes

**Es\_ee.php**

Estadísticas por escalafón

**Es\_ec.php**

Estadísticas por contrato

**Es\_cel.php**

Estadística que presenta el numero de empleados de la Universidad y/oi facultad clasificados por empresa prestadora de servicio de telefonía móvil (OLA-COMCEL-MOVISTAR)

**CONTENEDOR FACULTAD**

**Numero de Archivos:**12

**Peso:** 101Kb

**CREAR**

Crea\_facultad.php

Crea una facultad o dependencia de la Universidad

Crea\_facultad1.php

Inserta en la base de datos los datos que vienen de crea\_facultad.php

**MODIFICAR**

**Mod\_facultad.php**

Solicita al usuario que seleccione la facultad a modificar.

**Mod\_facultad1.php**

Carga los datos de la facultad para que el usuario puede elegir que modificar.

### **Mod\_facultad2.php**

Actualiza en la base de datos los datos que el usuario modifico de la facultad.

## **CONSULTAR**

### **Con\_facultad.php**

Muestra una vista básica de las facultades o dependencias de la Universidad.

### **Md\_facultad.php**

Esta web se activa si el usuario en la vista basica da clic sobre alguna facultad especifica. De esta manera muestra un web detallada de todos los datos de la facultad, como programas asociados, areas, numero de docentes, proyectos de investigación etc

### **Md\_facultad1.php**

Es otra forma de acceder a md\_facultad.php, (Home Administrador)solo que esta pide los datos directamente de la facultad que desea ver.

## **BORRAR**

### **Borra\_facultad.php**

Sollicita al usuario que seleccione la facultad que desea eliminar.

### **Borra\_facultad2.php**

Elimina la facultad seleccionada por el usuario siempre y cuando el sistema no dependa de ella.

## **CONTENEDOR HORARIO**

**Numero de Archivos:**14

**Peso:** 182Kb

## **CREAR**

### **Crea\_horario.php**

Solicita los datos de facultad, programa, pensum y programa para la asignación de horarios de clase en todos los grupos existentes.

### **Crea\_horario2.php**

Permite el cambio de docentes en las materias, asigna la hora inicial y cantidad de horas para cada materia en determinado día de la semana. Posee validaciones respecto al número de horas de la materia, disponibilidad de docente, horas previas seleccionadas, evitando cruces de horas o materias.

### **Crea\_horario3.php**

Almacena los datos de docentes, horario asignado y cambios en la disponibilidad.

## **MODIFICAR**

### **Mod\_horario.php**

Solicita los datos de facultad, programa, pensum y programa para el cambio de horarios de clase en todos los grupos existentes.

#### **Mod\_horario2.php**

Permite el cambio de docentes en las materias, cambios en la hora inicial y cantidad de horas para cada materia en determinado día de la semana. Posee validaciones respecto al número de horas de la materia, disponibilidad de docente, horas previas seleccionadas, evitando cruces de horas o materias.

#### **Mod\_horario3.php**

Almacena los cambios en docentes, horario asignado y cambios en la disponibilidad.

### **CONSULTAR**

#### **Con\_horario.php**

Le permite al usuario seleccionar dos tipos de consulta, una de horarios por docente y otra de horarios por semestre-grupo

### **BORRAR**

No aplica por que se modifica

### **CONTENEDOR MATERIA**

**Numero de Archivos:**14

**Peso:** 98.4Kb

### **CREAR**

#### **Crea\_materia.php**

Solicita los datos para la creación de materias.

#### **Crea\_materia1.php**

Inserta los datos suministrados por el usuario en la base de datos creando así la materia, posteriormente el servidor entrega una respuesta de transacción exitosa mostrando el código de la materia que fue asignado, el cual consta de dos dígitos por parámetro bajo el siguiente orden:

Programa+Semestre+Consecutivo+Area

### **MODIFICAR**

#### **Mod\_materia.php**

Solicita la materia a modificar.

#### **mod\_materia1.php**

Carga los datos de la materia seleccionada para ser modificados.

#### **mod\_materia2.php**

Actualiza en la base de datos la materia en base a los parámetros que el usuario modifico.

## **CONSULTAR**

### **Con\_materia.php**

Muestra una consulta básica de las materias de un programa. Permitiendo crear un enlace dentro del nombre de la materia para mostrar una consulta mas especifica.

### **Con\_materiab.php**

Web que se activa en el momento de hacer un clic sobre alguna materia en con\_materia.php, esta web muestra una consulta mas avanzada de la materia que se selecciono.

Las siguientes dos consultas no están dentro del contenedor pero pueden ser accedidas desde la parte inferior de con\_materia.php.

**rep\_mat\_area.php** Para su descripción consulte el contenedor reportes

**mat\_sem.php** Para su descripción consulte el contenedor reportes

## **BORRAR**

### **Borra\_materia.php**

Solicita al usuario la materia a modificar.

### **borra\_materia2.php**

Elimina la materia que el usuario selecciono siempre y cuando el sistema no dependa de ella.

## **CONTENEDOR OFRECIMIENTO**

**Posee contenedor calendario**

**Numero de Archivos:22**

**Peso: 137.4Kb**

## **CREAR**

### **Crea\_ofrecimiento.php**

Selecciona el programa, pensum, y cantidad de grupos a generar ofrecimiento.

### **Crea\_ofrecimiento2.php**

Según la cantidad de grupos a ofrecer, aparecen unos seleccionadores para indicar que materias serán ofrecidas.

### **Crea\_ofrecimiento3.php**

Permite asignarle un docente a cada ofrecimiento.

### **Crea\_ofrecimiento4.php**

Almacena la información en ofrecimiento y docente ofrecimiento.

## **MODIFICAR**

### **Mod\_ofrecimiento.php**

Selecciona el programa, pensum y materia a modificar.

### **Mod\_ofrecimiento2.php**

Modifica los datos generales del ofrecimiento como son cantidad de alumnos, docente, fecha de iniciación, fecha de finalización, aprobados, reprobados, estado del ofrecimiento.

### **Mod\_ofrecimiento3.php**

Almacena en el sistema los cambios realizados.

## **CONSULTAR**

### **Con\_ofrecimiento.php**

Selecciona el programa y semestre sobre el cual desea ver los ofrecimientos.

### **Con\_ofrecimiento1.php**

Muestra los ofrecimientos creados para el programa y el semestre.

## **BORRAR**

No aplica

## **CONTENEDOR PENSUM**

**Posee contenedor calendario**

**Numero de Archivos:16**

**Peso: 93.6 Kb**

## **CREAR**

### **Crea\_pensum.php**

Web que pide los datos al usuario para la creación del pensum

### **crea\_pensum1.php**

Web que inserta los datos del pensum en la base de datos y crea automáticamente un formulario donde carga las materias para asociarlas al pensum

### **pensum\_materias.php**

Inserta en la base de datos las materias que el usuario selecciono para el pensumm

## **MODIFICAR**

### **mod\_pensum.php**

Web que solicita al usuario el pensum a modificar.

### **mod\_pensum\_materias.php**

Carga las materias asociadas al pensum para que el las quite y/o adicione mas.

**mod\_pensum1.php**

Actualiza en la base de datos los datos modificados en el pensum.

## **CONSULTAR**

**Con\_pensum.php**

Muestra los datos generales del pensum.

## **BORRAR**

No aplica para pensum

## **CONTENEDOR PROGRAMA**

**Numero de Archivos:12**

**Peso: 115Kb**

## **CREAR**

**crea\_programa.php**

Solicita los datos para la creación del programa.

**crea\_programa1.php**

Inserta los datos del programa a crear, pero primero verificando que no exista.

## **MODIFICAR**

**mod\_programa.php**

Solicita el programa a modificar.

**mod\_programa1.php**

Carga los datos del programa para que el usuario determine que desea modificar.

**mod\_programa2.php**

Actualiza en la base de datos los datos modificados por el usuario.

## **CONSULTAR**

**con\_programa.php**

Muestra los datos generales del programa permitiendo hacer una consulta avanzada sobre ellos al momento de hacer clic sobre alguno de los que se listan en la consulta.

**con\_programab.php**

Muestra una consulta avanzada del programa en una web tipo reporte para su fácil impresión

## **BORRAR**

**borra\_programa.php**

Web que solicita al usuario que programa quiere eliminar.

**borra\_programa2.php**

Web que elimina de la base de datos el programa seleccionado siempre y cuando el sistema no dependa de el.

**CONTENEDOR RECURSOS**

**Numero de Archivos:**22

**Peso:** 158 Kb

Este contenedor presenta la creación de recursos generales con los que puede contar la Universidad y también la asignación y operación de estos sobre las materias y las aulas.

**Recursos****CREAR****crea\_recurso.php**

Esta web permite crear cualquier tipo recurso audiovisual o inmobiliario que pueda ser posteriormente adicionado como recurso académico en aulas o materias

**crea\_recurso1.php**

Esta web adiciona o crea en la base de datos el recurso que se acaba de crear en crea\_recurso1.php

**MODIFICAR****mod\_recurso.php**

Web que pide al usuario el recurso a modificar

**mod\_recurso1.php**

Carga los datos del recurso que se desea modificar

**mod\_recurso2.php**

Web que actualiza los datos del recurso seleccionado para modificar.

**CONSULTAR****con\_recurso.php**

Muestra los recursos que se han creado

**BORRAR****borra\_recurso.php**

Web que pide al usuario el recurso a eliminar.

**borra\_recurso2.php**

Web que elimina el recurso siempre y cuando no hayan otros datos del sistema que dependan de el.

## Recursos en materias

### CREAR

**crea\_recurso materia.php**

Web que asigna un recurso a una materia.

**crea\_recurso materia1.php**

Web que inserta los datos en la base de datos.

### MODIFICAR

No aplica para esta

### CONSULTAR

**con\_recurso materia.php**

Muestra los recursos que tiene una materia.

### BORRAR

**borra\_recurso materia.php**

Selecciona la materia a la que desea eliminarle el recurso.

**borra\_recurso materia1.php**

Carga los recursos que tiene asignados la materia.

**borra\_recurso materia2.php**

Elimina el recurso asignado a la materia.

## Recursos en aulas

### CREAR

**crea\_recurso aula.php**

Web que asigna un recurso a una materia.

**crea\_recurso aula1.php**

Web que inserta los datos en la base de datos.

### MODIFICAR

No aplica para esta

### CONSULTAR

**con\_recurso aula.php**

Muestra los recursos que tiene una materia.

### BORRAR

**borra\_recurso aula.php**

Selecciona la materia a la que desea eliminarle el recurso.

**borra\_recurso aula1.php**

Carga los recursos que tiene asignados la materia.

**borra\_recursoaula2.php**

Elimina el recurso asignado a la materia.

## **CONTENEDOR REPORTES**

**Numero de Archivos:**26

**Peso:** 271 Kb

### **Reporte Materias por Area**

Este reporte es solicitado desde la web de consulta de áreas o consulta de materias y muestra una consulta de las materias según un área específica:

**rep\_mat\_area.php**

Solicita los datos para clasificar la consulta de materias por area.

**rep\_mat\_area1.php**

Muestra las materias asociadas al área seleccionada en rep\_mat\_area.php

### **Reporte Materia-Semestre**

El siguiente reporte muestra las materias de un determinado programa y semestre académico.

**mat\_sem.php**

Solicita programa y semestre para el reporte

**mat\_sem1.php**

Muestra las materias a partir de la clasificación por programa y semestre

### **Reporte Docentes-Contrato**

Este reporte muestra los docentes asociados a un tipo de contrato. Se accede a través de la web principal de consulta de empleados.

**doc\_contrato.php**

Solicita los datos para clasificar la consulta de docente por contrato en doc\_contrato1.php.

**doc\_contrato1.php**

Muestra los empleados asociados a una facultad y clasificados en un tipo de contratación (TC, MT, C).

### **Reporte Docentes-Escalafón**

Este reporte muestra los docentes asociados a un escalafón. Se accede a través de la web principal de consulta de empleados.

#### **doc\_esc.php**

Solicita los datos para clasificar la consulta de docente por escalafón.

#### **doc\_esc1.php**

Muestra los empleados de una facultad clasificados de acuerdo a su escalafón.

### **Reporte Docentes-Facultad Secundaria**

Este reporte muestra los docentes asociados a una segunda facultad diferente a la principal en la cual se le hace el contrato. Se accede a través de la web principal de consulta de empleados.

#### **Doc\_facs.php**

Muestra las facultades secundarias a las que pertenece un docente.

### **Reporte Docentes-Área Principal**

El siguiente reporte muestra los docentes asociados a un área específica de la facultad principal asociada al docente:

#### **doc\_mat\_area.php**

Solicita los datos para clasificar la consulta de docente por área.

#### **doc\_mat\_area1.php**

Muestra los docentes de una facultad clasificados de acuerdo a su área principal asociada.

### **Reporte Horario-Docente**

El siguiente reporte muestra el horario de un determinado docente partiendo de su facultad.

#### **Hor\_docente.php**

Solicita el docente a quien se le desea consultar el usuario

#### **Hor\_docente1.php**

Muestra el horario semanal del docente seleccionado

### **Reporte Horario-Grupo**

El siguiente reporte muestra el horario de un determinado grupo partiendo del programa, semestre y pensum.

#### **Hor\_grupo.php falta el ultimo list**

Solicita los parámetros de selección de grupo.

### **Hor\_grupo1.php**

Muestra el horario del grupo que se ha seleccionado en hor\_grupo.php

### **Reporte Proyectos por tipo**

El siguiente reporte muestra cada una de las cargas académicas que se crean en cada facultad, clasificadas por su tipo(Investigación, Asesorías, etc), su acceso se hace desde el menú de carga y consulta de tipos de carga.

#### **Pro\_tipo.php**

Solicita al usuario seleccionar el tipo de proyectos existentes por facultad que desea consultar

#### **Pro\_tipo1.php**

Muestra los proyectos asociados a su tipo con las descripciones generales de cada uno

### **Reporte Planeacion académica**

**Modo de acceso:** Menú Estadísticas

Se puede decir que junto con el horario es el reporte mas importante de cronos. En el se muestra toda asignación de carga académica que se planea en el inicio del semestre y durante el semestre, discriminando por filas a el docente y con sus respectivas columnas discriminadas por cada una de las cargas adicionales que fueron asignadas a el con el numero de horas respectivas, Involucrando al final de cada columna una suma total de horas asignadas para cada tipo de carga adicional y una suma total de filas mostrando la suma total de horas del docente.

Si la suma de horas de la fila del docente supera el numero de horas tope en la que se inscribe su contrato este saldrá en una celda de color azul clara avisando al decano o superior donde debe eliminar horas, o apoyarlo a tomar una decisión, en caso que el contrato no le permita bonificación:

#### **Md\_planeacion:**

Solicita el tipo de contrato sobre el cual desea visualizar el reporte de planeacion academica.

#### **Md\_planeacion1.php**

Muestra el reporte de planeacion académica final de todos los docentes asociados al contrato y a cada uno de los proyectos que maneja la facultad.

## **CONTENEDOR REQUISITO**

**Numero de Archivos:**13

**Peso:** 91.7 Kb

### **CREAR**

#### **crea\_requisito.php**

Permite crear requisitos(Pre y Co) a las diferentes materias. Aquí se debe seleccionar el tipo de requisito a crear y la materia base sobre la cual se creara.

#### **requisitomaterias.php**

Carga las materias que van a ser el tipo de requisito de la materia base.

#### **crea\_requisito1.php**

Inserta en la base de datos los requisitos que se acaban de seleccionar.

### **MODIFICAR**

#### **mod\_requisito.php**

Aquí se selecciona la materia sobre la cual se desea hacer la modificación.

#### **mod\_requisitomaterias.php**

Carga los requisitos de la materia sobre los cuales se desea modificar(Quitar o colocar nuevos requisitos).

#### **mod\_requisito1.php**

Actualiza en la base de datos los requisitos de la materia que se cambiaron.

### **CONSULTAR**

#### **con\_requisito.php**

Se selecciona la Materia sobre la cual se desea tener información de los requisitos

#### **con\_requisito1.php**

carga la información de la materia con sus respectivos requisitos

### **BORRAR**

No aplica para requisito

## **CONTENEDOR RESERVA\_AULA**

**Posee contenedor calendario**

**Numero de Archivos:**15

**Peso:** 82.5Kb

### **CREAR**

#### **crea\_reservaaula.php**

Aca seleccionamos los parámetros iniciales de una reserva como lo es la fecha y la hora

#### **crea\_reservaaula1.php**

Carga las aulas disponibles de acuerdo a los parámetros iniciales

**crea\_reservaaula2.php**

Crea La reserva del aula en la base de datos.

**CONSULTAR**

con\_reservaaula.php

Muestra las aulas que estan reservadas

**BORRAR**

**borra\_reservaaula.php**

Se debe seleccionar el aula sobre la cual se desea eliminar la reserva, igual que la hora de inicio y fecha

**borra\_reservaaula2.php**

Se elimina la reserva del aula en la base de datos en el caso de que si exista la reserva.

**MODIFICAR**

No aplica para este menú

**CONTENEDOR IMAGES**

Archivos de Imagen: 223

Peso: 3.61 Megas

Este contenedor es el encargado de clasificar y almacenar todas las imágenes de la aplicación, los cuales son llamados desde las paginas webs.

Archivos Web PHP 290

## **ANEXO U GENERACIÓN DE UN NUEVO PERIODO ACADÉMICO**

Para generar un nuevo periodo académico es recomendable primero que todo crear el backup de la base de datos desde la web en la opción del menú del administrador BACKUP, a continuación se muestra como opera la creación de un nuevo semestre académico.

- 1- se genera el nombre del próximo periodo académico, descrito a continuación:  
cronos+año+periodo(el periodo puede ser 1 o 2 dependiendo la fecha en que se haga el nuevo periodo si es un mes  $\leq 7$  será 1 de lo contrario será dos. Un periodo académico puede ser cronos2006\_P.
- 2- Se crea la base de datos con el nombre descrito en el punto uno, en base al periodo académico que se cursa en el momento, esta base de datos quedara exactamente igual y con los mismos datos que la del semestre en curso.
- 3- Posteriormente creada la base de datos del nuevo periodo académico, se ejecuta una línea de comando que hace que el periodo académico que estaba en curso quede ahora en modo consulta y se active el nuevo periodo académico creado en el punto dos para el trabajo de este nuevo periodo.
- 4- Los nombres de las bases de datos son ingresados en una base de datos diferente a la del ACA y son solicitados en la web principal para que el usuario decida sobre que periodo académico desea acceder.

Para generar un nuevo periodo académico el administrador del sistema accederá a la opción CREAR NUEVO PERIODO en el panel central de acciones del sistema en la web de Administrador y solo con dar clic en la opción, esta se ejecutara automáticamente.

NOTA: Solo el administrador del sistema puede crear backups y periodos académicos nuevos.

**Para acceder al código fuente de cómo generar un periodo académico nuevo visite el Archivo nuevo\_periodo.php en el contenedor config**