Modelo de Diseño

Sistema VASPA

VASPA Team

Nicolás Sartini



*El propósito del Modelo de Diseño es empezar a realizar los casos de uso desarrollados durante las etapas anteriores en especial la etapa de captura de requerimientos. Es decir, tomar el Modelo de Casos de Uso y las Especificaciones Suplementarias creadas con anterioridad entre otros insumos y generar un modelo de diseño que pueda ser usado por los desarrolladores durante la etapa de implementación.*



Tabla de contenido

[Introducción 4](#_Toc32150710)

[Propósito 4](#_Toc32150711)

[Definiciones, siglas y abreviaturas. 4](#_Toc32150712)

[Referencias 4](#_Toc32150713)

[Visión general 5](#_Toc32150714)

[Diseño de Objetos 5](#_Toc32150715)

[Carrera 5](#_Toc32150716)

[Plan 6](#_Toc32150717)

[Asignatura 6](#_Toc32150718)

[Programa 7](#_Toc32150719)

[Profesor 11](#_Toc32150720)

[Departamento 12](#_Toc32150721)

[Diagramas 14](#_Toc32150722)

[Diagrama de Clases 14](#_Toc32150723)

[Diagrama de Secuencia 15](#_Toc32150724)

Modelo de Diseño

Introducción

El Modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tiene impacto en el sistema a desarrollar.

Propósito

Los propósitos de este documento son:

* Adquirir una compresión en profundidad de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de interfaz de usuario, etc.
* Crear una entrada apropiada y un punto de partida para actividades de implementación subsiguientes capturando los requisitos o subsistemas individuales, interfaces y clases.
* Ser capaces de descomponer los trabajos de implementación en partes más manejables. Esto resulta útil en los casos en los que la descomposición no puede ser realizada en la captura de requisitos o análisis.

Definiciones, siglas y abreviaturas.

Las palabras desconocidas o ambiguas que son utilizadas en este documento se encuentran definidas en el documento “Glosario”, por ende se debe remitir al mismo para encontrar términos importantes de este documento.

Referencias

* Especificación de Requerimientos.
* Modelo de Casos de Uso.

Visión general

* Las secciones subsiguientes se componen de:
* **Diseño de objetos**, donde se describen brevemente los objetos a implementar, se enumeran sus atributos y métodos.
* **Diagramas**, finalmente la sección de diagramas, presenta el diagrama de clases y de secuencia.

Diseño de Objetos

Carrera

* Descripción: Se trata de un objeto que representa una carrera perteneciente a la Unpa - Uarg.
* Objetivo: Almacenar los atributos de la carrera creada.
* Atributos:

- código

- nombre

* Métodos:

- alta

- modificacion

- baja

- buscar

GETTER´s

- getCodigo

- getNombre

- getColeccion

SETTER´s

- setCodigo

- setNombre

Plan

* Descripción: Se trata de un objeto que representa un plan de estudios perteneciente a una carrera de la Unpa - Uarg.
* Objetivo: Almacenar los atributos del plan creado.
* Atributos:

- código

- anio

* Métodos:

- alta

- modificacion

- baja

- buscar

GETTER´s

- getCodigo

- getAnio

- getColeccion

SETTER´s

- setCodigo

- setAnio

Asignatura

* Descripción: Se trata de un objeto que representa una asignatura de un plan de estudios perteneciente a una carrera de la Unpa - Uarg.
* Objetivo: Almacenar los atributos de la asignatura creada.
* Atributos:

- código

- nombre

- departamento

- contenidosMinimos

* Métodos:

- alta

- modificacion

- baja

- buscar

GETTER´s

- getCodigo

- getNombre

- getDepartamento

- getContenidosMinimos

- getColeccion

SETTER´s

- setCodigo

- setNombre

- setDepartamento

- setContenidosMinimos

Programa

* Descripción: Se trata de un objeto que representa un programa de asignatura de un plan de estudios perteneciente a una carrera de la Unpa - Uarg.
* Objetivo: Almacenar los atributos del programa creado.
* Atributos:

- id

- anio

- anioCarrera

- horasTeoria

- horasPractica

- horasOtros

- regimenCursada

- observacionesHoras

- observacionesCursada

- docentesTeoria

- docentesPractica

- fundamentacion

- objetivosGenerales

- organizacionContenidos

- criteriosEvaluacion

- metodologiaPresencial

- regularizacionPresencial

- aprobacionPresencial

- metodologiaSatep

- regularizacionSatep

- aprobacionSatep

- metodologiaLibre

- aprobacionLibre

- ubicacion

* Métodos:

- alta

- modificacion

- baja

- buscar

GETTER´s

- getId

- getAnio

- getAnioCarrera

- getHorasTeoria

- getHorasPractica

- getHorasOtros

- getRegimenCursada

- getObservacionesHoras

- getObservacionesCursada

- getDocentesTeoria

- getDocentesPractica

- getFundamentacion

- getObjetivosGenerales

- getOrganizacionContenidos

- getCriteriosEvaluacion

- getMetodologiaPresencial

- getRegularizacionPresencial

- getAprobacionPresencial

- getMetodologiaSatep

- getRegularizacionSatep

- getAprobacionSatep

- getMetodologiaLibre

- getAprobacionLibre

- getUbicacion

- getColeccion

SETTER´s

- setId

- setAnio

- setAnioCarrera

- setHorasTeoria

- setHorasPractica

- setHorasOtros

- setRegimenCursada

- setObservacionesHoras

- setObservacionesCursada

- setDocentesTeoria

- setDocentesPractica

- setFundamentacion

- setObjetivosGenerales

- setOrganizacionContenidos

- setCriteriosEvaluacion

- setMetodologiaPresencial

- setRegularizacionPresencial

- setAprobacionPresencial

- setMetodologiaSatep

- setRegularizacionSatep

- setAprobacionSatep

- setMetodologiaLibre

- setAprobacionLibre

- setUbicacion

Profesor

* Descripción: Se trata de un objeto que representa un profesor responsable de programas pertenecientes a una carrera de la Unpa - Uarg.
* Objetivo: Almacenar los atributos del profesor creado.
* Atributos:

- dni

- nombre

- apellido

- email

- cargo

- categoria

- id

* Métodos:

- alta

- modificacion

- baja

- buscar

GETTER´s

- getDni

- getNombre

- getApellido

- getEmail

- getCargo

- getCategoria

- getId

- getColeccion

SETTER´s

- setDni

- setNombre

- setApellido

- setEmail

- setCargo

- setCategoria

- setId

Departamento

* Descripción: Se trata de un objeto que representa un Departamento de la Unpa - Uarg.
* Objetivo: Almacenar los atributos del departamento creado.
* Atributos:

- nombre

- id

* Métodos:

- alta

- modificacion

- baja

- buscar

GETTER´s

- getNombre

- getId

- getColeccion

SETTER´s

- setNombre

- setId

# Diagramas

Diagrama de Clases

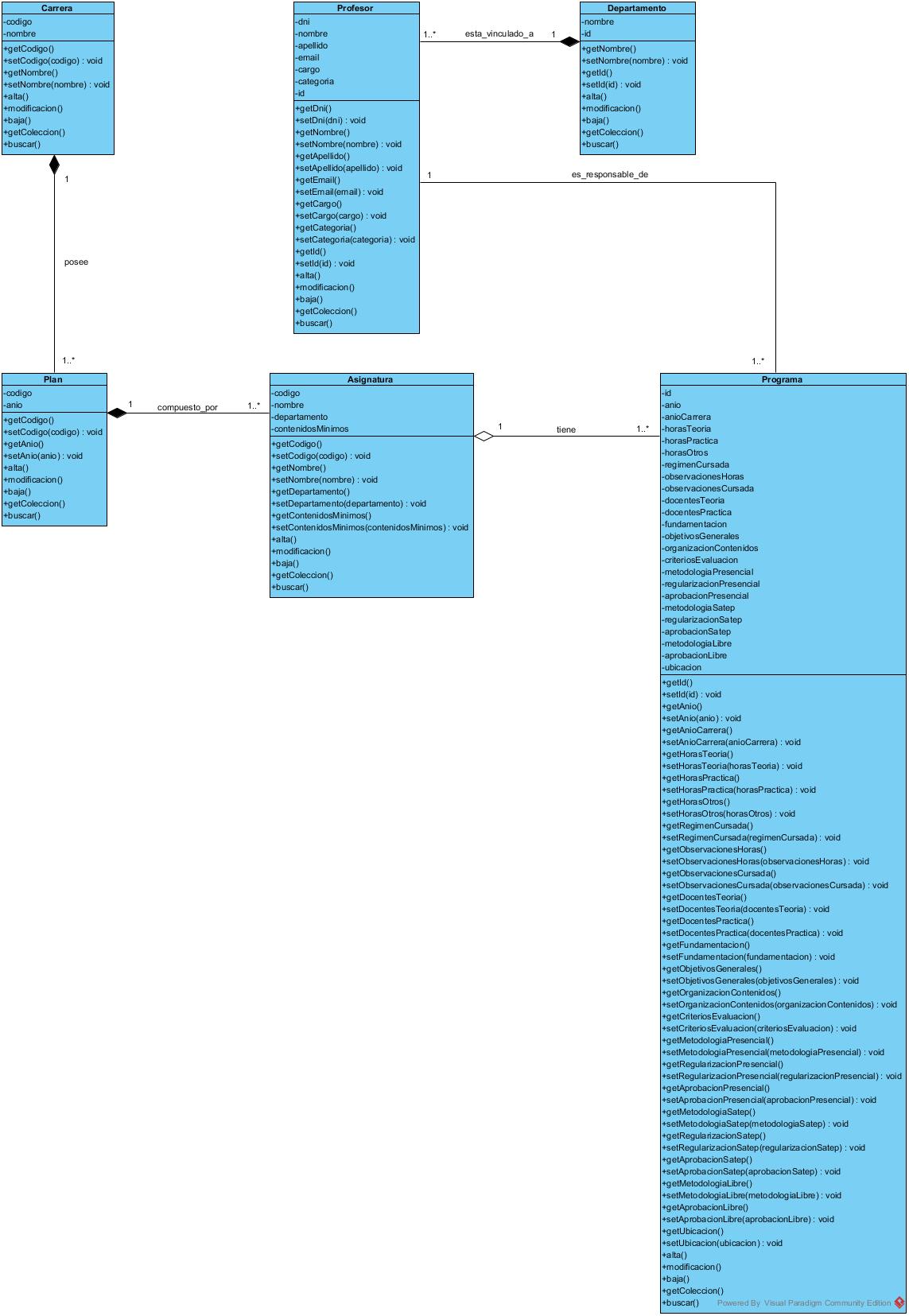


Diagrama de Secuencia

Los diagramas de secuencia correspondientes a cada caso de uso del sistema se encuentran realizados en los respectivos documentos de "Especificación" de cada caso de uso individual, por ende se debe remitir a los mismos para visualizar dichos diagramas.