

---

ISO/IEC/IEEE 42010:2011

Descripción de la Arquitectura  
para el sistema  
**Scooby Attendance**

Realizado por:  
Ruelas Quenaya, Merisabel  
Soto Begazo, Juan Manuel  
Villaroel Ramos, Fiorela

UNSA  
18 de agosto de 2022

---

# Índice general

<b>1. Introducción</b>	<b>II</b>
1.1. Propósito . . . . .	III
1.2. Alcance . . . . .	III
1.3. Definiciones, acrónimos y abreviaturas . . . . .	III
<b>2. Representación arquitectónica</b>	<b>IV</b>
<b>3. Objetivos y limitaciones de la arquitectura</b>	<b>VI</b>
3.1. Requisitos Funcionales . . . . .	VI
3.2. Requisitos No-Funcionales . . . . .	VII
<b>4. Vista Escenario</b>	<b>IX</b>
<b>5. Vista de Implementación</b>	<b>X</b>
5.1. Visión de Datos . . . . .	XI
<b>6. Vista de Implantación</b>	<b>XIII</b>
<b>7. Vista de Proceso: Diagrama de Componentes</b>	<b>XV</b>

# Capítulo 1

## Introducción

Para obtener una visión arquitectónica del sistema de gestión escolar de nombre SCOOBY ATTENDANCE, este documento viene a dilucidar los diversos aspectos del sistema. El objetivo de este documento es transmitir las opciones dadas en términos de arquitectura que se adoptarán para el desarrollo del sistema.

El propósito de este documento es clasificar claramente la arquitectura, que se estableció para el desarrollo de SCOOBY ATTENDANCE, el enfoque es seguir el desarrollo en la arquitectura estándar de tres capas con MVC (Model View Controller), donde la arquitectura de tres capas se divide en, capa de presentación que es la interacción directamente con el usuario, es a través de ella que se realizan las solicitudes como consultas y otros, El MVC (Modelo Vista Controlador) viene con un Modelo que se ocupa de todas las reglas de negocio, una Vista con su presentación visual que se conoce como la interfaz, y el Controlador que tiene el deber de controlar la comunicación entre las capas de la Vista y el Modelo.

En el documento se define lo que es sistema va a contemplar en términos arquitectónicos, especificando sus casos de uso, las vistas lógicas, las capas, así como su proceso, implementación y despliegue. Se ha prestado atención a todas las decisiones de arquitectura en este documento que serán únicas para el desarrollo del Sistema. La aplicación es un entorno WEB que tiende a utilizar una base de datos que estará dentro del entorno de implementación que es un servidor web, siguiendo principios de buenas prácticas.

## **1.1. Propósito**

El objetivo de este documento es ofrecer de forma clara las distintas visiones que poseen los modelos arquitectónicos del sistema Maruge, aportando las características necesarias para el control de sus actividades arquitectónicas, conformando así todo el procedimiento de desarrollo del sistema.

## **1.2. Alcance**

La arquitectura del proyecto está totalmente fundamentada y detallada, para proporcionar una base al equipo de desarrollo, presentando cómo se comportará el sistema. Este documento se refiere a las características identificadas en el documento de requerimiento de Software, que trata de las necesidades y reglas del negocio que el sistema debe satisfacer. Este documento contiene registros y temas importantes relacionados con la arquitectura del sistema, dando pautas y soporte completo para el uso de sus tecnologías.

## **1.3. Definiciones, acrónimos y abreviaturas**

- EPCC: Escuela Profesional de Ciencia de la Computación de la Universidad Nacional de San Agustín
- API: Conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizada por otro software como una capa de abstracción.
- SIS-SA: Sistema Scooby Attendance
- UNSA: Universidad Nacional de San Agustín
- MVC: Modelo Vista Controlador

## Capítulo 2

# Representación arquitectónica

El sistema se desarrollará siguiendo un patrón de arquitectura de tres capas (Presentación, Reglas de negocio y Datos) con el MVC (Modelo, Vista y Controlador), siguiendo las siguientes reglas, la capa de presentación es la capa que proporciona todas las interfaces de Scooby Attendance al usuario, esta capa está directamente relacionada con la capa de Vista, es donde el usuario obtiene las respuestas de la capa de Negocio, que se utiliza para controlar las reglas de negocio del sistema.

Esta capa contiene los controladores de la aplicación, estos controladores son los que pasan las reglas de negocio solicitadas a la capa de implementación que está relacionada con el modelo, que se encarga de mantener la persistencia con la base de datos.

# MODELO - VISTA - CONTROLADOR

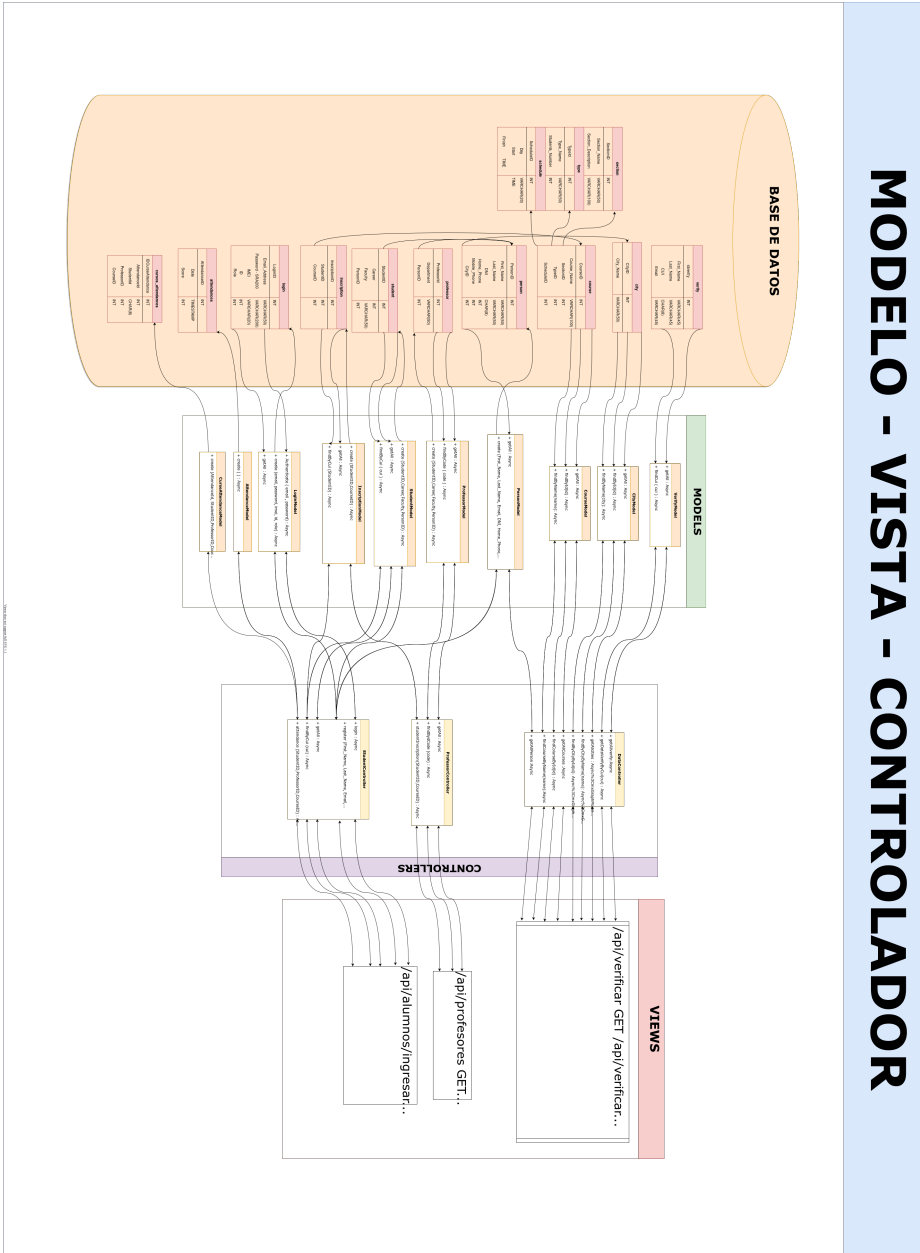


Figura 2.1: Representación arquitectónica

## Capítulo 3

# Objetivos y limitaciones de la arquitectura

Todos los requisitos no funcionales están explícitos en las Especificaciones de Requisitos documentadas son los que definen los límites del sistema y afectan directamente a la arquitectura.

### 3.1. Requisitos Funcionales

- Iniciar sesión.
- Cerrar sesión.
- Añadir asignaturas.
- Visualizar las asistencias.
- Visualizar el horario.
- Agregar estudiantes al curso.
- Gestionar la asistencia de los estudiantes a las clases.
- Registrar a los nuevos alumnos.
- Registrar la asistencia de los alumnos.
- Registrar los detalles del curso y la información de la asignatura.

## 3.2. Requisitos No-Funcionales

### ■ Mantenibilidad

- El sistema debe disponer de una información fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible
- La interfaz debe estar complementada con un buen sistema de ayuda (la administración puede recaer en personal con poca experiencia en el uso de aplicaciones informáticas).

### ■ Requisitos de rendimiento

- Garantizar que el diseño de las consultas u otro proceso no afecte el desempeño de la base de datos, ni considerablemente el tráfico de la red.

### ■ Seguridad

- En este sistema, la autenticación del usuario se hará mediante el inicio de sesión por nombre de usuario y contraseña y se clasificará por tipo de usuario. Los usuarios tendrán acceso al sistema según los permisos clasificados para ese tipo de usuario.
- Garantizar la confiabilidad, la seguridad y el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanente y simultáneamente, sin que se afecte el tiempo de respuesta.
- Garantizar la seguridad del sistema con respecto a la información y datos que se manejan tales sean información personal y contraseñas.

### ■ Fiabilidad

- El sistema tiene una interfaz coherente para que el sistema sea fácil de usar y en la interfaz del sistema se utilizan botones y formularios para introducir datos relacionados con un módulo específico.

### ■ Disponibilidad

- La disponibilidad del sistema debe ser continua con un nivel de servicio para los usuarios de 7 días por 24 horas, garantizando un esquema adecuado que permita la posible falla en cualquiera de sus componentes, contar con una contingencia, generación de alarmas.



## ■ **Portabilidad**

- La compatibilidad con los dispositivos y la capacidad de respuesta, garantizan que el sitio se ejecute de forma uniforme en todos los dispositivos sin cambiar el estilo de la página web.

## Capítulo 4

# Vista Escenario

En la figura 4.1 "Diagrama de casos de uso", se hace una representación de todos los casos de uso a los que responde el sistema.

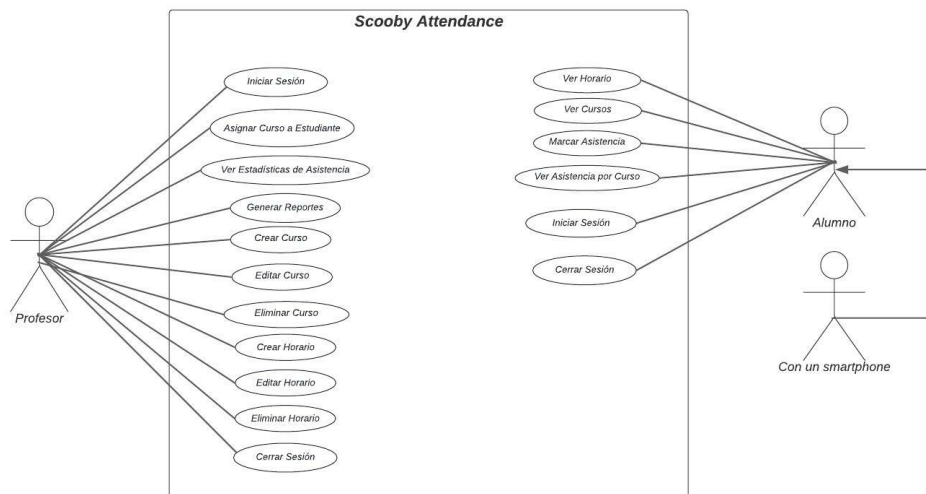


Figura 4.1: Casos de Uso

## Capítulo 5

# Vista de Implementación

Scooby Attendance tendrá 3 niveles de acceso, y cada uno de ellos se divide en dos niveles arquitectónicos el primero sigue el modelo cliente-servidor, que sigue las reglas de presentación, reglas de negocio y datos y el segundo niveles de arquitectura es MVC, que se basará en los estándares de buenas prácticas, siguiendo las reglas de rutas y las tres capas vista, modelo y controlador. Las rutas responsables de dirigir a los usuarios a la vista solicitada a través de su comunicación con el control que está siempre en comunicación con el modelo que es responsable persistencia con la base de datos.

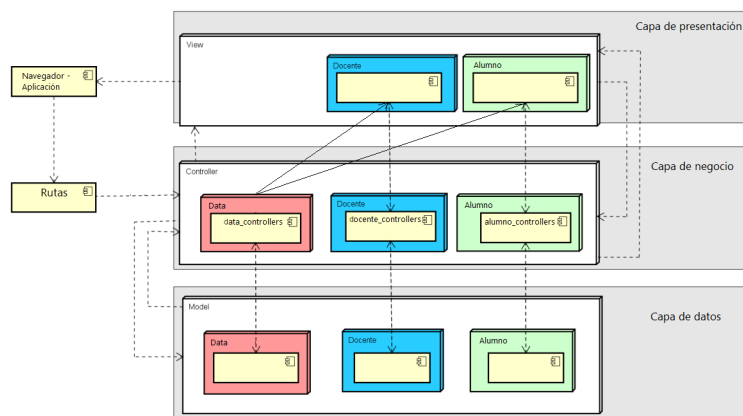


Figura 5.1: Diagrama de Clases

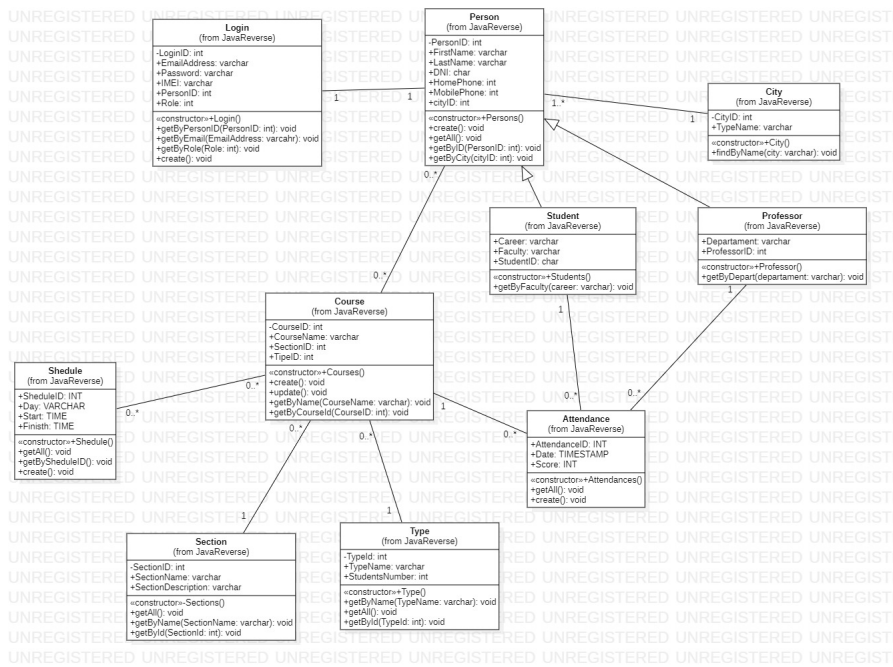


Figura 5.2: Diagrama de Clases

## 5.1. Visión de Datos

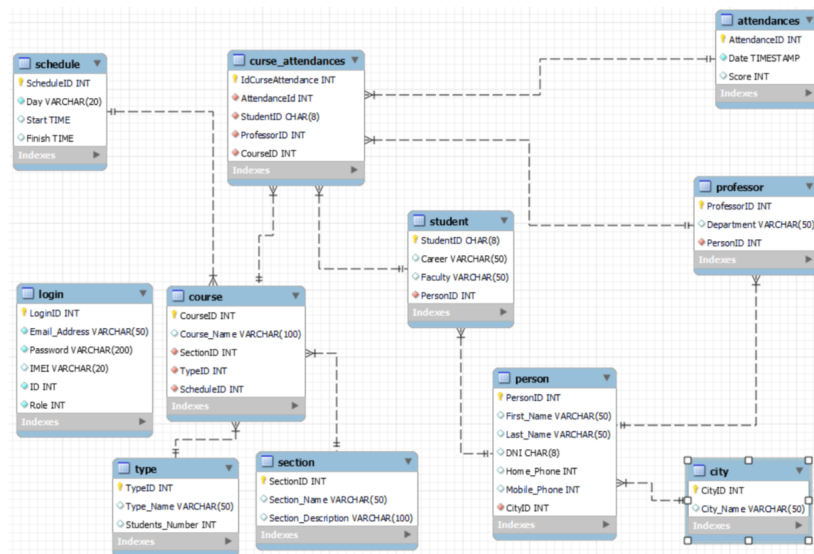


Figura 5.3: Diagrama de Clases

## Capítulo 6

# Vista de Implantación

Scooby Attendance es una aplicación web que sigue un patrón Cliente-Servidor, su despliegue se compone de tres fases: Cliente, que accede a la aplicación; Servidor de Aplicaciones, que a través de una petición HTTP/HTTPS interactúa con Scooby Attendance y sus reglas de negocio, accediendo a los recursos de Scooby Attendance y la capa de Base de Datos, que es el lugar donde se almacenan todos los datos.

Scooby Attendance requiere un procesador mínimo de 1Ghz, 1GB de Ram, 16GB de espacio en disco y una conexión a Internet de 1 megabit.

Para el desarrollador de aplicaciones se requiere un procesador Intel Xeon Dual Core 2.3 o equivalente, 4GB de Ram y 50GB de espacio en disco.

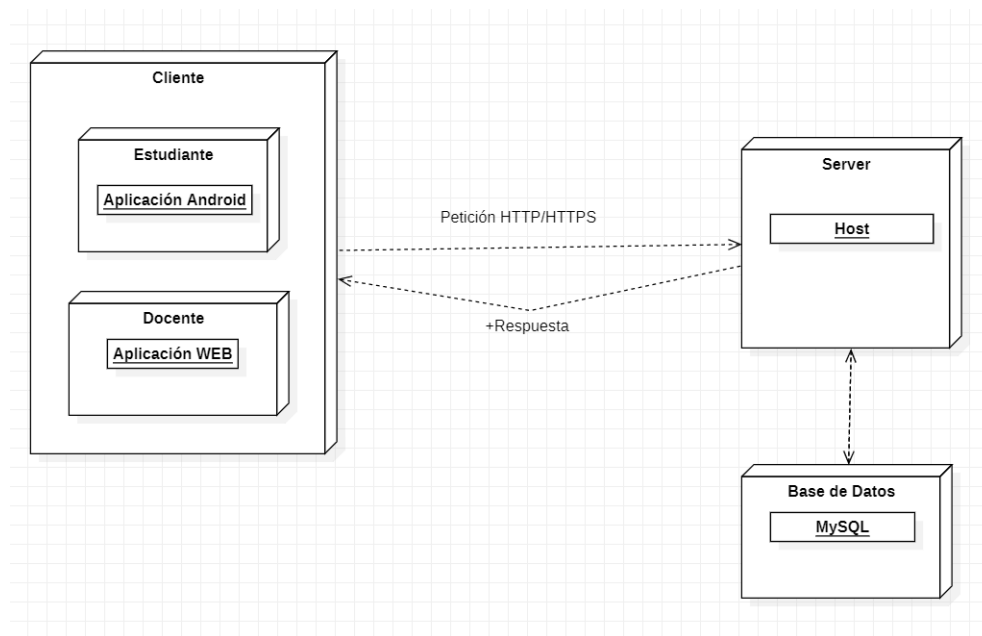


Figura 6.1: Diagrama de Despliegue

## **Capítulo 7**

# **Vista de Proceso: Diagrama de Componentes**



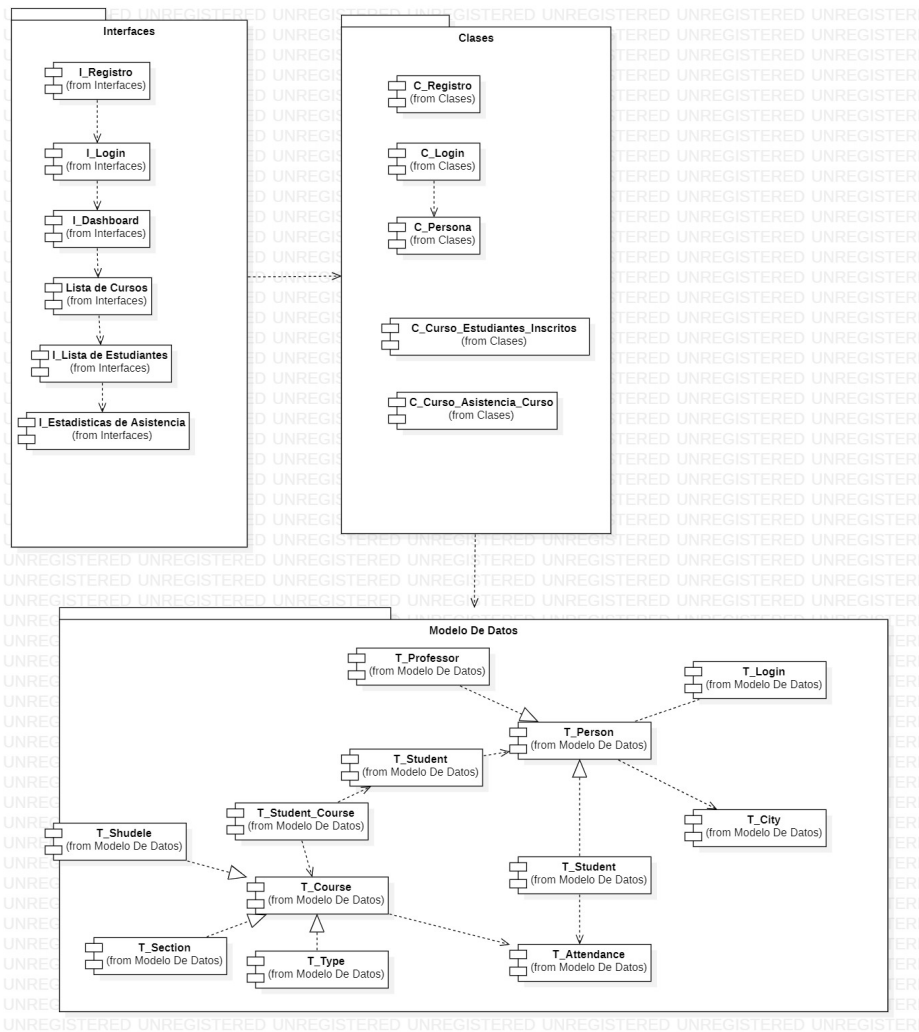


Figura 7.1: Diagrama General de Componentes

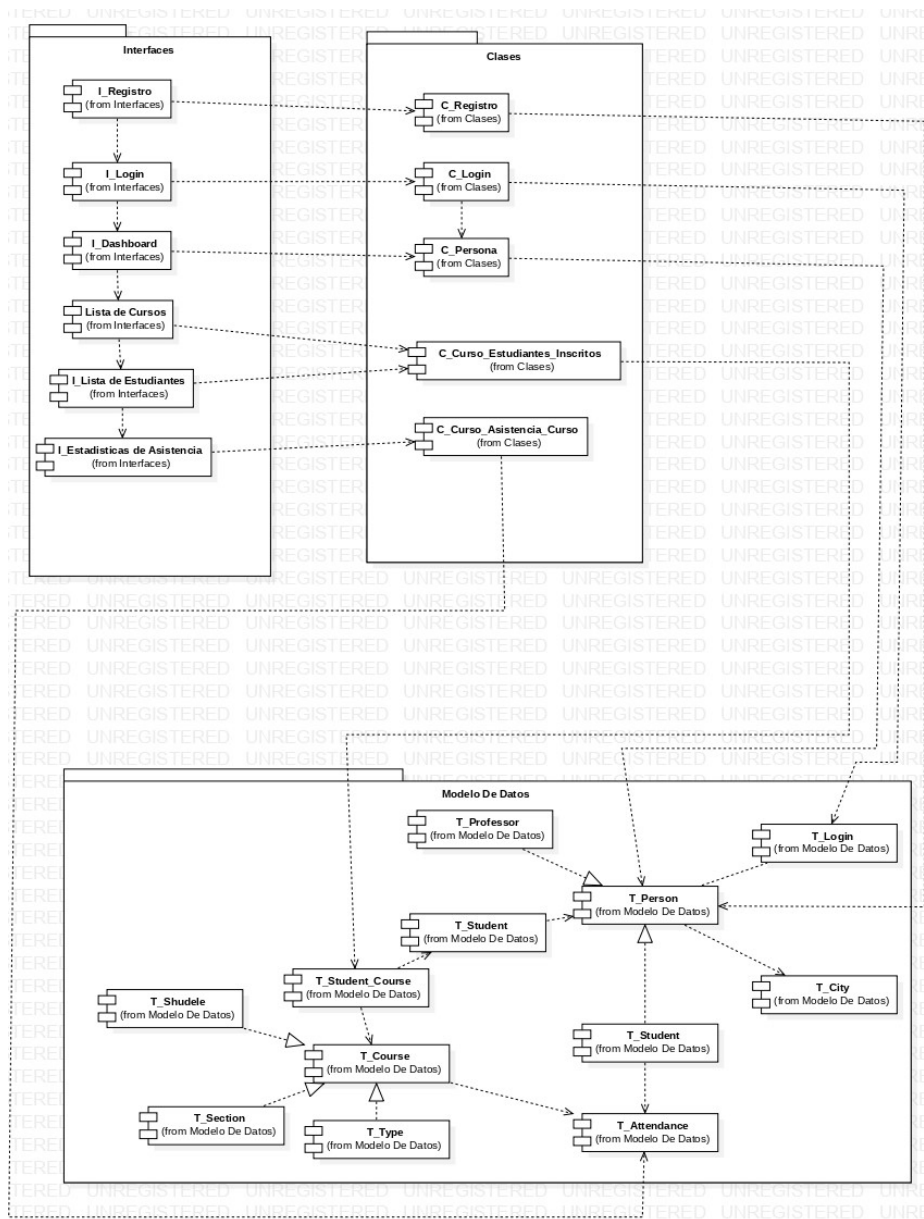


Figura 7.2: Diagrama General de Componentes para Usuario del Tipo Estudiante

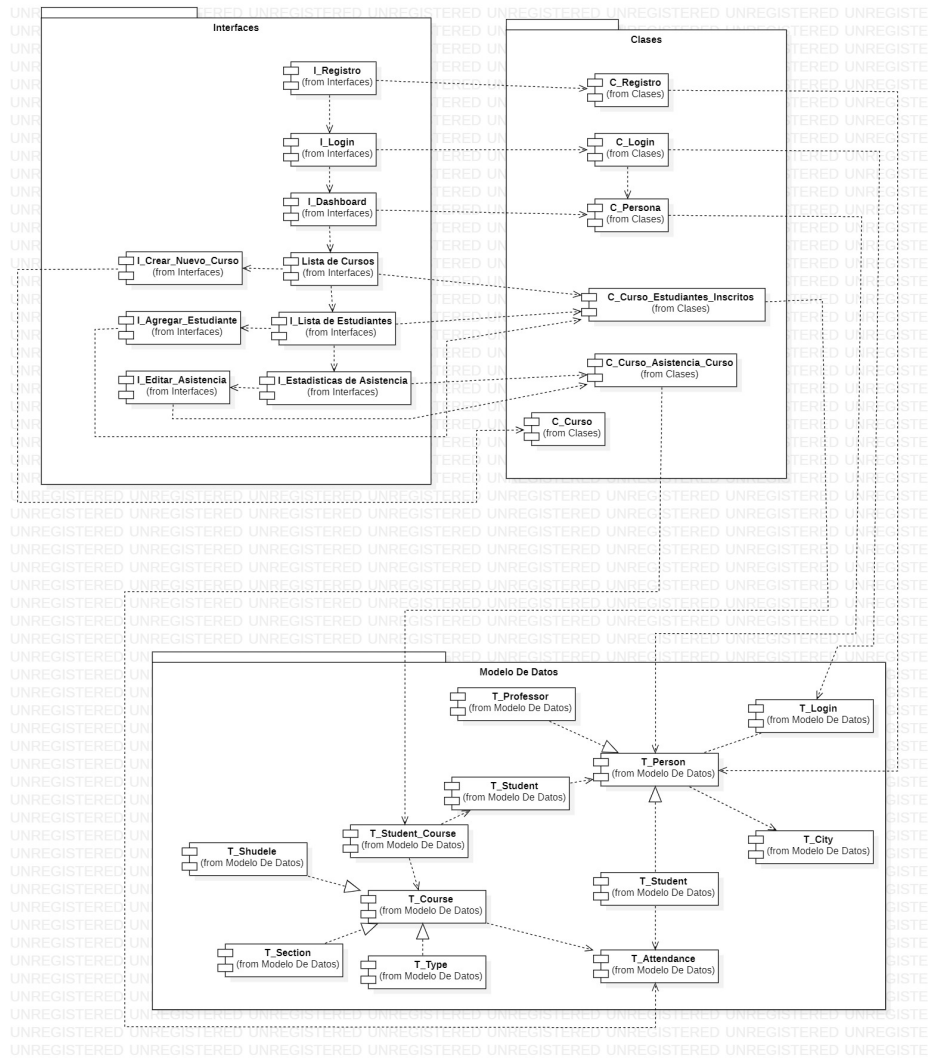


Figura 7.3: Diagrama General de Componentes para Usuario del Tipo Profesor