INSTITUTO NACIONAL DE APRENDIZAJE

MODULO

PROGRAMACIÓN ORIENTADA A OBJETOS

ETAPA 1

M. A. LUIS ANGEL CHACON ZUÑIGA

DANNY SOTO JIMENEZ

SEDE POLIVALENTE NARANJO

DICIEMBRE, 2021

**Contenido**

[**Contenido** 2](#_Toc93226031)

[**Tabla de ilustraciones** 2](#_Toc93226032)

[**Introducción** 3](#_Toc93226033)

[**Objetivo General** 4](#_Toc93226034)

[**Objetivos Específicos** 4](#_Toc93226035)

[**Problema Empresarial** 5](#_Toc93226036)

[**Diseño de Aplicación Orientada a Objetos** 6](#_Toc93226037)

[Historias De Usuario 6](#_Toc93226038)

[Anexos 19](#_Toc93226039)

**Tabla de ilustraciones**

[Ilustración 1 Administrar cursos 17](https://inacr-my.sharepoint.com/personal/701890285_ina_cr/Documents/Documents/Programador%20de%20aplicaciones/POO/Proyecto-POO-Etapa-01/Etapa%2001%20Mejorada.docx#_Toc92981393)

[Ilustración 2 administrar profesores 17](#_Toc92981394)

[Ilustración 3 crear matrícula 18](#_Toc92981395)

[Ilustración 4 agendar clase 18](https://inacr-my.sharepoint.com/personal/701890285_ina_cr/Documents/Documents/Programador%20de%20aplicaciones/POO/Proyecto-POO-Etapa-01/Etapa%2001%20Mejorada.docx#_Toc92981396)

**Introducción**

Durante este viaje que, a falta poco de terminar, se han logrado obtener grandez conocimientos en el área de programación, no ha sido fácil, y no será fácil. Cada módulo ha sido fiel comprobante de nuestro potencial, la metodología utilizada ha dado fruto y es tangible en lo que podemos crear.

Hoy nos enfrentamos a un nuevo reto, que en apariencia es muy difícil, quizás aún más que los anteriores, pero si realmente meditamos, lograremos tener una visión más amplia realmente de la dificultad. En estas instancias hemos recibido todas las herramientas necesarias y entre tantas habilidades adquiridas, podemos mencionar que la constante investigación es una de las más apreciadas sin duda. Cada problema o contratiempo al que nos enfrentemos, sabremos como solucionarlo y en caso contrario tenemos presente que alguien en algún momento tuvo la misma situación y salió adelante.

Durante el presente proyecto, se estará desarrollando la primera etapa de la arquitectura del software que estaremos implementado en este módulo, se tomarán en cuenta cada uno de los detalles relevantes por las partes involucradas como, por ejemplo; diseño, documentación, las herramientas con que se creará, entre otros. En la etapa 1, se estará haciendo énfasis en el diseño lógico de la base de datos, basados en las historias de usuario, diccionario de datos y los diagramas UML.

Es importante mencionar la importancia de esta etapa, ya que será el reflejo de nuestro proyecto final, cada mínimo detalle que dejemos escapar posiblemente nos afectará en las instancias finales del mismo. Por otro lado, si detallamos claramente cada requerimiento del proyecto, tendremos una satisfactoria finalización.

**Objetivo General**

Crear un software funcional, que nos permita poner en práctica nuestras habilidades. Para ello utilizaremos los conocimientos adquiridos en clase y, además, las investigaciones del módulo de programación orientada a objetos.

**Objetivos Específicos**

* Diseñar los diagramas necesarios para la creación del software.
* Evaluar las historias de usuario y generar una solución para cada una de ellas.
* Establecer los cimientos de nuestra base de datos, mediante un diccionario de datos.

**Problema Empresarial**

Diseñe y desarrolle una aplicación para administrar el proceso de matrícula de un Instituto de Idiomas, tome en cuenta que se ofrece: inglés, francés, alemán y mandarín, cada uno tiene un programa diferente ( inglés son 12 cursos con una duración de 80 horas cada uno, francés son 4 cursos con una duración de 100 horas cada uno, alemán son 10 cursos con una duración de 94 horas cada uno y mandarín son 14 cursos con una duración de 120 horas cada uno), las clases son en línea, distribuidas de la siguiente forma: 75% asincrónica, donde el estudiante puede consumir el material en el momento que lo considere pertinente, y el restante 25% debe ser a través de clases sincrónicas, para ello deberá programar la clase con al menos 2 días de anticipación. Cada matrícula es personalizada, por lo que cuando un estudiante se matrícula, el sistema debe indicar la fecha de finalización del curso, para ello el estudiante puede escoger entre 4 niveles de intensidad (bajo = 1 hora por día, medio = 2 horas por día, alto = 3 horas por día, intensivo = 4 horas por día), no debe contabilizar sábados, domingos ni feriados.

El sistema debe realizar como mínimo:

* Los mantenimientos de las tablas (CRUD).
* Matricular un estudiante, debe considerar la cantidad de horas del curso y las horas por día según el nivel seleccionado, debe calcular la fecha de finalización.
* Asignar el horario del profesor, tome en cuenta que el instituto atiende de 8am a 10pm de lunes a viernes, sin embargo, un profesor puede trabajar un máximo de 8 horas por día en el horario que solicite, un profesor solo puede impartir un idioma.
* Agendar horas de estudio sincrónicas, en este caso debe contemplar que la fecha seleccionada sea dos días posteriores a la actual y que se encuentre entre la fecha de inicio y final del curso, además debe contemplar que exista un profesor disponible en el horario solicitado, un profesor solo puede atender un estudiante por clase.
* Si un estudiante no se presenta a la cita sincrónica, pierde las horas de estudio y no se le reprograman, solo puede reprogramar la clase sincrónica con am menos 5 horas de anticipación.
* Debe llevar el control de pagos de los estudiantes, el costo por curso es de ¢2 000 por el total de horas del curso, un estudiante no puede matricular si tiene facturas pendientes.

**Diseño de Aplicación Orientada a Objetos**

# Historias De Usuario

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como asistente de gerencia.

Quiero administrar cursos.

Para poder ampliar la oferta académica del instituto.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 1 | **Usuario**: Asistente de gerencia | |
| **Nombre historia**: Administrar cursos. | | |
| **Prioridad en negocio**: Alta | | **Riesgo en desarrollo**:  Medio |
| **Puntos estimados**: 10 puntos. | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: Necesito administrar la información de los cursos ofrecidos por el instituto. | | |
| **Criterios de aceptación**:   1. Se creará un botón donde se pueda administrar los cursos, tendrá las siguientes opciones como datos a completar:    1. Identificación del curso.    2. Nombre del curso.    3. Cantidad de horas.    4. Precio por hora.    5. Estado del curso (Inactivo), para cambiar a “Activo” debe de haber al menos un estudiante matriculado. 2. En la interfaz donde se administran los cursos, tendrá disponibles los botones crear, actualizar y eliminar.   **Tareas asignadas:**   1. Crear el interfaz necesario para cumplir con la historia. Para ello se debe de crear el botón administrar cursos y seguidamente todos los campos referentes a la administración de este. 2. Crear una consulta que muestre los cursos disponibles. 3. Crear un procedimiento almacenado para crear un curso nuevo. 4. Crear un procedimiento almacenado para actualizar un curso. 5. Crear un procedimiento almacenado para eliminar un curso.   **Reglas de negocio:**   1. La administración de cursos únicamente será responsabilidad de la administración del centro de idiomas. 2. Cuando el sistema esté disponible para que los estudiantes puedan hacer matriculas, para ese momento ya deben estar los cursos creados. | | |

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como administrador del centro de idiomas.

Quiero administrar los datos de los profesores.

Para poder tenerlos disponibles cuando se realice

una matrícula.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 2 | **Usuario**: Administrador del centro de idiomas. | |
| **Nombre historia**: Administrar datos de profesores. | | |
| **Prioridad en negocio**:  Alta | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados**: 10 pts. | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: Los profesores deben de estar agregados en el sistema, para poder matricular el curso que el estudiante desee. | | |
| **Criterios de aceptación**:   * Tendremos a disposición un botón para “Administrar profesores”. * Cuando presionamos el botón “Administrar profesores” se mostrarán las opciones correspondientes, las cuales serán:   1. Crear profesor.   2. Actualizar profesor.   3. Eliminar profesor (En este caso la eliminación será únicamente lógica). * Se debe hacer una comprobación previa de la existencia del profesor para poder manipular los datos. * Cuando creamos un profesor se deben de solicitar la siguiente información:  1. Número de identificación (Número de cédula). 2. Nombre completo. 3. Número telefónico. 4. Correo electrónico. 5. Fecha de nacimiento. 6. Cuenta bancaria. 7. Dirección exacta. 8. Estado (Será activo o inactivo). 9. Curso certificado (inglés, francés, alemán o mandarín). 10. Borrado.  * Si necesitamos actualizar los datos del profesor, podremos buscarlo por medio del número de identificación. Seguidamente tendremos a disposición todos los campos para modificar con excepción del número de identificación (si este se encuentra registrado). * Para buscarlo tendremos que presionar un botón de buscar ubicado al lado del número de identificación ingresado.   **Tareas asignadas:**   1. Crear la interfaz gráfica para administrar los datos del profesor (botón). 2. Crear los procedimientos almacenados para crear, actualizar y eliminar un profesor, recalcando que la eliminación es únicamente lógica. 3. Crear una consulta para verificar si el profesor existe.   **Reglas de negocio:**   1. No se podrá eliminar un profesor completamente del sistema, para solucionar el problema se debe hacer un borrado lógico. | | |

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como estudiante.

Quiero matricular un curso.

Para poder estudiar el idioma desea.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 3 | **Usuario**: Estudiante. | |
| **Nombre historia**: Matricular un curso. | | |
| **Prioridad en negocio:**  Alta | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados**: 10 pts. | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: Me gustaría poder hacer la matrícula de un curso. | | |
| **Criterios de aceptación**:   1. Al crear una matrícula, se debe de ingresar el número de identificación para realizar el trámite (este será la referencia para nuestras futuras consultas). 2. Cuando doy clic en el botón de matrícula deben de mostrarse los espacios requeridos de información para matricularme como, por ejemplo: El número de identificación estudiante, el curso que desea matricular , la intensidad que desea (bajo = 1 hora por día, medio = 2 horas por día, alto = 3 horas por día, intensivo = 4 horas por día), fecha de finalización (en este se debe hacer el cálculo necesario para mostrar la fecha final, se toma en cuenta la intensidad, el horario laboral del instituto es de lunes a viernes de 8 am a 10 pm, exceptuando días feriados y fines de semana), fecha de inicio. 3. Cuando selecciono las opciones de intensidad y curso, se actualiza la fecha final automáticamente. 4. Es indispensable que se calculen los porcentajes de clases asincrónicas y sincrónicas, siendo un 75% asincrónica y 25% sincrónica. Esta información se deberá mostrar en un campo cada uno, se actualizará cada vez que se hagan cambios en la matrícula, pero sin que se haya finalizado el trámite.   **Tareas asignadas:**   * Crear una consulta que muestre los cursos disponibles, tomar como referencia el idioma, para que los cursos se puedan cargar. * Crear una consulta de los cursos que están disponibles en el idioma seleccionado. * Crear un procedimiento almacenado para matricular estudiante. * Hacer los cálculos necesarios para determinar el día de finalización del curso, tomando en cuenta la intensidad, fines de semana y feriados.   **Reglas de negocio:** | | |

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como estudiante.

Quiero pagar un curso.

Para poder Iniciar las clases.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 4 | **Usuario**: Estudiante. | |
| **Nombre historia**: Pagar un curso. | | |
| **Prioridad en negocio:**  Alta | | **Riesgo en desarrollo:**  Medio |
| **Puntos estimados**: 10 pts. | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: Me gustaría hacer los pagos de cada curso que matricule. | | |
| **Criterios de aceptación**:   * Me gustaría tener un botón donde pueda dar clic y hacer el pago del curso de mi interés. * Con el número de identificación podré buscar en esta sección el curso que he matriculado y pagarlo. * Si la búsqueda es satisfactoria, se cargará el monto a cancelar del curso, tener en cuenta que este campo debe de ser calculado, ya que se necesita saber la intensidad con la que se estudiará y el precio por hora (2000). Además, se mostrarán la cantidad de horas a pagar, el nombre del curso, * Si todavía no he pagado el curso, se debe de mostrar en estado pendiente, cuando realizo el pago debe de cambiar al estado cancelado.   **Tareas asignadas:**   * Consultar en la tabla de matrícula los datos del estudiante con el respectivo número de cédula, en estos datos se debe de tomar en cuenta el nombre del estudiante, las horas totales de curso, el precio final (este debe ser calculado). * Crear un trigger que cambie el estado de matrícula de pendiente a cancelado.   **Reglas de negocio:** | | |

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como estudiante

Quiero agendar clase sincrónica

Para poder Recibir las clases matriculadas.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 5 | **Usuario**: Estudiante | |
| **Nombre historia**: Agendar clase sincrónica. | | |
| **Prioridad en negocio**:  Alta | | **Riesgo en desarrollo**:  Medio |
| **Puntos estimados**: 10 puntos | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: Me gustaría poder agendar una clase sincrónica en el horario de mi conveniencia. | | |
| **Criterios de aceptación**:   * En el sistema se deben mostrar las siguientes opciones para agendar correctamente una clase:  1. Fecha de la clase (No se imparte clase los días feriados o fines de semana). 2. Hora (debe de cumplir el horario del instituto). 3. Profesor asignado (Generado automáticamente por el sistema). Para esto se necesita que el estudiante seleccione todos los datos referentes a la clase sincrónica y sabiendo el horario se verificará el profesor disponible en ese horario. Para el estudiante será visible el nombre del profesor.   **Tareas asignadas:**   * Crear un procedimiento almacenado que asigne un profesor a la clase, debe de tener presente que el profesor tiene que estar libre en ese horario, debe estar capacitado para dar el curso correspondiente y se tiene que tomar en cuenta el horario laboral del profesor. Finalmente, después de haber realizado el procedimiento, se mostrará el nombre completo del profesor asignado.   **Reglas de negocio:**   * Las clases podrán ser agendadas únicamente de lunes a viernes de 8 am a 10pm. * No se impartirá clases los días feriados. | | |

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como asistente administrativa

Quiero administrar los datos del estudiante.

Para poder mantener actualizada nuestra base de datos.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 6 | **Usuario**: Asistente administrativa. | |
| **Nombre historia**: Administrar los datos del estudiante. | | |
| **Prioridad en negocio**:  Alta | | **Riesgo en desarrollo**:  Medio |
| **Puntos estimados**: 10 puntos | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: | | |
| **Criterios de aceptación**:   * Consultar los estudiantes que están registrados en la base de datos del instituto. * Me gustaría poder filtrar las búsquedas por (Matriculados, Egresados, inactivos). * Cuando eliminamos los datos del estudiante, se debe hacer un borrado lógico.   **Tareas asignadas:**  **Reglas de negocio:**   * No se podrá eliminar un estudiante completamente del sistema, la eliminación será lógica. | | |

Dibujo de personaje amarillo

Descripción generada automáticamente con confianza baja

Como contador

Quiero Administrar los pagos de los estudiantes.

Para poder llevar un control los pagos de los cursos.

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 7 | **Usuario**: contador. | |
| **Nombre historia**: Administrar los pagos del estudiante. | | |
| **Prioridad en negocio**:  Alta | | **Riesgo en desarrollo**:  Medio |
| **Puntos estimados**: 10 puntos | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: | | |
| **Criterios de aceptación**:   * Consultar los datos de los estudiantes, esta consulta me podrá filtrar los datos de los pagos como (Pendiente, Cancelado). Cada vez que haga esta consulta se debe de generar un total ya sea de pendiente o de cancelado. * Me gustaría tener el resultado de la suma de todos los pagos realizados en el rango de dos fechas que yo ingrese.   **Tareas asignadas:**   * Crear un botón para hacer la búsqueda de la consulta seleccionada. * Hacer la selección de las opciones de consulta desplegables, estas opciones serán “Pendiente” y “Cancelado”. * Crear las consultas para mostrar el total de la suma de los pagos realizados, ya se de pendiente o de cancelado. * Crear una consulta que reciba una fecha de inicio y una fecha de final, debe de mostrar el total de pagos realizados, se debe habilitar unas opciones para elegir pendiente o cancelados.   **Reglas de negocio:**  Dibujo de personaje amarillo  Descripción generada automáticamente con confianza baja  Como estudiante.  Quiero modificar la fecha de las clases sincrónicas.  Para poder eventual mente posponer mis clases. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de Usuario** | | |
| **Número**: 8 | **Usuario**: Estudiante. | |
| **Nombre historia**: Modificar la fecha de las clases sincrónicas. | | |
| **Prioridad en negocio**:  Alta | | **Riesgo en desarrollo**:  Medio |
| **Puntos estimados**: 10 puntos | | |
| **Programador responsable**: Danny Soto J. | | |
| **Descripción**: En ocasiones no podré asistir a las clases por motivos personales, en esas ocasiones me gustaría modificar esa ficha. | | |
| **Criterios de aceptación**:  **Tareas asignadas:**  **Reglas de negocio:**   * Se podrán modificar las clases únicamente con 5 horas de anticipación. | | |

# Diagrama de Clases

# Diagrama Conceptual

# Diagrama Lógico