

Universidad de las Fuerzas Armadas ESPE

**Departamento de Ciencias de la Computación  
(DCC0)**

Carrera de Ingeniería Tecnología de la  
Información  
(En línea)

**NRC 5451: Ingeniería de Software II**

**Integrantes del Grupo 5:**

Karla Cazares, Fernando Carrera, Lizeth Iza, Erick  
Maldonado, Jaime Rojas.

**Docente:**

Dr. Efraín Rodrigo Fonseca Carrera

**06 de junio del 2022**

## Introducción

Actualmente es primordial para cualquier institución la utilización de herramientas de software, en especial las aplicaciones web, ya que permite resguardo de la información de manera segura y el fácil acceso a la información desde cualquier punto de acceso de una red.

El propósito de este trabajo es presentar la arquitectura de una aplicación web, la cual será administrada para dar seguimiento en una institución educativa, con sus usuarios y el responsable para que se atiendan reportes, además el sistema arrojará estadísticas cada mes de los reportes, también la aplicación tendrá la gestión total de los departamentos y accesos de usuarios a la aplicación.

## Objetivos

### Objetivo General:

- Proponer una arquitectura de software para un caso práctico de estudio, considerando la evolución del software.

### Objetivos Específicos:

- Plantear la problemática y requerimientos del sistema.
- Diseñar la arquitectura del sistema utilizando el patrón MVC.
- Describir la metodología, métodos, técnicas usadas en la arquitectura.

## Desarrollo

1. **Plantee la problemática a ser abordada dando a conocer las características de su contexto y los requerimientos generales del sistema, considerando una posible evolución de los mismos.**

## Problemática

La Unidad Educativa requiere una aplicación web que le permita gestionar todos los procesos académicos, administrativos, y reportería enmarcados en el contexto de los lineamientos del mineduc.

### **Requerimientos generales:**

- Control de usuarios
- Gestión de calificaciones
- Gestión de asignaturas
- Gestión de periodos
- Gestión de activos fijos

- Reportería en General

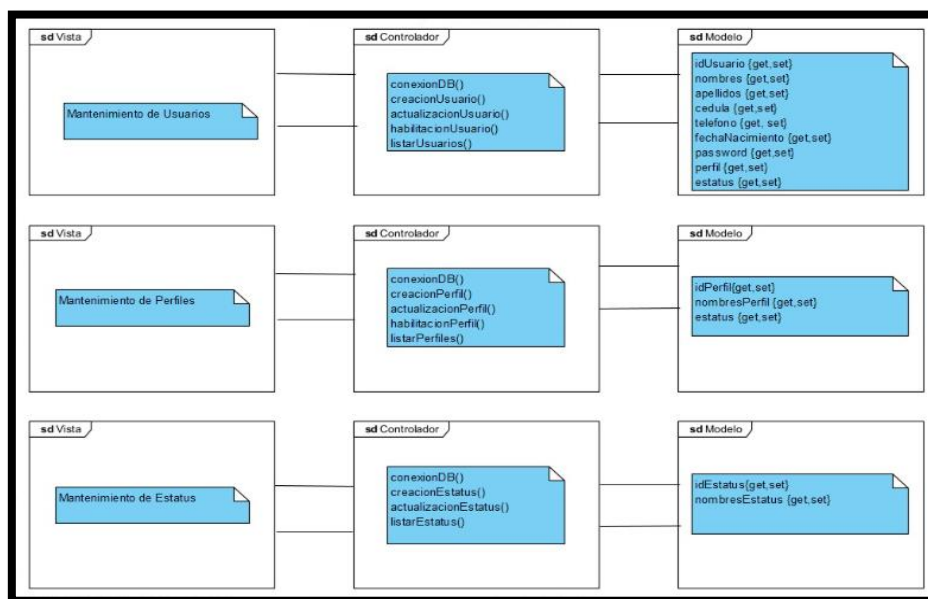
Cada uno de los requerimientos serán desarrollados tratando de ser compatibles con la aplicación Carmenta del mineduc. Para la interfaz de software se utilizará los siguientes componentes de software:

- **Base de Datos:** MySQL
- **Sistema Operativo:** Windows 10
- **IDE de desarrollo:** Visual Studio
- **Lenguaje de Programación:** PHP

2. **Diseñe la arquitectura del sistema utilizando de preferencia el patrón MVC, justificando su pertinencia y todos los elementos tomados en consideración.**

En la siguiente arquitectura propuesta se hará uso del patrón MVC

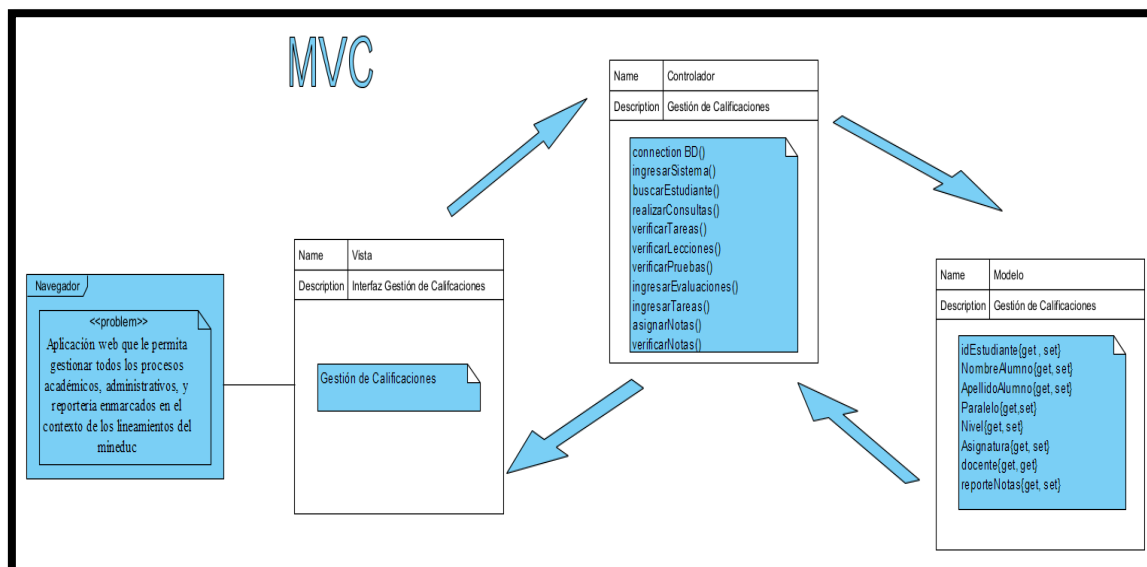
- **Control de usuarios**



*Ilustración 1: Control de Usuarios*

En el modelo MVC para el control de usuarios es pertinente ocupar esta arquitectura debido a que los diferentes atributos de los modelos deben ser utilizados por el resto de funcionalidades del sistema tanto en los apartados de get que permitirán manejar una información fiable en todo el entorno, de igual forma al utilizar la metodología SCRUM vamos conociendo de primera mano cómo nuestro desarrollo se está adaptando a las necesidades del cliente y logra obtener los resultados esperados.

- Gestión de calificaciones



*Ilustración 2: Gestión de Calificaciones*

Con respecto al modelo el sistema debe permitir al docente gestionar la información de cada estudiante correspondiente a nombreAlumno, apellidoAlumno, paralelo, nivel, asignatura, docente y reporte de notas que utilizan getters and setters para almacenar los diferentes atributos individuales para cada objeto o una clase de ser el caso.

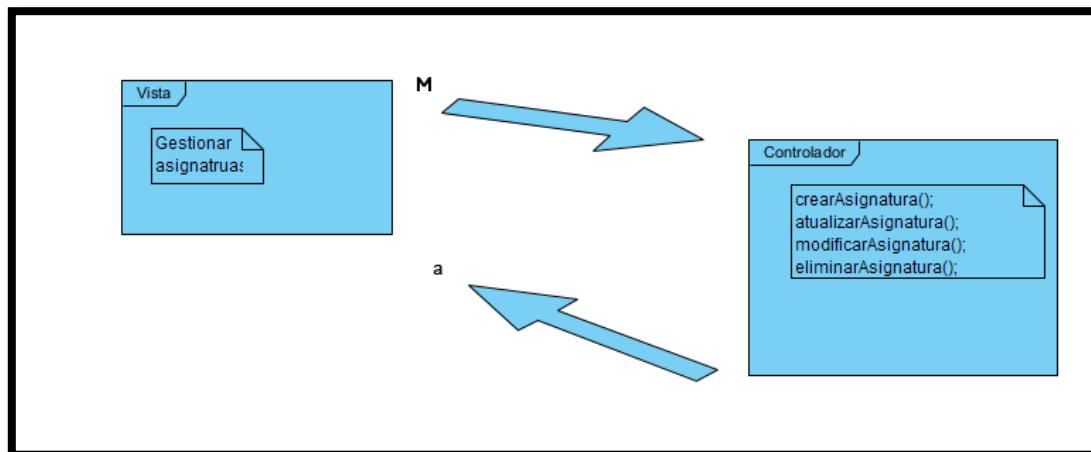
Ante estos parámetros se puede facilitar para la etapa del desarrollo ya que cada componente tiene dependencia lógica y física respecto con las tareas o actividades del docente asignado para este proceso, es decir, el programador se puede especializar en el diseño y sobre todo organizar de forma eficiente su trabajo.

En el modelo de Controlador intervienen los métodos o funciones que se generan en cada operación para que el docente asignado pueda realizar las tareas esenciales para cumplir con su trabajo tales como ingresarSistema(), buscarEstudiante(), realizarConsultas(), verificarTareas(), verificarLecciones(), verificarPruebas(), ingresarEvaluaciones(), ingresarTareas(), asignarNotas(), verificarNotas().

Todo esto partiendo del modelo controlador a partir del gestor de base de datos que el aplicativo tendrá implementado para este proceso.

Este conjunto de procesos tiene un fin en general que es mostrar mediante el modelo vista una interfaz al usuario donde se establecerán cada una de las ventanas y mapas de navegación según el requerimiento funcional del aplicativo, siendo de gran ayuda para la interacción entre usuarios administradores y el usuario final.

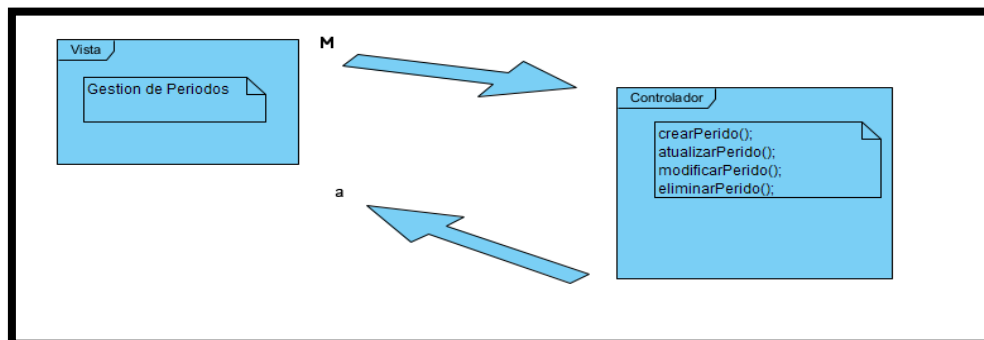
- **Gestión de asignaturas**



*Ilustración 3: Gestión de Asignatura*

La gestión de las funciones requeridas por un usuario para crear y gestionar los datos lo hacemos mediante CRUD, aquí las operaciones están específicamente adaptadas a los requisitos del sistema y de usuario

- **Gestión de periodos**



*Ilustración 4: Gestión de Periodos*

De la misma forma funciona con la gestión de periodos, aquí las operaciones están específicamente adaptadas a los requisitos del sistema y de usuario

- **Gestión de activos fijos**

Mediante la arquitectura MVC se ha realizado el gestión de activos fijos el cual le va a permitir al área administrativa gestionar los datos de la institución como el NombreInstitucion, Ciudad, Ubicación que hacen uso de los get, set, también gestionar los datos de las autoridades, como nombre, apellidos y cargo y por último llevar el registro del mantenimientos de categorías de los equipos, permitiéndole crear, modificar, eliminar equipos informáticos de la institución.

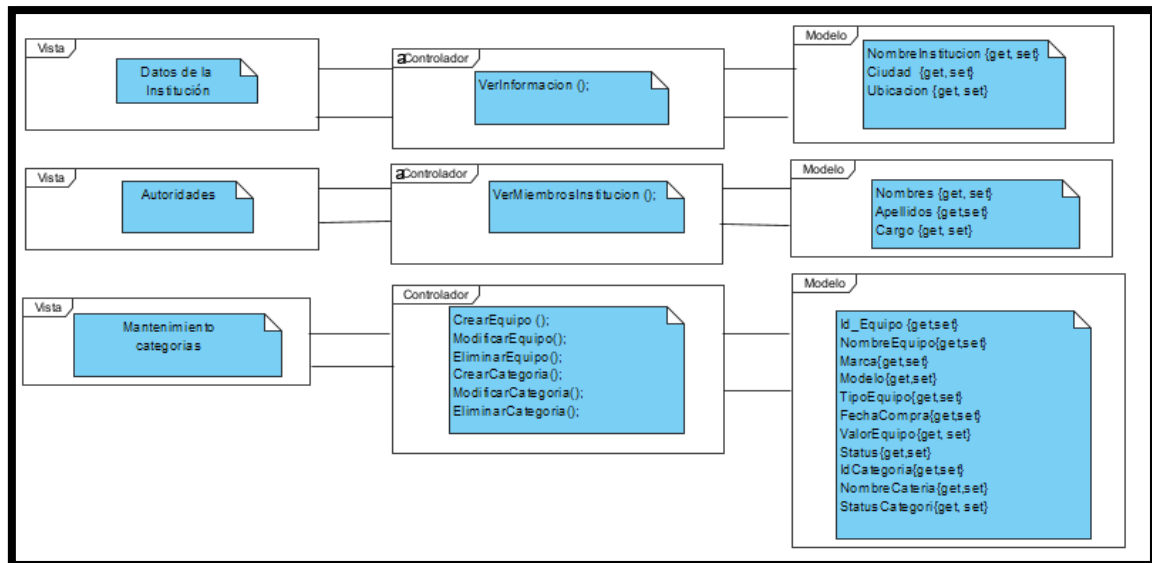


Ilustración 5: Gestión de activos fijos

### ● Reporteria en General

Mediante esta implementación de esta arquitectura se mostrará los reportes designados de los usuarios para poder atenderlos, donde también pueden ver el historial de reportes, también al igual que los otros usuarios de la aplicación puede ver la tabla de reportes totales y estadísticas.

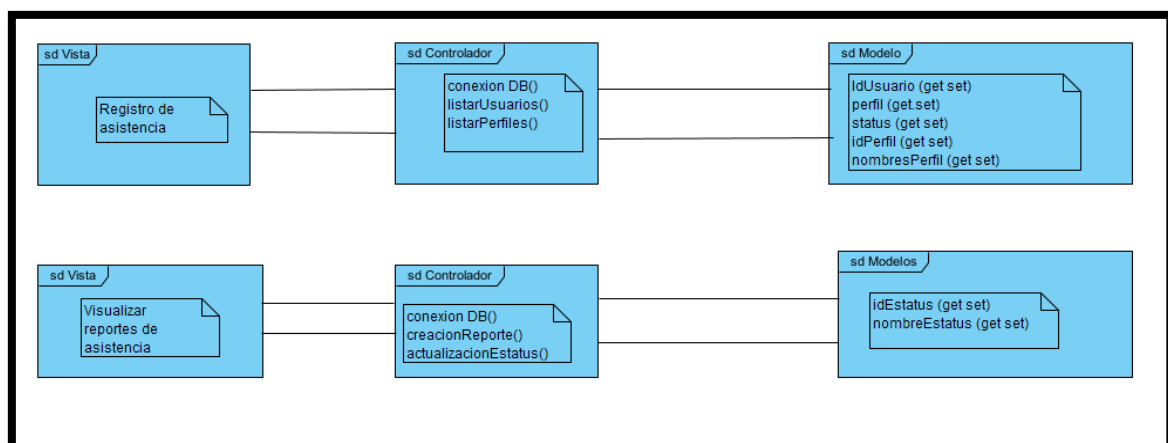


Ilustración 6: Gestión de reporteria

## Conclusiones

- El modelo MVC en desarrollo de software genera constantes cambios desde su estructura hasta su etapa final de implementación, debido a que en base a las necesidades de cada aplicativo el desarrollador puede proponer la metodología y las técnicas para plasmar de forma adecuada las necesidades que tendrá el aplicativo.
- Mediante la arquitectura del sistema usando el patrón de diseño MVC permite la separación clara de donde se ubica cada tipo de lógica de manera ordenada y facilitando el mantenimiento y la escalabilidad del sistema.
- El uso de la metodología SCRUM nos permitirá tener de primera mano los resultados de las pruebas realizadas por el cliente final y de esta manera en lo posible generar los cambios necesarios sin afectar el resto de los módulos.
- Mediante esta implementación de nuestro proyecto se cumplieron las expectativas y estamos considerando nuevas propuestas que enriquecerán aún más el sistema a desarrollar.

## Anexos:

- **Enlace del video:**

<https://youtu.be/pAqNBtHAXbA>

## Bibliografía

- Edgar Bernal. (06 de 2010). *CIMAT*. Obtenido de <https://cimat.repositorioinstitucional.mx/jspui/bitstream/1008/206/2/TE%20326.pdf>
- R.Jimenez, F. M. (s.f.). Utilización de la arquitectura Modelo - Vista – Controlador (MVC). 11. Obtenido de <https://redi.uta.edu.ec/jspui/bitstream/123456789/37301/1/JIMENEZ%20RUIZ%20EDWIN%20RUBEN%20-2017.pdf>