# Edd

# Sistema web de planificación horaria

NOMBRE: Raúl Eduardo Durán Carrasco

CARRERA: Ingeniería en Informática

ASIGNATURA: Taller de Proyecto de Software

PROFESOR: Alan Cartes Fica

FECHA:

ÍNDICE

[1 INTRODUCCIÓN 5](#_Toc486719893)

[2 DESCRIPCIÓN EMPRESA CLIENTE 7](#_Toc486719894)

[3 FORMULACIÓN Y DELIMITACIÓN DEL PROBLEMA EN ESTUDIO 8](#_Toc486719895)

[3.1 PROYECTO PERTINENTE 8](#_Toc486719896)

[3.2 JUSTIFICACIÓN 9](#_Toc486719897)

[3.3 ALCANCES 10](#_Toc486719898)

[3.4 LÍMITES 13](#_Toc486719899)

[4 OBJETIVOS 14](#_Toc486719900)

[4.1 OBJETIVO GENERAL 14](#_Toc486719901)

[4.2 OBJETIVOS ESPECÍFICOS 14](#_Toc486719902)

[5 METODOLOGÍA DE PROGRAMACIÓN A UTILIZAR 15](#_Toc486719903)

[5.1 CICLO DE VIDA 15](#_Toc486719904)

[6 FACTIBILIDADES 16](#_Toc486719905)

[6.1 VIABILIDAD TÉCNICA 16](#_Toc486719906)

[6.1.1 DISEÑO DE BASE DE DATOS 16](#_Toc486719907)

[6.1.2 MODELAMIENTO UML 16](#_Toc486719908)

[6.1.3 LIBRERÍAS 16](#_Toc486719909)

[6.1.4 LENGUAJES DE PROGRAMACIÓN 17](#_Toc486719910)

[6.1.5 MOTOR DE BASE DE DATOS 18](#_Toc486719911)

[6.1.6 SERVIDOR 19](#_Toc486719912)

[6.1.7 EQUIPAMIENTO PARA EL DESARROLLO 19](#_Toc486719913)

[6.2 VIABILIDAD OPERACIONAL 20](#_Toc486719914)

[6.3 VIABILIDAD FINANCIERA 21](#_Toc486719915)

[6.4 VIABILIDAD ECONÓMICA 23](#_Toc486719916)

[6.4.1 COSTOS 23](#_Toc486719917)

[6.4.2 COMERCIALIZACIÓN 24](#_Toc486719918)

[6.5 VIABILIDAD LEGAL 24](#_Toc486719919)

[7 MODELO DE NEGOCIOS 26](#_Toc486719920)

[8 DISEÑO LÓGICO 27](#_Toc486719921)

[8.1 MODELO IDENTIDAD RELACIÓN 27](#_Toc486719922)

[8.2 DIAGRAMA DE CLASES 28](#_Toc486719923)

[8.3 DIAGRAMA DE CASOS DE USO 29](#_Toc486719924)

[8.4 DIAGRAMA DE ACTIVIDADES 31](#_Toc486719925)

[8.5 DISEÑO DE BASE DE DATOS 32](#_Toc486719926)

[9 REQUERIMIENTOS DEL SISTEMA 33](#_Toc486719927)

[9.1 REQUISITOS NO FUNCIONALES 33](#_Toc486719928)

[9.2 REQUISITOS FUNCIONALES 34](#_Toc486719929)

[10 POLITICAS DE PROGRAMACIÓN 43](#_Toc486719930)

[10.1 CONTROL DE VERSIONES 43](#_Toc486719931)

[10.2 MATRIZ DE RIESGOS 43](#_Toc486719932)

[10.3 PLAN DE CONTINGENCIA 43](#_Toc486719933)

[10.4 MEDIDAS MITIGADORAS 44](#_Toc486719934)

[10.5 PLAN DE PRUEBAS 44](#_Toc486719935)

[10.5.1 PRUEBAS UNITARIAS 44](#_Toc486719936)

[10.5.2 PRUEBAS DE INTEGRACIÓN 44](#_Toc486719937)

[10.6 CAJA BLANCA 44](#_Toc486719938)

[11 INTERFACES 45](#_Toc486719939)

[12 CONCLUSIÓN 46](#_Toc486719940)

[13 ANEXOS 47](#_Toc486719941)

# INTRODUCCIÓN

La planificación horaria demanda de una alta cantidad de recursos de tiempo y análisis, además de una altísima comunicación entre los actores involucrados en el proceso para evitar o disminuir los muy probables errores de choque de disponibilidad y falta de calidad en los horarios planificados.

Con esto dicho, el proyecto ayuda directamente a los encargados de planificación de la Universidad Tecnológica de Chile INACAP, sede Chillán con la compleja planificación de horarios académicos para las distintas secciones de cada carrera con la finalidad de mejorar la eficiencia del proceso de planificación mejorando la calidad de los horarios y el tiempo utilizado en este proceso.

En la actualidad, INACAP posee un sistema de planificación informático que permite realizar el proceso de una manera relativamente sencilla en relación a la cantidad de variables que debe manejar (incluyendo qué tan grandes son). Sin embargo, las planificaciones horarias siguen mostrando dificultades al momento de ser realizadas y con la alta cantidad de variables se producen errores de choques de disponibilidad de docentes y salas, teniendo que ocupar una alta cantidad de tiempo en este proceso, además, se producen errores implícitos que no están solucionados por ninguna normativa establecida en la institución, afectando negativamente en la calidad de trabajo de los docentes e, incluso, en su calidad de vida.

Como se acaba de mencionar, la única solución para esto sería implementar una normativa dentro de la institución educacional para mejorar la calidad de las cargas académicas de los docentes, lo cual implicaría una mayor preocupación por esto mismo, disminuyendo los errores mencionados con anterioridad. Esto se desarrollará en más profundidad dentro de los fundamentos teóricos. Sin embargo, la otra solución posible a esta problemática que, cabe destacar, no solucionaría el problema existente en los 7 docentes, sería un sistema web de planificación horaria automatizada que entregaría la posibilidad de entregar horarios académicos de forma visual planificados por el sistema mismo a través de las variables involucradas en esto. Este permitiría una disminución considerable en los tiempos y complejidad del proceso debido a que todo el análisis y concentración necesarias para el proceso estaría dada por este sistema, además, la excesiva complejidad y tiempo extra utilizado después del proceso en términos de cambios a posteriori de la planificación, también serían disminuidos. Este proyecto apunta a la mejora del proceso en términos de tiempo y complejidad en el manejo de las variables.

Se considera también necesario mencionar que, según los requerimientos iniciales del cliente, este sistema debiese haber sido un Sistema Experto, sin embargo, luego del análisis de las tecnologías posibles a utilizar y las capacidades y requisitos que entrega y solicita un Sistema Experto[[1]](#footnote-1), se llegó a la conclusión que la construcción como sistema no experto sería suficiente para satisfacer todos los requerimientos y expectativas de los stakeholders[[2]](#footnote-2). Recordar que siempre es necesario mantener una comunicación constante con todos los involucrados en el proyecto para buscar la mejor solución en conjunto y aumentar las probabilidades de éxito.

Desde el punto de vista técnico, el software utilizará una arquitectura de software de modelo de tres capas[[3]](#footnote-3), lo que permitiría un desarrollo de software web adecuado para el tipo de implementación del proyecto y requerimientos del cliente (seguridad, sobre todo). Con un total de diez áreas dentro de INACAP sede Chillán, se espera una cantidad de conexiones máximas de treinta al servidor durante el periodo de planificación.

# DESCRIPCIÓN EMPRESA CLIENTE

# FORMULACIÓN Y DELIMITACIÓN DEL PROBLEMA EN ESTUDIO

Antiguamente, las planificaciones de horario de los semestres académicos eran realizadas en papel, sin un sistema que permita la temprana visualización de posibles topones o errores en la claridad de la disponibilidad de los docentes y salas. Este proceso tomaba un excesivo tiempo debido a su altísima complejidad, haciéndolo muy difícil de llevar.

En la actualidad, las planificaciones de horario de los semestres académicos son realizadas a través de un sistema incorporado en la plataforma de INACAP. Este sistema disminuye los errores mencionados con la forma anterior de planificación, sin embargo, no se simplifica el proceso y la complejidad de este (aunque no es tanta como en la forma anterior) sigue siendo un tema a la hora de planificar. Los tiempos siguen siendo muy altos y aún se presentan problemas de topones, falta de claridad en las disponibilidades de docentes y salas conllevando a una alta probabilidad de modificaciones en el horario en un plazo extenso de marcha blanca.

## PROYECTO PERTINENTE

El proyecto será un sistema web (debido a los requerimientos del cliente para favorecer a la portabilidad) que permitirá la generación de horarios académicos de forma automatizada a través de todas las variables necesarias del proceso como la disponibilidad docente, las salas, las mayas, las asignaturas, entre otros. El sistema razonará en relación de los requerimientos de la institución para la generación de horarios y entregaría hasta tres soluciones horarias académicas, disminuyendo considerablemente la complejidad y el tiempo utilizado en este proceso por parte de las personas. Debido a esto mismo, también permitiría disminuir cualquier complejidad de cambio de disponibilidad de docente, crecimiento de la sección, etc, ocurrida después de la creación de estos horarios, ya que el usuario solo tendría que modificar la disponibilidad del docente o el dato necesario y volver a solicitar la planificación para esa sección, semestre y maya en específico.

## JUSTIFICACIÓN

Primero que todo en esta sección, cabe destacar lo siguiente:

La normativa mencionada en la introducción implicaría que la disponibilidad de los docentes estuviera bajo ciertos márgenes regulatorios, eliminando la posibilidad existente de que ellos sean quienes tengan el control total sobre su disponibilidad, teniendo que adaptarse. De cualquier otra forma, al existir muchísima variación dentro de todas las disponibilidades de los docentes, es prácticamente imposible realizar un horario de tal grado de que este sea compacto en relación a la carga académica docente. Esta nueva normativa sería absolutamente necesaria si se buscase entregar una solución al gran problema de las ventanas en los docentes ya que, de no implementarse, tendría que seguir llenándose los espacios vacíos en el horario por aquellos docentes con disponibilidad que, comúnmente, poseen una disponibilidad completa.

Sin embargo, y en razón de tesis y como proceso de investigación aplicada, se ha optado por la solución de un sistema web, dejando registrado todo lo necesario para que en un futuro pueda implementarse esta nueva normativa y demostrando que, sin esta, por muy automatizado o inteligente que sea el sistema, no podrá entregar una solución que satisfaga a todos los implicados en este proceso.

En cuanto al sistema web, la planificación actual es un proceso manejado por seres humanos y requiere una alta capacidad de análisis y concentración directamente proporcional al tamaño del área en la que se sitúa el planificador. Sumado al trabajado que debe realizar además de planificador (atención de alumnos, docentes y otras actividades), la concentración resulta ser muy dependiente de todo esto y disminuiría su capacidad, aumentando la cantidad de errores implícitos posibles dentro del proceso, además, si no solo tomamos en consideración al mejor de los casos, este proceso se 15 vuelve más tortuoso al tener que modificar las variables. Sin embargo, si logramos reemplazar el manejo humano en los pasos más complejos del proceso por maquinas, seremos capaces de disminuir considerablemente toda esta carga, no solo disminuyendo los tiempos y la complejidad, sino también los errores implícitos, beneficiando considerablemente a los stakeholders.

De esta manera, se cumpliría con todos los requerimientos del proyecto y se impactaría de forma positiva en el proceso de planificación en cuanto a los usuarios.

## ALCANCES

El sistema web de planificación horaria Edd permitiría la planificación de horarios académicos en la Universidad Tecnológica de Chile, INACAP, sede Chillán, de forma automatizada y entregando de una a tres soluciones horarias. Esto lo realizaría a través del análisis de la información ingresada por los usuarios al sistema. La información que se menciona está constituida por:

* Docente:
  + Disponibilidad
    - Cantidad de horas máximas
    - Días
    - Bloques diarios
  + Prioridad
  + Ruta formativa
  + Información personal
  + Asignaturas asignadas
  + Área
* Maya:
  + Asignaturas
    - Historial de docentes
    - Especialidad o no
    - Cantidad de horas
    - Necesidad de equipamiento tecnológico
    - Nombre
    - Semestre
  + Código
  + Carrera
  + Cantidad de asignaturas
* Secciones:
  + Curso
    - Tamaño
    - Código
  + Diurno o Vespertino
  + Código
  + Nombre
  + Carrera
  + Semestre
* Salas:
  + Características
    - Capacidad de alumnos
    - Equipamiento tecnológico
    - Ubicación
    - Código
  + Disponibilidad
* Carrera:
  + Código
  + Nombre
* Área
  + Código
  + Nombre

El sistema solo planificará una semana en relación al rango de días escogidos antes del proceso. Esto es debido a que no se entrega un archivo para exportación y, por ejemplo, si un día llegase a ser festivo, este solo se salta y no se recuperan clases fuera del rango escogido para estas.

## LÍMITES

* Interfaz para ser usada en web.
* Lenguajes y tecnologías en uso: NodeJS, JSON, Javascript, HTML5, CSS3, MariaDB, Enterprise Architect, Ubuntu Server, VPS.
* El VPS donde se implementará el sistema deberá poseer la capacidad de soportar y ejecutar aplicaciones escritas en NodeJS y bases de datos en MariaDB.
* La base de datos deberá contar con un Usuario y Área ya inscritos.
* El sistema podrá planificar solo hasta tres soluciones horarias, entendiendo que puede existir la posibilidad de crear menos soluciones con el mínimo de una.
* El sistema no entregará un archivo para ser exportado, sino que solo permitirá la visualización para que este sea copiado de forma visual al sistema de planificación de INACAP.
* Si un Docente termina su jornada al último bloque del día, al día siguiente, no podrá entrar al primer bloque del día. Esto podrá saltarse en razón que lo solicite el Docente.
* La sumatoria de las horas totales de las Asignaturas asignadas a un Docente no puede superar la cantidad de horas especificadas en la disponibilidad de este. Si se asigna una Asignatura a un Docente y con esta supera la cantidad de horas en su disponibilidad, entonces el sistema deberá informar de esto al usuario y no permitir asignarle la Asignatura.
* No debe existir un día con dos o menos bloques de clases por Sección.
* El Docente no debe tener más de trece bloques académicos asignados por día.
* Los horarios planificados deben ser compactos, es decir, no debe existir más de una ventana de cuarenta y cinco minutos por semana en cada Sección.
* Puede quedar un día libre dentro del rango de días escogido.
* Al modificar la cantidad de horas disponibles en el Docente, esta no debe bajar más allá del 30% en relación al semestre anterior símil, es decir, si se modifica el segundo semestre del año, no debe bajar más del 30% en relación al segundo semestre del año anterior.

# OBJETIVOS

## OBJETIVO GENERAL

Desarrollar un sistema web capaz de planificar de una a tres soluciones horarias para los semestres académicos de la Universidad Tecnológica de Chile, INACAP, sede Chillán antes del 30 de noviembre del 2017.

## OBJETIVOS ESPECÍFICOS

* Disminuir la carga y la complejidad de la planificación horaria de los semestres académicos para los usuarios.
* Disminuir errores de choque y falta de claridad en la disponibilidad de los docentes y salas.
* Planificar horarios de mayor calidad según las reglas establecidas por la institución.

# METODOLOGÍA DE PROGRAMACIÓN A UTILIZAR

## CICLO DE VIDA

Se utilizará una adaptación de SCRUM[[4]](#footnote-4). El uso de una metodología ágil o adaptación de esta permite ir verificando y validando cada mantenedor y algoritmo implementado en el proyecto de una manera más veloz y no sujeta a una extensa cantidad de pasos que pueden llegar a ser innecesarios. Ya que el desarrollo ágil se basa en el desarrollo iterativo e incremental[[5]](#footnote-5), este modelo trabajaría completa y funcionalmente al proyecto en cuestión, el cual incluye una alta cantidad de mantenedores y algoritmos que dependen de estos.

# FACTIBILIDADES

## VIABILIDAD TÉCNICA

Acerca de los requisitos técnicos del proyecto, se pueden dividir en distintas categorías.

### DISEÑO DE BASE DE DATOS

Para el diseño físico de la base de datos, se utilizarán diagramas MER que permiten un fácil entendimiento y visualización previa y posterior en cuanto a bases de datos relacionales.

### MODELAMIENTO UML

Para el modelamiento del comportamiento, estructura e interacción del usuario con el sistema, se utilizará UML v2.5 que permite la diagramación de los procesos, actividades, pasos, respuestas, etc, del sistema de una forma sencilla y aclaradora.

### LIBRERÍAS

Debido a que el software se desarrollará para plataforma web, se ha optado por la utilización de la librería NodeJS[[6]](#footnote-6).

NodeJS es un entorno de ejecución para JavaScript construido con el motor de JavaScript V8 de Chrome. Node.js usa un modelo de operaciones E/S sin bloqueo y orientado a eventos, que lo hace liviano y eficiente. Esto permitiría múltiples conexiones al sistema y trabajar con procesos de alta demanda sin necesidad de que se exista un bloqueo por parte del servidor. A diferencia de lenguajes como PHP, NodeJS permite la escritura de un sistema asíncrono con una mayor naturalidad, otorgando más estabilidad y disponibilidad al sistema en procesos de alta demanda de procesamiento.

### LENGUAJES DE PROGRAMACIÓN

Ya que el sistema se desarrollará en un ambiente web para el fácil acceso y portabilidad demandada por los usuarios, se ha concluido que el mejor lenguaje a utilizar como base sería Javascript. La programación en Javascript estará dictada por el estilo impuesto por la librería NodeJS y se añadirá la tecnología JSON[[7]](#footnote-7) para el fácil traspaso de la información bajo este lenguaje.

Cabe destacar, que la elección de un lenguaje de programación puede reducirse a una decisión trivial, debido a que los lenguajes de programación no son un elemento crítico al momento de hablar de seguridad y/o performance, sobre todo en el proyecto en cuestión. Recordar que este tipo de temas tienen mayor relación con las técnicas de programación aplicadas, cómo protejo el servidor y base de datos además de la programación en sí y siempre manteniendo los pilares de la seguridad informática. Entonces, ¿por qué la elección de Javascript y no cualquier otro lenguaje que permita la programación del lado del servidor? La elección es gracias a lo moderno del lenguaje (se adapta completamente a los más modernos estilos de programación), posee una comunidad muy activa que siempre está actualizando sus frameworks[[8]](#footnote-8) mejorando cada vez más el rendimiento del lenguaje y entrega una muy completa compatibilidad con todos los navegadores modernos más populares.

### MOTOR DE BASE DE DATOS

Tomando en consideración las ventajas que ofrece, la facilidad de implementación con los lenguajes anteriormente mencionados, el reconocimiento de los datos utilizados por el sistema y la capacidad de poder estructurar estos mismos, se escoge como motor de base datos a MariaDB.

MariaDB es un sistema de administración relacional de bases de datos basado en MySQL. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

Algunas de las ventajas de MariaDB son:

* Columnas dinámicas, que proporcionan al usuario columnas virtuales en las tablas.
* El optimizador de MariaDB -que se encuentra en el núcleo de cualquier SGBDfunciona claramente más rápido con cargas complejas que MySQL.
* Escalabilidad y flexibilidad.
* Robusto soporte transaccional.

Acerca de por qué no otro sistema de base de datos relacional, este sistema satisface a cabalidad los requerimientos del proyecto y utilizar un sistema distinto y más complejo de base de datos sería innecesario. Acerca de sistemas de base de datos parecidos a MariaDB, se pudo identificar casi ninguna diferencia, solo diferenciándose en aspectos completamente opcionales o no requeridos para el proyecto en cuestión.

### SERVIDOR

Debido a que el sistema se realizará en un ambiente web, es necesario un servidor que aloje el sistema completo para su acceso.

Se ha optado por un servidor privado virtual que permite la instalación del sistema operativo Ubuntu Server que entrega todo lo necesario para el funcionamiento del sistema, es gratuito y adaptable para cualquier situación. El servidor de alojamiento debe al menos tener:

* 1GB de RAM.
* Procesador de 1 núcleo
* 20GB de espacio disponible
* 1TB de transferencia

Estas son las prestaciones mínimas que debería poseer el sistema para funcionar en relación al software utilizado y los datos de los que se alimentará el sistema. Sin embargo, es necesario más adelante evaluar la necesidad de velocidad mínima del procesador debido al alto procesamiento que se necesitaría.

### EQUIPAMIENTO PARA EL DESARROLLO

Para el desarrollo del sistema, es necesario el arriendo de un notebook o computador de escritorio con, al menos, las siguientes características:

* Windows 8.1/10 x64
* INTEL Celeron de 1.6 GHz o superior
* 4GB de memoria RAM
* 500GB de Espacio en Disco

Las características mencionadas satisfacen las necesidades de los softwares ocupados. Recordar que Javascript puede ser programado a través de simples editores de código sin la necesidad de un IDE[[9]](#footnote-9).

## VIABILIDAD OPERACIONAL

En cuanto a los conocimientos necesarios para desarrollar el proyecto, se definen las siguientes tablas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CARGO | DESARROLLO WEB | DISEÑO | SERVIDORES | BD |
| Programador Web | **4** | **2** | **1** | **1** |
| Diseñador Gráfico | **2** | **4** | **0** | **0** |
| Administrador de Servidores | **1** | **0** | **4** | **2** |
| Administrador de Base de Datos | **0** | **0** | **1** | **4** |

* *Tabla del 0 al 4 donde 0 es ningún conocimiento previo y 4 es conocimiento de la lógica y estilo de programación/diseño/administración según sea el caso.*

## VIABILIDAD FINANCIERA

El proyecto se financiaría con un crédito de consumo en relación con su costo. Según el SERNAC, estos son los bancos más adecuados para realizar un crédito de consumo:

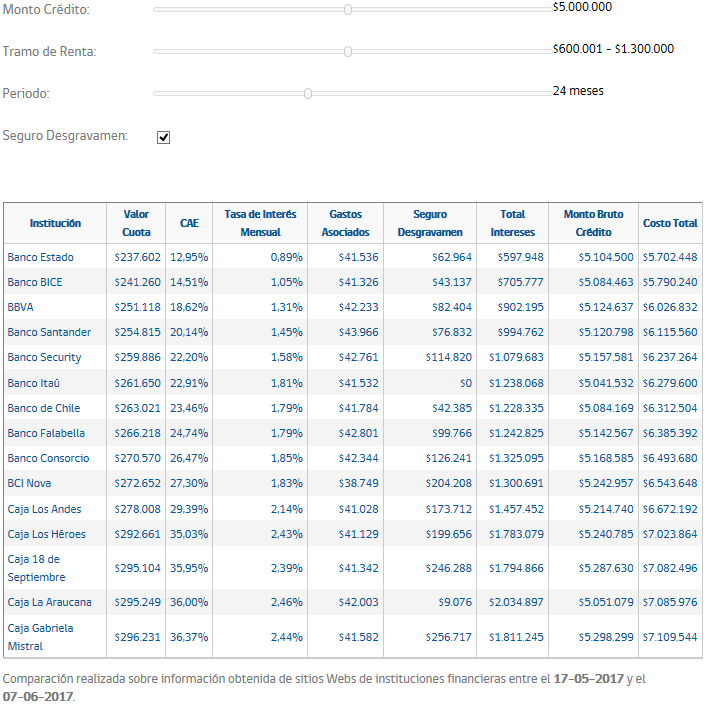


Ilustración - Tabla de créditos de consumo en bancos chilenos

Se han realizado las siguientes simulaciones en los bancos BBVA y Banco Estado, dando los siguientes resultados:

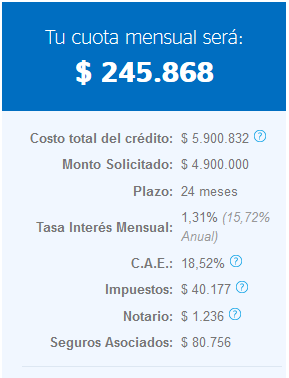


Ilustración - Simulación de crédito de consumo en el banco BBVA



Ilustración - Simulación de crédito de consumo en el Banco Estado

*\* Se solicita $4.900.000 en la simulación del banco BBVA debido a que solo permite cifras de 100.000 en 100.000.*

Acerca de los ingresos que se obtendrán por el sistema, estos serán a través de la compra de códigos serial en la plataforma web que permitan planificar una cierta cantidad de horarios.

## VIABILIDAD ECONÓMICA

En esta sección se verán los costos monetarios del proyecto, como se financiará y las posibilidades de comercialización de este.

### COSTOS

Los costos del proyecto están representados y ordenados en la siguiente tabla:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Cantidad | 1º Mes | 2º Mes | 3º Mes | 1º Año | 2º Año |
| VPS | 1 | $3.351 | $3.351 | $3.351 | $40.212 | $40.212 |
| Notebook | 1 | $179.990 | - | - | $179.990 |  |
| Ubuntu Linux | 1 | $0 | - | - | - | - |
| Ubuntu Server | 1 | $0 | - | - | - | - |
| NodeJS | 1 | $0 | - | - | - | - |
| MariaDB | 1 | $0 | - | - | - | - |
| Certificado SSL | 1 | $48.990 | - | - | $48.990 | $48.990 |
| Diseñador Gráfico | 1 | - | $600.000 | - | $600.000 | - |
| Ingeniero en Informática | 1 | $1.300.000 | $1.300.000 | $1.300.000 | $3.900.000 | - |
| TOTAL | **9** | **$3.435.682** | **$1.903.351** | **$1.303.351** | **$4.769.192** | **$89.202** |

Tabla - Tabla de costos del proyecto

La empresa prestadora del servicio de VPS elegida es DigitalOcean debido a su buena reputación en la comunidad y el precio entregado (5 dólares). Las empresas que prestan servicios de arriendo de equipos son muy poco conocidas o lo hacen a empresas y eventos en particular, por lo tanto, se ha preferido adquirir un notebook que satisfaga las prestaciones necesarias para el desarrollo del proyecto. El certificado SSL ha sido cotizado en GoDaddy debido a su precio y buena reputación en la comunidad.

Finalmente, los sueldos a pagar fueron obtenidos desde la página del gobierno de Chile mifuturo.cl

### COMERCIALIZACIÓN

Las posibilidades de comercialización del proyecto se encontrarían en las instituciones académicas que tengan un nivel organizacional (a nivel de horario académico) parecido a la Universidad Tecnológica de Chile, INACAP.

La posibilidad más clara en la actualidad es la universidad ya mencionada, debido a que esta es la solicitante del sistema.

## VIABILIDAD LEGAL

Este proyecto está bajo protección de la ley N°17.336 señalando la propiedad intelectual según la actual constitución de Chile, protege la creación de la obra, así como también se refiere según el artículo 16 y 17 de la presente ley.

Tampoco infringe la Ley N° 19.223 en cuanto a la materia se refiere de los delitos informáticos en sistemas de información dictada por la actual constitución de la Chile, esta idea no aplica para ninguno de los 4 artículos de la presente ley.

Los softwares utilizados de manera gratuita en el desarrollo de este proyecto son utilizados bajo la licencia GNU permite la utilización de dicha herramienta con fines comerciales.

El software Sublime Text 3 es un caso especial de software propietario, sin embargo, en su página web no especifica que la versión de evaluación no pueda ser usada para fines comerciales.

# MODELO DE NEGOCIOS

# DISEÑO LÓGICO

El diseño lógico de sistemas se refiere a lo que hará el nuevo sistema

El diseño lógico es una descripción de los requisitos funcionales de un sistema. En otras palabras, es la expresión conceptual de lo que hará el sistema para resolver los problemas identificados en el análisis previo. A falta de este paso, los aspectos técnicos del sistema (como los dispositivos de hardware que deban adquirirse) con frecuencia oscurecen la solución.

## MODELO IDENTIDAD RELACIÓN

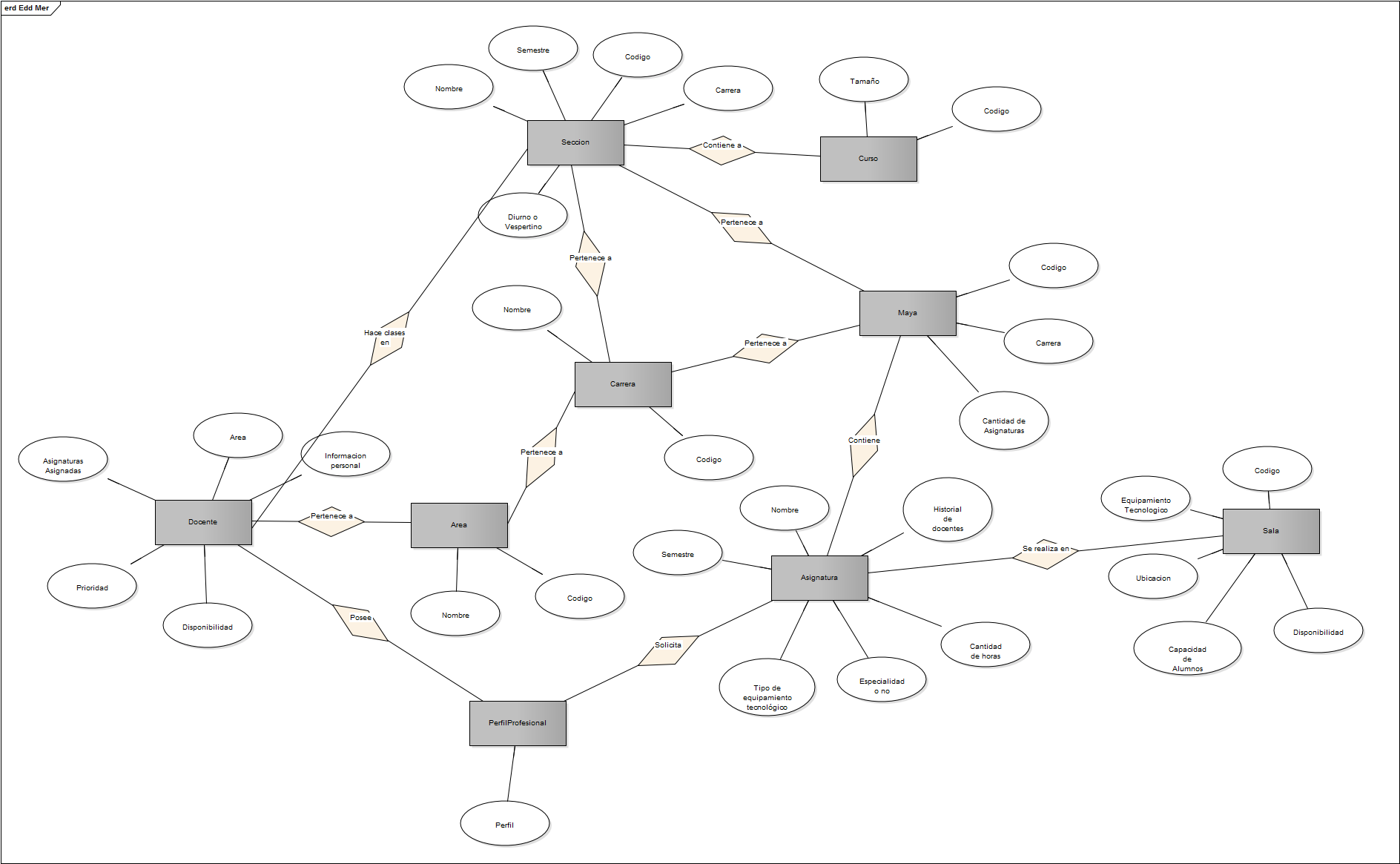


Ilustración - Modelo Entidad Relación del proyecto

## DIAGRAMA DE CLASES

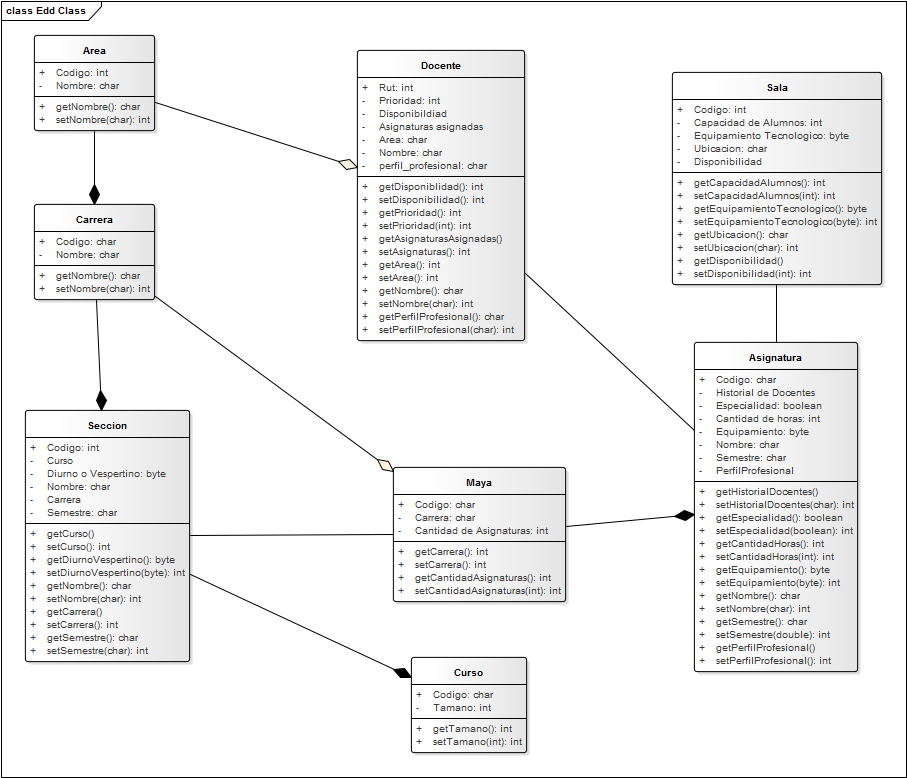


Ilustración - Diagrama de clases del proyecto

## DIAGRAMA DE CASOS DE USO

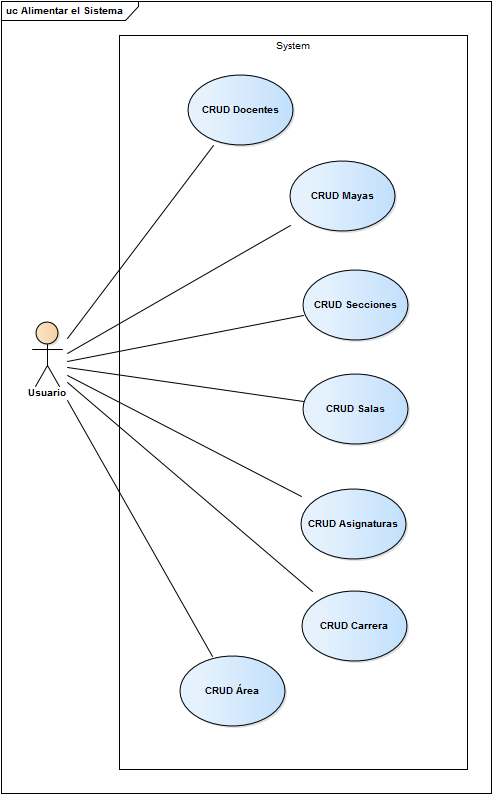


Ilustración - Diagrama de Casos de Uso para la alimentación del sistema



Ilustración - Diagrama de Casos de Uso de las demás funciones del sistema

## DIAGRAMA DE ACTIVIDADES

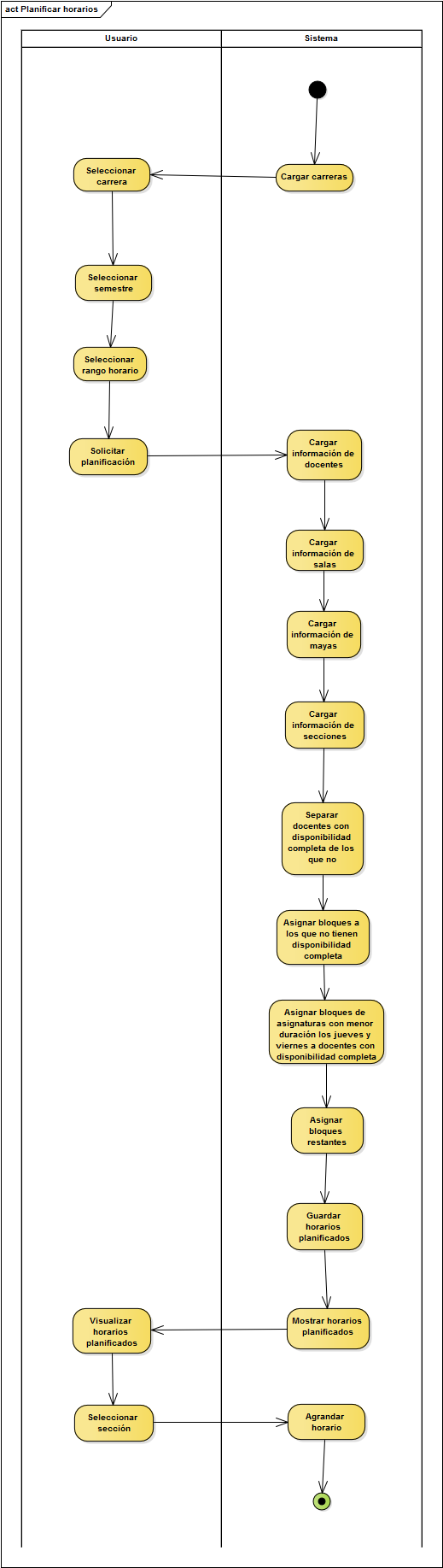


Ilustración - Diagrama de actividades del proceso "Planificar Horarios"

## DISEÑO DE BASE DE DATOS

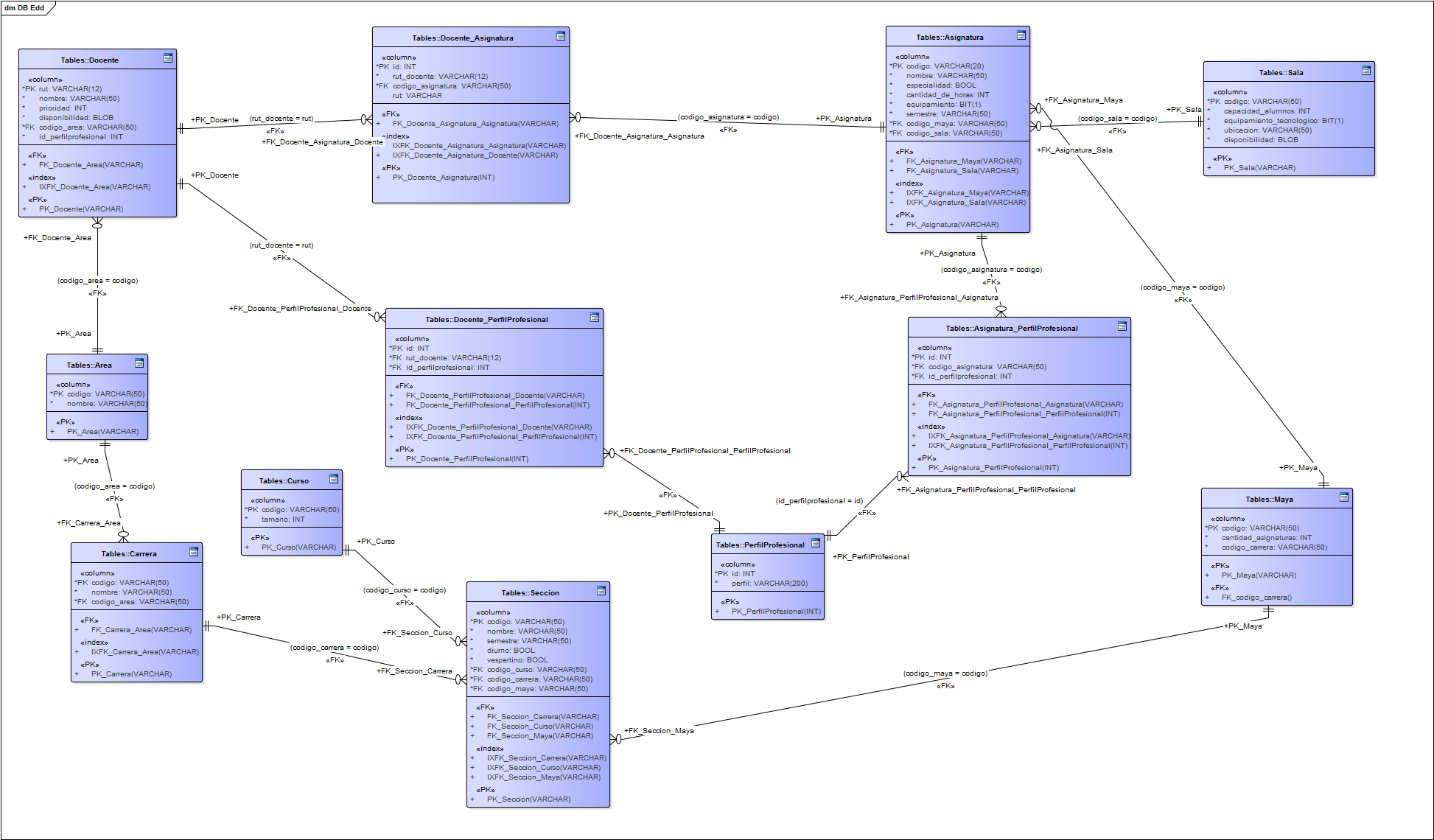


Ilustración - Diseño de base de datos del proyecto

# REQUERIMIENTOS DEL SISTEMA

## REQUISITOS NO FUNCIONALES

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF01 |
| **Nombre del Requerimiento:** | Interfaz del sistema. |
| **Características:** | El sistema presentará una interfaz de usuario sencilla y responsive para que sea de fácil manejo a los usuarios del sistema. |
| **Descripción del requerimiento:** | El diseño del sistema deberá ser de fácil manejo, es decir, debe poseer una navegabilidad y uso intuitivo además de presentar un formato responsive. |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RNF02 |
| **Nombre del Requerimiento:** | Seguridad |
| **Características:** | El sistema contará con un sistema de login para el acceso al panel administrativo. Sobre los datos, se asegurará su confiabilidad, integridad y accesibilidad. |
| **Descripción del requerimiento:** | Se asegurarán las bases de la seguridad de la información. El sistema deberá contar con los certificados y cortafuegos correspondientes. Se deberán usar técnicas de programación que estén en beneficio de la seguridad de la información. |
| **Prioridad del requerimiento:**  Alta | |

## REQUISITOS FUNCIONALES

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF01 |
| **Nombre del Requerimiento:** | CRUD Docentes |
| **Características:** | El usuario podrá crear, leer, actualizar y eliminar la información de los docentes en el sistema. |
| **Descripción del requerimiento:** | La información manipulada de los docentes en el sistema es la siguiente:   * Docente   + Disponibilidad     - Cantidad de horas máximas     - Días     - Bloques diarios   + Prioridad   + Ruta formativa   + Información personal   + Asignaturas asignadas   + Área   El sistema también debe almacenar las fechas en las que se hicieron modificaciones en la disponibilidad del Docente para satisfacer la restricción dictada con anterioridad.  La información de la Disponibilidad del Docente será cargada a través de un archivo Excel formateado. El sistema deberá ser capaz de leer este archivo y obtener los datos de este. |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF02 |
| **Nombre del Requerimiento:** | CRUD Mayas |
| **Características:** | El usuario podrá crear, leer, actualizar y eliminar la información de las mayas en el sistema. |
| **Descripción del requerimiento:** | La información manipulada de las mayas en el sistema es la siguiente:   * Maya   + Código   + Carrera   + Semestre     - Código     - Asignaturas asignadas   + Cantidad de asignaturas |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF03 |
| **Nombre del Requerimiento:** | CRUD Secciones |
| **Características:** | El usuario podrá crear, leer, modificar y eliminar la información de las secciones en el sistema. |
| **Descripción del requerimiento:** | La información manipulada de las secciones en el sistema es la siguiente:   * Secciones   + Curso     - Tamaño     - Código   + Diurno o Vespertino   + Código   + Nombre   + Carrera |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF04 |
| **Nombre del Requerimiento:** | CRUD Salas |
| **Características:** | El usuario podrá crear, leer, modificar y eliminar la información las salas en el sistema. |
| **Descripción del requerimiento:** | La información manipulada de las salas en el sistema es la siguiente:   * Salas   + Características     - Capacidad de alumnos     - Equipamiento tecnológico     - Ubicación     - Código   + Disponibilidad |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF05 |
| **Nombre del Requerimiento:** | CRUD Asignaturas |
| **Características:** | El usuario podrá crear, leer, modificar y eliminar la información de las asignaturas en el sistema. |
| **Descripción del requerimiento:** | La información manipulada en el sistema de las asignaturas es la siguiente:   * Asignaturas   + Historial de docentes   + Especialidad o no   + Cantidad de horas   + Necesidad de equipamiento tecnológico   + Nombre |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF06 |
| **Nombre del Requerimiento:** | CRUD Carrera |
| **Características:** | El usuario podrá crear, leer, modificar y eliminar la información de las carreras en el sistema. |
| **Descripción del requerimiento:** | La información manipulada en el sistema de las carreras es la siguiente:   * Carrera   + Código   + Nombre |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF07 |
| **Nombre del Requerimiento:** | CRUD Área |
| **Características:** | El usuario podrá crear, leer, modificar y eliminar la información del área académica en el sistema. |
| **Descripción del requerimiento:** | La información manipulada en el sistema del área académica es la siguiente:   * Área   + Nombre   + Código |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF08 |
| **Nombre del Requerimiento:** | Modificar información personal |
| **Características:** | El usuario podrá visualizar y modificar su información personal en el sistema. |
| **Descripción del requerimiento:** | La información que podrá manipular el usuario en el sistema acerca de sí mismo es la siguiente:   * Usuario   + Nombre completo   + Área   + Contraseña   + Deshabilitar cuenta |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF09 |
| **Nombre del Requerimiento:** | Planificar horario |
| **Características:** | El usuario podrá solicitar la planificación de soluciones de horario académico a través de la selección de la información necesaria e inicio del proceso en el sistema con la pulsación de un botón en el mismo. |
| **Descripción del requerimiento:** | La información necesaria para el inicio del proceso de planificación horaria es la siguiente:   * Sección * Maya * Docentes * Salas * Rango de horas diarios * Duración de bloque académico en minutos * Rango de días * Semestre   Recordar que todos los constructos deben estar completos con anterioridad según la información ingresada en el sistema, si no es así, el sistema informará del error. |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF10 |
| **Nombre del Requerimiento:** | Visualizar horarios planificados |
| **Características:** | El usuario tendrá acceso a la visualización de los horarios planificados anteriormente en el sistema. |
| **Descripción del requerimiento:** | Para poder visualizar los horarios académicos planificados con anterioridad en el sistema, el usuario deberá seleccionar de los siguientes datos:   * Semestre * Sección * Maya |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF11 |
| **Nombre del Requerimiento:** | Visualizar registro de actividades |
| **Características:** | El usuario podrá visualizar un registro de todas las actividades realizadas en el sistema por los usuarios en un rango de días determinado. También dispondrá de una caja de búsqueda para localizar cualquier registro que contenga las palabras que el usuario digite. |
| **Descripción del requerimiento:** | La información visualizada contendrá la siguiente información:   * Nombre del usuario * Acción realizada * Fecha y hora en la que realizó la acción |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

|  |  |
| --- | --- |
| **Identificación del requerimiento:** | RF12 |
| **Nombre del Requerimiento:** | Crear usuario |
| **Características:** | El usuario tendrá la capacidad de crear nuevos usuarios en el sistema. |
| **Descripción del requerimiento:** | La información necesaria para la creación de nuevos usuarios es la siguiente:   * Nombre de usuario * Nombre completo * Contraseña * Área |
| **Requerimiento NO funcional:** | * RNF01 * RNF02 |
| **Prioridad del requerimiento:**  Alta | |

# POLITICAS DE PROGRAMACIÓN

En cuanto a la programación del software, se debe mantener un orden y claridad para el correcto entendimiento del código. Es necesaria la aplicación de indentaciones en cada asignación y los comentarios deben encontrarse antes de cada función y clase explicando el qué hace, sus parámetros y variables. Estos dos últimos deben ir acompañados con la nomenclatura “@params”, “@vars” según sea el caso.

En relación al análisis de rendimiento para la programación, cada persona que participe en el desarrollo del proyecto debe ir llenando una planilla en Excel acerca de la actividad que está realizando, una breve descripción, una referencia al Issue en Github, fecha de inicio, fecha que se comprometió a terminar y fecha de término real, además de la cantidad de horas dedicada a esa actividad.

## CONTROL DE VERSIONES

En cuanto al control de versiones del software, se utilizará la herramienta Git en conjunto con la plataforma Github debido a la capacidad de alojar repositorios de forma gratuita y de fácil administración del desarrollo.

Cada actividad será iniciada como un Issue con sus correspondientes responsables, categorías y descripción. Se deberá crear una rama especial para el Issue en cuestión y solo realizar un merge de ramas al estar completamente testeada y funcional. La actividad debe ser referenciada en cada commit realizado para esta actividad.

## MATRIZ DE RIESGOS

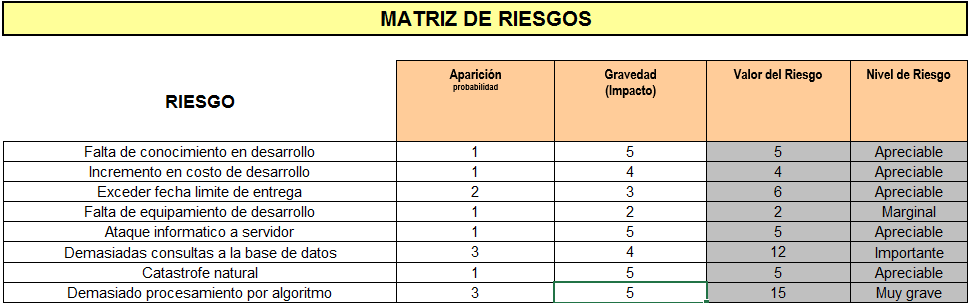


Ilustración - Matriz de riesgos del proyecto

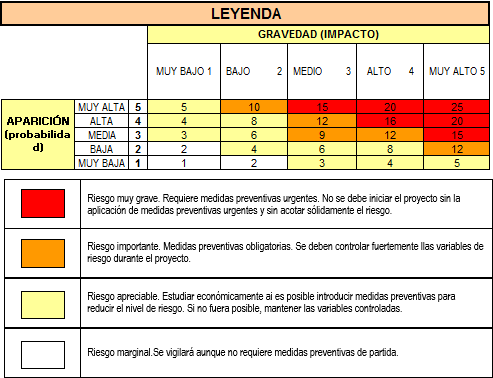


Ilustración - Leyenda de la matriz de riesgos

## PLAN DE CONTINGENCIA

## MEDIDAS MITIGADORAS

## PLAN DE PRUEBAS

### PRUEBAS UNITARIAS

### PRUEBAS DE INTEGRACIÓN

## CAJA BLANCA

# INTERFACES

# CONCLUSIÓN

Como se ha podido apreciar a lo largo del documento, el proyecto podrá mejorar considerablemente el proceso en cuestión disminuyendo la complejidad y el tiempo utilizado en esto gracias a la evaluación de las variables por parte del sistema y no de las personas. Además, también se debe considerar los cambios producidos después de la creación de un horario, los cuales impactan en gran medida a estos mismos ya que, en su mayoría, son cambios en la disponibilidad de los docentes.

Sin embargo, se recuerda encarecidamente que las necesidades de la institución van más allá de este proceso de planificación horaria, llegando hasta los docentes mismos que, sin ellos, no podría ser posible el funcionamiento de una institución académica. Como bien se ha mencionado en los Fundamentos Teóricos, se recuerda que la creación de una normativa que permita la regulación de la disponibilidad de los docentes es absolutamente necesaria para poder alcanzar un nivel de compactación adecuada y razonable, sin esta, las enormes variaciones entre cada disponibilidad seguirán existiendo haciendo prácticamente imposible alcanzar el nivel de compactación deseado o al menos cercano.

También se concluye que la comparación de los requerimientos mínimos de las tecnologías a utilizar para el desarrollo del proyecto con las tecnologías a contratar para la producción de este y la realización de distintas estimaciones con estos datos, entrega un nivel de certeza adecuado para poder seleccionar el mejor ambiente para este fin.

Finalmente, se concluye que un análisis de no solo las tecnologías, sino también a nivel operacional e incluso el conocimiento empírico entregarían una claridad suficiente para poder estimar los costos del proyecto.

# ANEXOS

1. https://es.wikipedia.org/wiki/Sistema\_experto [↑](#footnote-ref-1)
2. Involucrados e interesados en el proyecto. [↑](#footnote-ref-2)
3. https://es.wikipedia.org/wiki/Programaci%C3%B3n\_por\_capas [↑](#footnote-ref-3)
4. https://es.wikipedia.org/wiki/Scrum\_(desarrollo\_de\_software) [↑](#footnote-ref-4)
5. https://es.wikipedia.org/wiki/Desarrollo\_iterativo\_y\_creciente [↑](#footnote-ref-5)
6. https://nodejs.org/es/about/ [↑](#footnote-ref-6)
7. https://es.wikipedia.org/wiki/JSON [↑](#footnote-ref-7)
8. https://es.wikipedia.org/wiki/Framework [↑](#footnote-ref-8)
9. https://es.wikipedia.org/wiki/Entorno\_de\_desarrollo\_integrado [↑](#footnote-ref-9)