

**Universidad de Matanzas**

**Facultad de Ciencias Técnicas**

**Departamento de Informática**



**Trabajo de Diploma en opción del Título de Ingeniero Informático.**

**Tema de Investigación: Gestión de la información para el control a clase  
en la Universidad de Matanzas**

**Autor: Yariel Ortiz Rodríguez**

**Tutor: Dr.C. Walfredo González Hernández**

**Matanzas 2023**

**Declaratoria Autoría**

Declaro ser el único autor del presente trabajo de diploma y autorizo a la Universidad de Matanzas, hacer uso con carácter exclusivo de los derechos patrimoniales del mismo en su beneficio.

Para que así conste firmo el presente a los \_\_\_\_\_ días del mes \_\_\_\_\_ del año 2023.

-----

-----

Firma del autor

Yariel Ortiz Rodríguez

Hernández

Firma del Tutor

Dr.C. Walfredo González

## **Resumen**

En la Universidad de Matanzas "Camilo Cienfuegos" se recoge toda la información referente al control a clase a través de documentos Word. Este proceso causa un mal desarrollo de las actividades referentes a este tema por no tener una adecuada coordinación.

El objetivo de este trabajo es desarrollar una aplicación web para ayudar a la gestión de la información que se maneja en el proceso de control a clase, la cual constituye una atractiva propuesta que contribuye y facilita este proceso tanto en nuestro departamento como en Universidad en general. Para el desarrollo de la solución propuesta se realiza un análisis de las principales herramientas, tecnologías y metodologías que se utilizan en la construcción de un software. El proceso estuvo guiado por el uso de las siguientes herramientas y tecnologías: Visual Paradigm como herramienta CASE, UML como lenguaje de modelado, Django como marco de trabajo y Python como lenguaje de programación. Finalmente se obtuvo un sistema informático que permite informatizar y generar la información referente.

## **Summary**

At the University of Matanzas "Camilo Cienfuegos" all the information regarding class control is collected through Word documents. This process causes a bad development of the activities related to this subject because of not having an adequate coordination.

The objective of this work is to develop a web application to help the management of the information that is handled in the process of class control, which is an attractive proposal that contributes and facilitates this process both in our department and in the University in general. For the development of the proposed solution, an analysis of the main tools, technologies and methodologies used in the construction of a software was carried out. The process was guided by the use of the following tools and technologies: Visual Paradigm as a CASE tool, UML as a modeling language, Django as a framework and Python as a programming language. Finally, a computer system was obtained that allows the computerization and generation of the related information.



## Tabla de contenido

Introducción .....	7
Hipótesis .....	8
Objeto de la investigación .....	8
Campo acción:.....	8
Objetivo general.....	8
Objetivos Específicos .....	9
Antecedentes de la investigación .....	9
Capítulo 1 .....	10
Marco teórico referencial .....	¡Error! Marcador no definido.
Efecto económico .....	¡Error! Marcador no definido.
Metodologías de desarrollo de software.....	10
Tabla: Comparación de metodologías. Fuente: (Pérez Pérez, 2012) .....	10
Metodologías ágiles de desarrollo .....	11
Tabla: Comparativa de XP y SCRUM .....	11
Metodología de desarrollo XP (Programación Extrema).....	12
Tecnologías y herramientas a utilizar .....	14
Arquitectura del software .....	14
Arquitectura Cliente-Servidor .....	14
Ventajas del Esquema Cliente/Servidor: .....	14
Figura 1 Arquitectura Cliente-Servidor.....	15
HTML (Hypertext Markup Language) .....	15
CSS (Cascading Style Sheets) .....	16
JavaScript .....	16
Lenguaje de programación .....	¡Error! Marcador no definido.
Python: .....	16
Marco de Trabajo Framework .....	17
Objetivos de los <i>Frameworks Web</i> .....	17
Django .....	17
Vue .....	18
Gestor de base de datos .....	18
Visual Studio Code: .....	19
Herramienta CASE: .....	19
Patrones de diseño .....	19

Modelo-vista-controlador (MVC).....	20
Análisis de los resultados .....	21
Conclusiones Parciales .....	21
Capítulo 2 .....	22
Caracterización del proceso de Control a las Actividades Docentes. ....	22
Diagrama de proceso del negocio (Orientado al control a clases).....	22
Diseño de la base de Datos .....	24
Metodología de desarrollo XP (Programación Extrema).....	25
Justificación y elección de la metodología XP .....	27
Equipo de trabajo.....	27
Planificación inicial. Resumen de (HU). Historias de Usuario .....	27
Las historias de usuario (HU).....	28
Planificación de Iteraciones .....	30
Plan de entrega .....	30
Historia de Usuario .....	31
Tarea de Ingeniería.....	34
Tareas a desarrollar .....	34
Tareas para cada HU.....	37
Pruebas de aceptación .....	39
Tabla: Caso de prueba de la creación de la base de datos. Fuente:	
Elaboración del autor .....	40
Tabla: Caso de prueba Autenticación. Fuente: Elaboración del autor .....	40
Tabla: Caso de prueba Gestionar usuario. Fuente: Elaboración del autor ..	41
Tabla: Caso de prueba Gestionar Categoría investigativa. Fuente:	
Elaboración del autor .....	42
Tabla: Caso de prueba Gestionar Categoría Profesional. Fuente:	
Elaboración del autor .....	42
Pruebas de compatibilidad .....	43
Estudio de factibilidad .....	44
Descripción de los resultados obtenidos. ....	45
Conclusiones del capítulo .....	45
Capítulo 3: Análisis de los resultados. Validación de la solución propuesta. ....	46
Panel de autenticación .....	48
Imagen 1. Formulario Dimensiones, atributos y criterios de medida.....	48

Imagen 2. Formulario del modelo para el control a la actividad docente ....	49
Imagen 3 Formulario donde el jefe de departamento asigna la fecha y hora del control .....	49
Imagen 4. Formulario para evaluar el control docente .....	50
Imagen 5. Formulario para, observar cada ítem evaluado en el control con su evaluación final .....	50
Imagen 6. Formulario para filtrar el control.....	51
Imagen 7. Estadísticas .....	51
Imagen 8. Formulario resumen general del proceso .....	52
Conclusiones .....	52
Recomendaciones .....	53
Bibliografía .....	53

## **Introducción**

En la actualidad, la tecnología se ha convertido en parte fundamental de las actividades del ser humano, incluso aquellas alejadas de esta tecnología se ha convertido en un elemento indispensable. Otro de los usos más frecuentes es como medio de difusión de información masivo a las personas en cualquier región del mundo. La aparición de la internet, la red de redes, en este ámbito las tecnologías de la información y comunicación juegan un papel fundamental, al proporcionar grandes ventajas tanto desde el punto de vista de agilidad y sencillez como del volumen físico de la información resultante del proceso realizado. De ahí que se desarrolle la necesidad de elaborar técnicas y programas computacionales que permitan una mejor organización de la información además de hacer más sencillo el trabajo a realizar (Steinrücke et al. 2023). Facilita la obtención de programas informáticos eficientes y funcionales, sobre todo para la educación.

La gestión de la actividad docente es un tema de gran importancia en el ámbito educativo (Casanova 2021). Tanto las universidades como centros de educación superior enfrentan desafíos en la gestión de la actividad docente, y un ejemplo de ello está en la falta de herramientas tecnológicas adecuadas para la planificación y seguimiento de las clases impartidas por los docentes, la gestión de la asistencia de los estudiantes y la evaluación de la calidad de la

enseñanza. Dentro de todo este proceso, la evaluación de las actividades docente es esencial pues retroalimenta a todos de las mejores experiencias educativas en las aulas.

En la Universidad de Matanzas "Camilo Cienfuegos" se recoge toda la información referente al control a clase a través de documentos Word. Esta forma de recolección de la información sobre las actividades referentes a este tema hace que sea fácilmente adulterada. Además, también existe desorganización en cuanto a la información que se recoge en dicho proceso y problemas en la seguridad y transparencia del proceso.

Según entrevistas realizadas al personal que interviene en el proceso de control a las actividades docentes que se desarrollan en la Universidad de Matanzas, se pudo llegar a la conclusión que no se estaba llevando un proceso organizado y seguro. Su realización de forma manual puede llevar a errores y una pérdida de tiempo y recursos valiosos. Otro de los problemas que se encuentran en el proceso es la descoordinación que existe entre los sujetos involucrados y la información que se recoge en el proceso de control por la falta de una aplicación que oriente y organice dicho proceso.

En la actualidad el proceso donde se gestiona la información obtenida del control a clase es una tarea bastante engorrosa debido a que todo este volumen de información es tratado de manera manual y se va registrando en papel, haciendo así el trabajo más difícil y desorganizado, de ahí la necesidad de crear un sistema informático que agilice y organice este proceso.

Lo anteriormente planteado permite al autor plantearse como **Problema científico**: ¿Cómo incrementar la eficiencia en la gestión de la información asociada al proceso de control a clase en la Universidad de Matanzas?

**Hipótesis**: si se implementa una aplicación web, entonces incrementa la eficiencia de la gestión de la información asociada al control a clase en la Universidad de Matanzas

**Objeto de la investigación**: la gestión del control a clase.

**Campo acción**: La informatización de la información asociada a los controles a clases.

**Objetivo general**: Desarrollar una aplicación web incrementar la eficiencia en la gestión de la información asociada al proceso de control a clase en la Universidad de Matanzas.



### **O b j e t i v o s E s p e c í f i c o s**

- 1- D e t e r m i n a c i ó n d e l m a r c o t e ó r i c o r e f e r e n c i a l p a r a e l d e s a r r o l l o d e l a a p l i c a c i ó n w e b q u e p o s i b i l i t e e l i n c r e m e n t o d e l a e f i c i e n c i a e n l a g e s t i ó n d e l a i n f o r m a c i ó n a s o c i a d a a l p r o c e s o d e c o n t r o l a c l a s e .
- 2- I m p l e m e n t a c i ó n d e l a a p l i c a c i ó n w e b q u e p o s i b i l i t e e l i n c r e m e n t o d e l a e f i c i e n c i a e n l a g e s t i ó n d e l a i n f o r m a c i ó n a s o c i a d a a l p r o c e s o d e c o n t r o l a c l a s e e n l a U n i v e r s i d a d d e M a t a n z a s d e l a U n i v e r s i d a d d e M a t a n z a s .
- 3- V a l i d a c i ó n l a a p l i c a c i ó n w e b q u e p o s i b i l i t e e l i n c r e m e n t o d e l a e f i c i e n c i a e n l a g e s t i ó n d e l a i n f o r m a c i ó n a s o c i a d a a l p r o c e s o d e c o n t r o l a c l a s e e n l a U n i v e r s i d a d d e M a t a n z a s

### **A n t e c e d e n t e s d e l a i n v e s t i g a c i ó n**

En el ámbito de la educación superior la gestión de las actividades docentes es un tema de gran importancia y complejidad. En la Universidad de Matanzas se han realizado diversos esfuerzos para mejorar la gestión de las actividades docentes incluyendo el desarrollo de un software, sin embargo, se ha identificado que dicho software se encuentra obsoleto y no ha sido implementado de manera efectiva.

**Capítulo 1: Marco teórico referencial para el desarrollo de la aplicación web que posibilite el incremento de la eficiencia en la gestión de la información asociada al proceso de control a clase**

En este capítulo se realiza la fundamentación teórica del trabajo se abordan los conceptos fundamentales asociados al dominio del problema, se describen las tendencias tecnológicas tomadas como bases en el desarrollo del sistema, análisis de los resultados y una breve descripción de las herramientas, tecnologías y metodología utilizada en el desarrollo de esta aplicación. Este proyecto representa una mejora significativa en el proceso docente de la Universidad de Matanzas tanto para el departamento Informática como para la universidad en general al poder gestionar todo el proceso de control a clase de manera informatizada. Esta herramienta informática está orientada también en ahorrar el tiempo y esfuerzo de todo el personal que interviene en el del control a clase, así como organizar y coordinar toda la información referente al mismo.

**1.1 Metodologías de desarrollo de software**

Las metodologías de desarrollo de software son un conjunto de pasos y procedimientos que deben seguirse para llevar a cabo el desarrollo de software con calidad. Estas brindan un conjunto de detalles organizativos añadiendo rigurosidad y normas permitiendo que los integrantes de un equipo de desarrollo puedan seguir un criterio común a la hora de realizar las tareas durante el desarrollo de un software. La constante innovación tecnológica hace que cada vez sea más necesaria la aplicación de nuevas metodologías adaptadas a los nuevos tiempos (Patiño-Toro et al. 2023).

Actualmente existen dos grupos de metodologías de desarrollo, las metodologías tradicionales y las metodologías ágiles (Hasan et al. 2021; Zhi et al. 2023).

**Tabla 1: Comparación de metodologías. Fuente: (Pérez Pérez, 2012)**

Metodologías Ágiles	Metodologías Tradicionales
Preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente por el equipo	Impuestas externamente

Proceso menos controlado, con pocos principios	Proceso muy controlado, numerosas normas
Contrato flexible e incluso inexistente	Contrato prefijado
El cliente es parte del desarrollo	Cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10)	Grupos grandes
Pocos artefactos	Más artefactos
Menor énfasis en la arquitectura del software	La arquitectura del software es esencial

Tras haber realizado una comparación de ambas metodologías de realización del software, se observa que las ágiles se ajustan más para conducir el desarrollo del software propuesto, debido a que este presenta pocos artefactos y roles, hay poco tiempo de desarrollo y es mucho más flexible con los cambios en el software.

#### **Metodologías ágiles de desarrollo**

En la actualidad existen un sin número de metodologías ágiles, unas más populares que otras, cada una aportando al desarrollo ágil distintos métodos que ayudan a mejorar de una manera eficaz la calidad del software. Entre las metodologías ágiles más utilizadas se encuentran Programación Extrema (XP) y Scrum (Sfaxi and Ben Aissa 2020).

Para la elección de la metodología a utilizar se deben tener en cuenta las características que reúnen el equipo, las necesidades específicas de la situación y las prioridades demandadas por el cliente. Deben analizarse además dos factores fundamentales: el tipo de proyecto que se desea desarrollar y el tiempo que se dispone para desarrollar el mismo.

**Tabla 2: Comparativa de XP y SCRUM. Fuente: Elaboración del autor.**

Características	XP	SCRUM
Iteraciones	Iteraciones muy cortas de una o dos semanas	Iteraciones (Sprints) con una duración de dos semanas a un

		mes
Cambios	Aceptan los cambios con más facilidad en sus iteraciones	No permiten cambios en sus sprints una vez que comienzan
Orden de trabajo	Trabajan en un orden estricto. Los requerimientos que serán desarrollados son priorizados por el cliente y el equipo debe desarrollar los en ese orden específico	El dueño del producto (product owner) de Scrum prioriza la pila del producto (product backlog) pero el equipo determina el orden en el que serán desarrollados
Buenas prácticas	El software debe ser validado en todo momento, hasta el punto de que las pruebas se escriben antes del software	La validación del software se realiza al final de cada sprint, en la Revisión del Sprint (Sprint Review)
Involucramiento del usuario	El cliente participa directa y activamente en el proyecto	El cliente participa a través del propietario del producto
Principales prácticas	Simplicidad, programación en pares	Reuniones de SCRUM
Tamaño y complejidad	Usualmente preferida para proyectos pequeños y menos complejos	Se puede aplicar a cualquier tamaño de proyecto, preferiblemente grandes y complejos

#### **Metodología de desarrollo XP (Programación Extrema)**

Es una metodología ágil que se centra principalmente en las necesidades del cliente y en las relaciones interpersonales como clave para el éxito en el desarrollo del software. Se basa en el trabajo en equipo, siempre

preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo en el equipo. Uno de sus principios está en la retroalimentación continua entre el cliente y el equipo de desarrollo, la comunicación constante entre todos los participantes, claridad en las soluciones implementadas y coraje para enfrentar los cambios que se puedan producir en el avance del software (Karambir and Sharma 2016).

El objetivo de XP es muy simple: la satisfacción del cliente. Esta metodología trata de dar al cliente el software que él necesita y cuando lo necesita. Por tanto, se debe responder muy rápido a las necesidades del cliente, incluso cuando los cambios sean al final de ciclo de la programación. El segundo objetivo es potenciar al máximo el trabajo en grupo. Tanto los jefes de proyecto, los clientes y desarrolladores, son parte del equipo y están involucrados en el desarrollo del software .

Algunas de las características con las que cuenta esta metodología son (Curcio et al. 2019; López et al. 2022):

- Es una metodología ágil pensada para proyectos cortos con requerimientos muy cambiantes y pequeños grupos de desarrollo.
- Es una metodología orientada fuertemente hacia la codificación.
- Es un proyecto pequeño donde todo el trabajo se realiza por una pareja de programadores.
- La generación de numerosos artefactos y roles no es necesaria debido a que el proyecto es a corto plazo y está centrado en ser desarrollado en el menor tiempo posible.
- Su principal filosofía es ofrecer entregas del sistema agregando funcionalidades paulatinamente en corto tiempo. Esto permite que el usuario tenga más claro si el sistema hace lo que él quiere y lo compromete en desarrollo.
- Presenta un diseño evolutivo, lo cual hace que no se preste especial importancia al análisis como fase independiente, puesto que se trabaja exclusivamente en función de las necesidades del momento.

Se decidió utilizar XP debido a que se adapta en gran medida tanto al tipo de proyecto a desarrollar como a las condiciones de trabajo. Ya que el proyecto es pequeño y XP está concebida para ser utilizada dentro de proyectos pequeños,

no existe un contrato previo especificando tiempo, recursos y alcance. El cliente forma parte del equipo de desarrollo y mediante la aplicación de XP se puede lograr una retroalimentación mayor. Además, se logra un producto que satisfaga sus necesidades.

## **1.2 Tecnologías y herramientas a utilizar**

Para el desarrollo de esta investigación, se hace necesario el estudio y selección de herramientas, tecnologías y metodologías de desarrollo con el propósito de darle cumplimiento al objetivo general. A continuación, se describen aspectos a tener en cuenta.

### **1.2.1 Arquitectura del software**

La arquitectura de software proporciona las bases para el desarrollo e implementación de sistemas informáticos. En el caso de las aplicaciones web desarrolladas en la actualidad se debe tener en cuenta la arquitectura Cliente/Servidor y el patrón de arquitectura Modelo Vista Controlador (MVC), ambos elementos se describen a continuación.

#### **Arquitectura Cliente-Servidor**

Esta arquitectura se divide en dos partes claramente diferenciadas, la primera es la parte del servidor y la segunda la de un conjunto de clientes. El servidor es la integración de una máquina con el hardware necesario para responder las peticiones de los clientes con el software necesario para ello. Por otro lado, los clientes suelen ser estaciones de trabajo que solicitan varios servicios al servidor. Ambas partes deben estar conectadas entre sí mediante una red.

Esta arquitectura también se puede aplicar a programas que se ejecutan sobre un solo ordenador, aunque es más conveniente utilizarla en un sistema operativo multiusuario distribuido a través de una red de computadoras. De aquí la ventaja de realizar una aplicación web, porque una vez instalado el software en un servidor, este podrá ser accedido por los usuarios de la aplicación a través de una red, mediante un cliente en una terminal.

#### **Ventajas del Esquema Cliente/Servidor:**

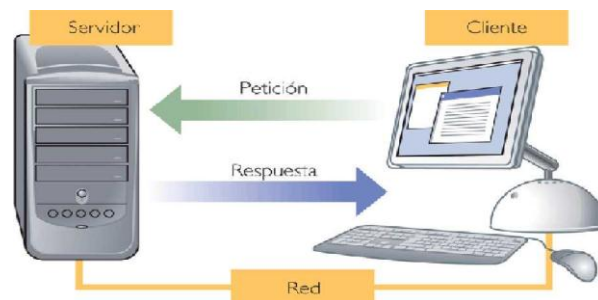
Administración centrada en el servidor: Los clientes tienen poca trascendencia en el esquema y sus necesidades de administración son menores.

Centralización de los recursos: Los recursos comunes a todos los usuarios se administran en el servidor. Así se evitan situaciones como la redundancia o inconsistencia de información en las bases de datos.

Mejora de la seguridad: Al disponer de un mecanismo central de autenticación, las posibilidades de acceso indebido se reducen considerablemente.

Escalabilidad de la instalación: Se pueden añadir o suprimir clientes sin que el funcionamiento de la red se vea afectado.

Una representación gráfica de este tipo de arquitectura sería la siguiente:



**Figura 1** Arquitectura Cliente-Servidor

En el modelo cliente servidor, el cliente envía un mensaje solicitando un determinado servicio a un servidor (hace una petición), y este envía uno o varios con la respuesta (provee el servicio), dicha arquitectura permite distribuir físicamente los procesos datos en la forma más eficiente. Esta idea se puede aplicar a programas que se ejecutan sobre una misma computadora puesto que es más ventajosa en un sistema operativo multiusuario distribuido a través de una red de computadoras.

#### **1.2.2. Lenguajes de programación.**

En la actualidad existen distintos lenguajes de programación para la Web, principalmente se destacan dos grupos, que se diferencian entre sí por el lugar que ocupan en la arquitectura Cliente - Servidor.

El primer grupo está formado por los lenguajes que se ejecutan en el lado cliente: **HTML, JavaScript y CSS**, todos encargados de aportar dinamismo a la aplicación en los navegadores. También existe un segundo grupo de lenguajes que se ejecutan en el lado servidor entre ellos se encuentran PHP, Java y Python, estos se caracterizan por desarrollar la lógica de negocio, además de ser los encargados del acceso a Bases de Datos y el tratamiento de la información.

#### **HTML (Hypertext Markup Language)**

HTML(HyperTextMarkupLanguage) es un lenguaje de marcado utilizado para estructurar y presentar contenido en la web. Es la base fundamental de la mayoría de los sitios web y se utiliza para definir la estructura y el formato

contenido, como textos, imágenes, enlaces y otros elementos multimedia (Docs,2023b). A lo largo de los años las versiones de HTML han evolucionado con el objetivo de adaptarse a los nuevos tiempos y así dar soporte a nuevas necesidades (estandarización de los sistemas de audio, vídeo) (Jiménez Ortega, 2017).

### **CSS (Cascading Style Sheets)**

Las hojas de estilo en cascada son un mecanismo que permite aplicar formato a los documentos escritos en HTML (y en otros lenguajes estructurados, como XML) separando el contenido de las páginas de su apariencia. Para el diseñador, esto significa que la información estará contenida en la página HTML, pero este archivo no debe definir cómo será visualizada esa información. Las indicaciones acerca de la composición visual del documento estarán especificadas en el archivo de la CSS. Lo que posibilita crear páginas web de una manera más exacta (Menglun Kuo et al. 2021; Saldías Kiefer and Reyes-Lillo 2021)

### **JavaScript**

Se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios (Pérez, 2019).

Un lenguaje de programación es un idioma diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Definido como un conjunto de símbolos y reglas sintácticas y semánticas que conforman su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una computadora (Louden,2004).

**Python:** es un lenguaje de alto nivel de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código, se utiliza para desarrollar aplicaciones de todo tipo. Se trata de un lenguaje de programación multiparadigma, ya que soporta parcialmente la orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un



lenguaje interpretado, dinámico y multiplataforma, posee una licencia de código abierto.

Su principal objetivo es la automatización de procesos, en este sentido, Python crea un código con gran legibilidad, que ahorra tiempo y recursos. Uno de sus puntos fuertes es que “comprueba los errores sobre la marcha” para solucionarlos cuando afectan a la memoria, lo que mantiene la integridad de la matriz y evita las complicaciones a la hora de escribir el código.

### **1.2.3. Marco de Trabajo Framework**

El término *framework* se define como un conjunto de herramientas, librerías, convenciones y buenas prácticas que pretenden unir las tareas repetitivas en módulos fácilmente reutilizables (Saavedra, 2009).

#### **Objetivos de los Frameworks Web**

Entre los principales objetivos de los *frameworks* web están (Saavedra, 2009)

- Acelerar el proceso de desarrollo.
- Permitir reutilizar código ya existente.
- Difundir las buenas prácticas de desarrollo gracias al uso de patrones de diseño.
- Disminuir el esfuerzo en el desarrollo.

#### **Django**

Django es un marco de trabajo web de alto nivel en Python que fomenta un desarrollo rápido y un diseño limpio y pragmático. Es gratuito y de código abierto. Este marco de trabajo se caracteriza porque permite construir aplicaciones de forma rápida, segura y con menos código (en comparación con otros marcos de trabajos). Además, cuenta con una comunidad de colaboradores muy grande a nivel mundial que se encarga de realizar las debidas actualizaciones tanto del marco de trabajo como de su documentación de forma diaria (Gómez García, 2018).

Django es un framework de desarrollo web en Python que facilita la creación de aplicaciones seguras, escalables y mantenibles. Proporciona un conjunto de herramientas para el manejo de bases de datos, el manejo de URLs, la administración de sesiones, el caché y la seguridad, las mismas aceleran y facilitan el desarrollo de aplicaciones web (Foundation, 2023b). Es gratuito y de código abierto. Este marco de trabajo se caracteriza porque permite construir aplicaciones de forma rápida, segura y con menos código (en comparación con

otros marcos de trabajos). Además, cuenta con una comunidad de colaboradores muy grande a nivel mundial que se encarga de realizarlas debidas actualizaciones tanto del marco de trabajo como de su documentación de forma diaria (Zneika 2019).

El patrón de arquitectura que emplea es Modelo-Vista-Plantilla (Model-View-Template o por sus siglas en inglésMVT) que sigue el mismo principio del Modelo-Vista-Controlador (Model-View-Controller o por sus siglas en inglés MVC), con la diferencia que la capa de presentación son las plantillas (Template) y la capa donde se encuentra toda la lógica de programación se denomina vista (View) (Saldías Kiefer and Reyes-Lillo 2021; Yang et al. 2021).

#### **Vue**

Vue (pronunciado /vju:/, como view) es un framework progresivo para construir interfaces de usuario. A diferencia de otros frameworks monolíticos, Vue está diseñado desde cero para ser utilizado incrementalmente. La librería central está enfocada solo en la capa de visualización, y es fácil de utilizar e integrar con otras librerías o proyectos existentes. Por otro lado, Vue también es perfectamente capaz de impulsar sofisticadas cuando se utiliza en combinación con herramientas modernas y librerías de apoyo (Widi Anggoro 2020).

#### **1.2.4. Gestor de base de datos**

**MySQL:** se define como uno de los sistemas de bases de datos más populares en el desarrollo de aplicaciones web, la cual se caracterizan por su alto rendimiento para trabajar con bases de datos. Además, gran cantidad de aplicaciones web están desarrolladas con PHP y MySQL (Eggen and Hole 2019).

Es un sistema que representa a un Sistema Gestor de Base de Datos (SGBD), donde se puede insertar las estructuras del catálogo. Para la ejecución de sentencias de definición y manejo de datos difusos, se realizará en un módulo externo, el cual este será el que se ocupe de enviar las consultas a la base de datos en SQL para de esta manera obtener los resultados y dar formato a los datos de salida para que sean mostrados al usuario (Serling-Boyd and Miloslavsky 2020). MySQL es la base de datos de código abierto más popular, ya que posee sus potentes funciones, además es fácil de configurar y usar (Zneika 2019).

#### **1.2.5. Herramientas**

**Visual Studio Code:** Es un editor de código fuente ligero pero potente que se ejecuta en su escritorio y está disponible para Windows, MacOS y Linux. Viene con soporte incorporado para JavaScript, TypeScript y Node.js y tiene un rico ecosistema de extensiones para otros lenguajes (como C++, C#, Java, Python, PHP, Go) y tiempos de ejecución (como .NET y Unity). (Microsoft, 2022)

#### **Herramienta CASE:**

Una herramienta CASE (Computer-Aided Software Engineering) es un software utilizado para apoyar el desarrollo de software de manera automatizada. Estas herramientas se utilizan en cada una de las etapas del ciclo de vida del software, desde la especificación de requisitos hasta la implementación y mantenimiento. Pueden incluir diferentes funcionalidades, como la gestión de requisitos, el modelado de sistemas, la generación de código, la depuración y el control de versiones. Estas permiten una mayor eficiencia y calidad en el desarrollo de software, reduciendo los errores y el tiempo necesario para el desarrollo y mantenimiento del software (Gurses et al. 2023).

#### **Herramientas CASE: Visual Paradigm -UML**

Es una herramienta UML profesional que soporta el ciclo de vida completo desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación. Presenta licencia gratuita y comercial. Es fácil de instalar y actualizar y compatible entre ediciones.

Se seleccionó como herramienta CASE a Visual Paradigm porque soporta el ciclo de vida completo del desarrollo de software, permite modelar los diagramas de la solución propuesta, realizar ingeniería tanto directa como inversa, permite el control de versiones y generar la documentación automáticamente en formatos como web o PDF. Además, este software de modelado UML (Ozkaya and Erata 2020) ayuda a una rápida construcción de aplicaciones de calidad, mejores y a un menor coste. A su vez es una herramienta libre y multiplataforma.

#### **1.2.6 Patrones de diseño.**

Los patrones de diseño son un conjunto de reglas que describen como afrontar tareas y solucionar problemas que surgen durante el desarrollo del software.

Estos identifican y especifican abstracciones que van más allá del simple ámbito de clases e instancias, o componentes. Para que una solución sea considerada un patrón debe poseer ciertas características como son efectividad habiendo resuelto problemas similares en ocasiones anteriores y reusabilidad permitiendo su aplicación a diferentes problemas de diseño en distintas circunstancias (Tullo et al. 2023; Wu 2020).

Se decide utilizar el patrón de diseño Modelo-Vista-Controlador que se explica a continuación:

#### **Modelo-vista-controlador (MVC)**

El modelo-vista-controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica del negocio de una aplicación, de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento.

De manera genérica, los componentes de MVC se podrían definir como sigue:

**El Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, tantas consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación (lógica de negocio) (Wu et al. 2023). El modelo envía a la vista aquella parte de la información que en cada momento se le solicita para que sea mostrada (típicamente a un usuario). Las peticiones de acceso o manipulación de información llegan al modelo a través del controlador.

**El Controlador:** Responde a eventos (usualmente acciones del usuario) e invoca peticiones al modelo cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro en una base de datos). Se podría decir que el controlador hace de intermediario entre la vista y el modelo.

La **Vista**: Presenta el modelo (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho modelo la información que debe representar como salida.

#### **Análisis de los resultados**

Una vez llevadas a cabo eficientemente las tareas de investigación propuestas se espera lograr un software capaz de:

1. Gestionar controles de la actividad docente de manera informatizada
2. Elaborar parámetros que se evaluarán en el control a las actividades docentes.
3. Sugerir una evaluación para los controles.
- 4- Gestionar una evolución de los parámetros a evaluar los controles y generar una gráfica para ver el comportamiento y desempeño de los profesores en las actividades docentes.
- 5- Generar reportes.

#### **Conclusiones Parciales**

Con la elaboración de este capítulo quedaron establecidas las bases teóricas que sustentan esta investigación, se definieron conceptos indispensables para la comprensión del objeto de estudio. Se definió XP como la metodología indicada para desarrollar la solución propuesta por esta investigación, se definieron las tecnologías para el desarrollo de la aplicación web Django para el servidor y Vue para el cliente, además. Se seleccionó MySQL como sistema gestor de base datos. Se eligieron las principales herramientas a utilizar, entre ellas Visual Studio Code y Visual Paradigm como herramientas CASE:

En resumen, se obtiene una base teórica que permite continuar con el desarrollo de esta investigación.

## **Capítulo 2: Descripción de la aplicación web que posibilite el incremento de la eficiencia en la gestión de la información asociada al proceso de control a clase en la Universidad de Matanzas de la Universidad de Matanzas**

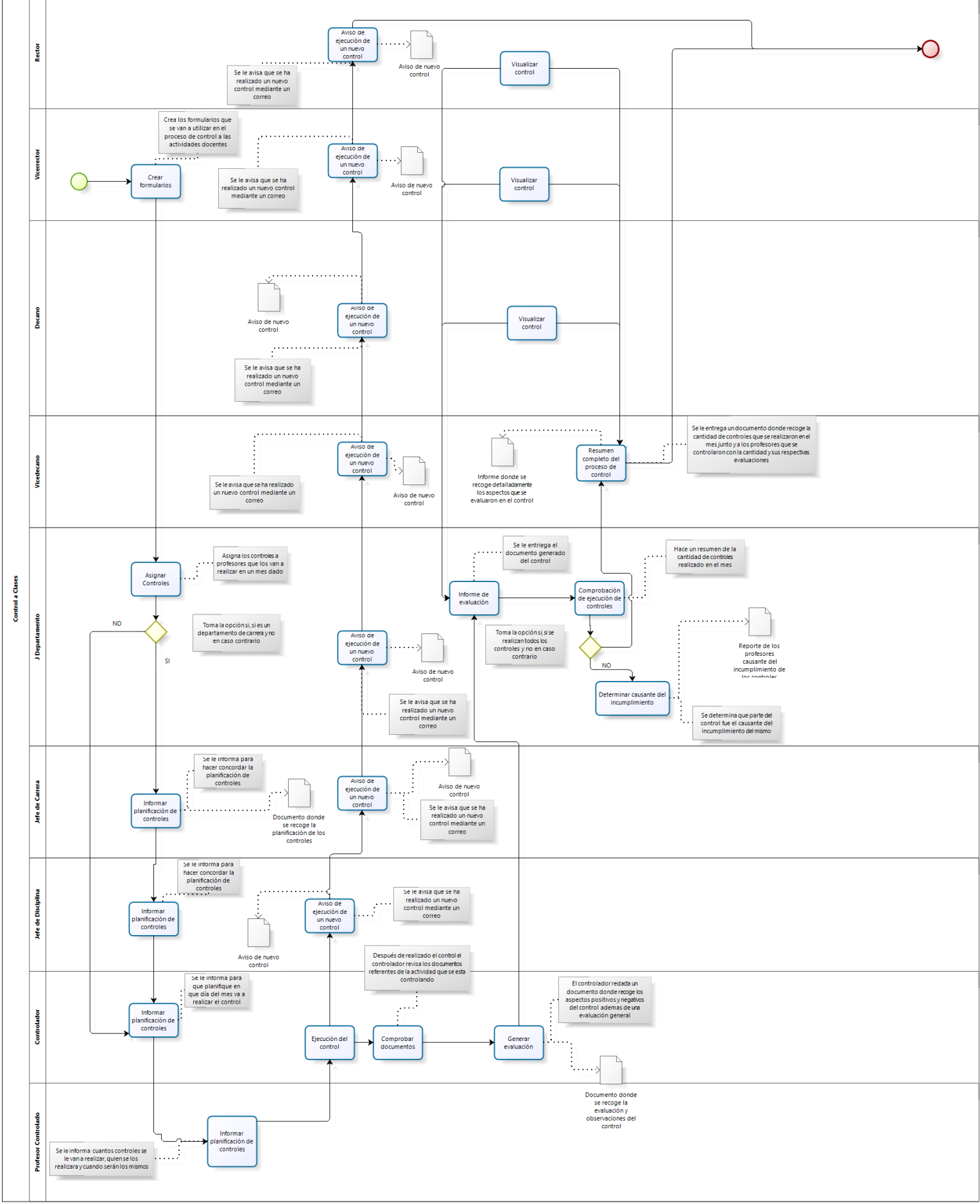
En este acápite se realiza el proceso de análisis, la descripción del sistema y el diseño de la solución propuesta. Se definen los requisitos funcionales y no funcionales. Se desarrolla la etapa de planificación basada en la metodología ágil de desarrollo de software XP donde se describen las historias de usuario, se presenta el equipo de trabajo, se definen las tareas y prueba de aceptación. Se muestra el diagrama de base de datos y se resumen los patrones utilizados en la solución propuesta.

### **2.1 Caracterización del proceso de Control a las Actividades Docentes.**

El proceso de control a las actividades docentes es uno de los procesos más importantes en el ámbito universitario debido a que retroalimenta a los ejecutores y a los dirigentes del estado real de cada uno de ellos. En las universidades las actividades docentes juegan un papel esencial pues son aquellas en las cuales se organiza o se materializan los procesos formativos. De ahí que el control a estas actividades sea de vital importancia para la comunidad universitaria pues impacta directamente en la formación del profesional.

Para la vicerrectoría docente de la Universidad de Matanzas no todas las actividades docentes tienen el mismo peso ni se evalúan de la misma manera, aunque reconoce la significativa importancia de la actividad denominada clase. Esta actividad se ejecuta a partir de la carrera y la disciplina para detectar los errores que se cometen lo cual permite la adopción de estrategias metodológicas para su corrección. En estos controles pueden intervenir varios controladores y controlados, en dependencia del nivel y el tipo de actividad a controlar. Cada uno de estos controles genera la documentación necesaria para describir la evaluación obtenida y su calificación en dependencia de la norma que haya establecido la institución

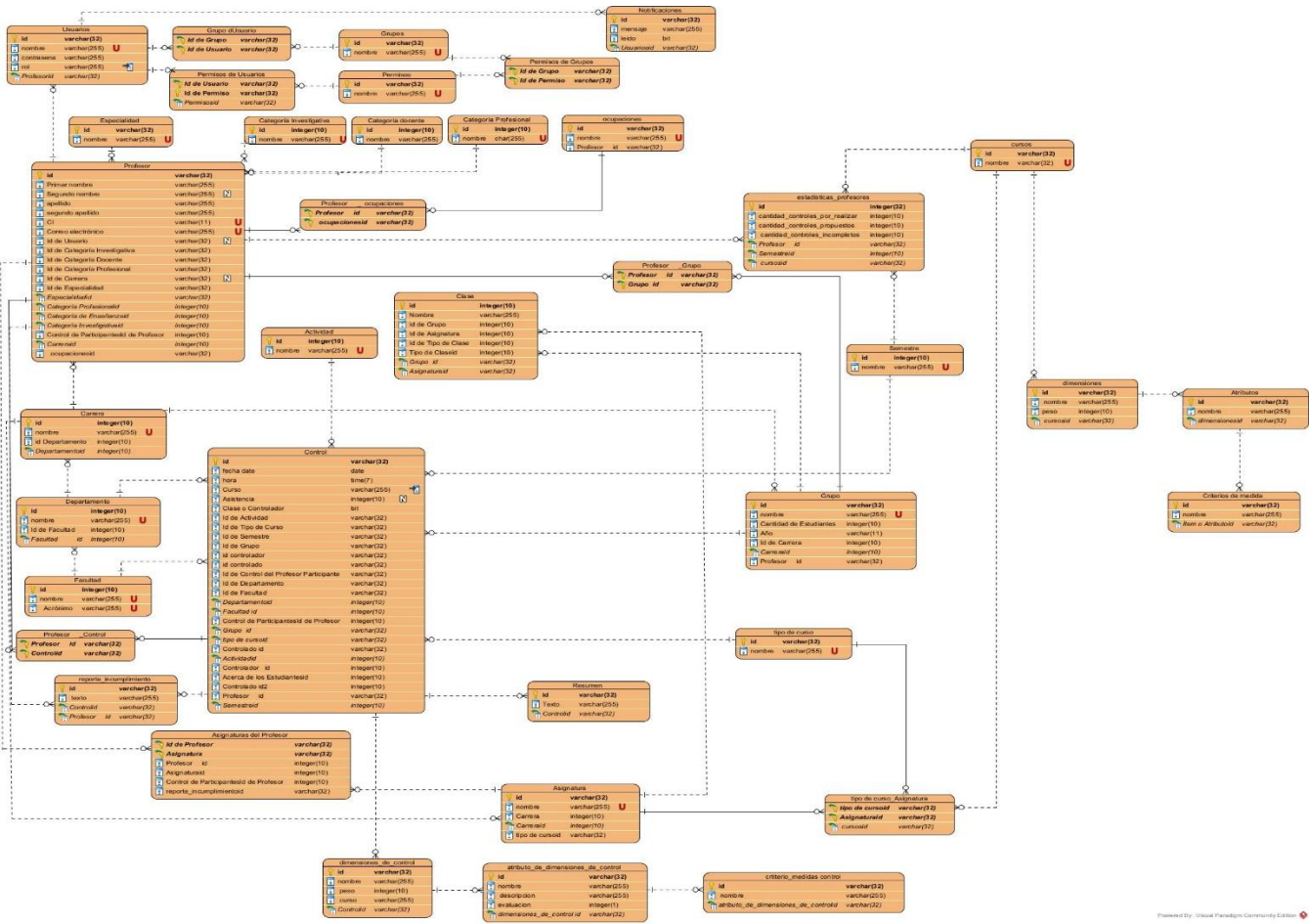
### **Diagrama de proceso del negocio (Orientado al control a clases)**



El control a clases, como se observa en la figura, comienza con el vicerrector creando el formulario que se va tomar como guía en dicho proceso en el cual se colocan las dimensiones, indicadores y criterios de medida. Siguiendo el flujo, el jefe de departamento, asignaría los controles que les realizar a cada profesor y qué papel va a desempeñar en el mismo. Si es un departamento de carrera se le informa al jefe de carrera y al de disciplina para una mejor organización sino se les notifica directamente a los profesores involucrados en los controles. A continuación, se realiza el control que comienza comprobando la actividad y posteriormente se solicitan los documentos. Una vez ejecutado el control el profesor controlador, por un lado, manda un aviso de ejecución de un nuevo control a todos los directivos, por otra parte, revisa los documentos referentes a la actividad que se está controlando y genera una evaluación donde recoge los aspectos del control, tanto positivos como negativos. De ahí se le entrega el informe de evaluación al jefe del departamento, es decir el documento generado del control, el cual comprueba la ejecución de los controles, realiza un resumen de la cantidad de controles realizados y verifica si se realizaron o no todos los controles, en caso de no haberse realizado determinar el motivo causante del incumplimiento, si se cumplieron se le entrega a la vicedecana el documento donde recoge la cantidad de controles que se realizaron, los profesores que se controlaron con la cantidad y sus respectivas evaluaciones y la vicedecana entrega el resumen general del proceso a la vicerrectora.

#### **Diseño de la base de Datos.**





## 2.2 Metodología de desarrollo XP (Programación Extrema)

Propone cuatro fases de desarrollo (Bouihia and Bahaja 2019)

➤ **Planificación:** se realiza el análisis del negocio y el levantamiento de requisitos, se elabora un plan de entrega y su estimación.

- **Las Historias de Usuarios:** son descritas por el cliente, en su propio lenguaje, como descripciones cortas de lo que el sistema debe realizar,
- **El Plan de Entregas (ReleasePlan):** las historias de usuarios serán agrupadas para conformar una entrega y el orden de las mismas. Este cronograma será el resultado de una reunión entre todos los actores del proyecto,
- **El Plan de Iteraciones (Iteration Plan):** las historias de usuarios seleccionadas para cada entrega son desarrolladas y probadas en un ciclo de iteración, de acuerdo al orden preestablecido,

➤ **Diseño:** en esta fase se diseña el sistema, se realiza el modelado de las clases y se define la arquitectura del sistema.

- **Simplicidad:** un diseño simple se implementa más rápidamente que uno complejo. Por ello XP propone implementar el diseño más simple posible que funcione
- **Recodificación (Refactoring):** consiste en escribir nuevamente parte del código de un programa, sin cambiar su funcionalidad, a los efectos de crearlo más simple, conciso y entendible. Las metodologías de XP sugieren recodificar cada vez que sea necesario

➤ **Implementación:** se desarrolla el sistema a partir del diseño realizado.

- **Disponibilidad del Cliente:** uno de los requerimientos de XP es tener al cliente disponible durante todo el proyecto. No solamente como apoyo a los desarrolladores, sino formando parte del grupo. El involucramiento del cliente es fundamental para que pueda desarrollarse un proyecto con la metodología XP
- **Programación en Pares:** XP propone que se desarrolle en pares de programadores, ambos trabajando juntos en un mismo ordenador. Al trabajar en pares se minimizan los errores y se logran mejores diseños, compensando la inversión en horas. El producto obtenido es por lo general de mejor calidad que cuando el desarrollo se realiza por programadores individuales

➤ **Pruebas:** se prueba y se verifica el correcto funcionamiento del sistema. Luego de realizadas las pruebas y la corrección de los errores detectados, se comienza una nueva iteración (Stradowski and Madeyski 2023).

**Detección y Corrección de Errores:** cuando se encuentra un error, éste debe ser corregido inmediatamente, y se deben tener precauciones para que errores similares no vuelvan a ocurrir. Asimismo, se generan nuevas pruebas para verificar que el error haya sido resuelto (Salahirad et al. 2023)

**Pruebas de Aceptación:** son creadas en base a las historias de usuarios, en cada ciclo de la iteración del desarrollo. El Cliente debe especificar uno o diversos escenarios para comprobar que una historia de usuario ha sido correctamente implementada. Asimismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se

puede considerar terminada hasta que pase correctamente todas las pruebas de aceptación (Fokides 2023)

**Justificación y elección de la metodología XP**

Considerando que en la solución propuesta el cliente es parte activa del equipo de desarrollo, que tiene bien definido el orden prioritario de las funcionalidades, las cuales, están sujetas a posibles cambios y que el desarrollo es llevado a cabo por un equipo, se hace necesario establecer buenas prácticas para un mejor funcionamiento del grupo, esto provoca que aspectos como el orden de trabajo, cambios, buenas prácticas, involucramiento del usuario tengan un peso mayor a la hora de decidir cual metodología escoger.

Por lo que se opta por seleccionar a XP como metodología de desarrollo de software debido a que se ajusta más a la situación en la que se diseña y se desarrolla la solución propuesta y además por estar dentro del grupo de metodologías ágiles, se caracteriza por su sencillez tanto en su aprendizaje, como en su aplicación. Es adecuada para cualquier tipo de proyecto y para equipos de cualquier tamaño.

**Equipo de trabajo.**

M i e m b r o s	R o l e s
DrC. W alfredo González	Jefe de Proyecto, Tester
Yanlis Rodríguez Orihuela	C l i e n t e
Yariel Ortiz Rodríguez	Programador, Tester

El cliente: escribe los requisitos y características del sistema como historias y pruebas funcionales, y les asigna las prioridades apropiadas y decide al final del proceso si cada requisito se cumple o no. El programador: es el responsable de escribir códigos de funcionamientos simples y de alta calidad, generalmente realizados en colaboración con otros programadores de acuerdo con la práctica de programación en pareja. El entrenador: responsable de guiar el proceso general.

**Planificación inicial. Resumen de (HU). Historias de Usuario.**

**Planificación inicial:** en la implementación de un software es necesaria una planificación desde el inicio pensar en soluciones y organizar de forma eficiente el modelado inicial del problema; para alcanzar este propósito se desarrollan las Historias de Usuarios. Partiendo de las ideas del cliente, obteniendo un punto de partida para el resto de la planificación del proyecto. Es importante

realizar un estimado del tiempo y de las entregas del proyecto en conjunto con el cliente, conscientes que estos aspectos pueden sufrir algunos cambios durante el desarrollo.

**Las historias de usuario (HU)** son la técnica utilizada en XP para especificar los requisitos del software (Wamema et al. 2023; Zhi, Gong, Ren, Liu, Zhou and Yamamoto 2023). Ayudan en la comunicación entre el cliente y el programador de esta manera pueden ir cambiando a medida que avanza el proyecto y que el cliente vea nuevas soluciones. Ayuda a definir las iteraciones con las que contará el proyecto, teniendo en cuenta, la prioridad en el negocio que es establecida por el cliente, el riesgo de desarrollo asignado por el programador y la estimación de tiempo en desarrollo de cada una que debe ser corto, de una a tres semanas. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales (Behutiye et al. 2022)

En la siguiente tabla se muestra un resumen de todas las HU planificadas inicialmente, en las que queda definido el nivel de prioridad con el que deben darle solución a las HU (P), el riesgo en desarrollo (R). Se define también la estimación del esfuerzo (E) requerido, que no es más que el tiempo en el que se concibió inicialmente el desarrollo de cada HU.

No.	Historia de Usuario	Prioridad	Riesgo	Esfuerzo	Iteración	Entrega
1	Diseño y creación de la Base de Datos.	Alta	Alto	1.5	1	1
2	Autenticarse.	Alta	Medio	0.5	1	1
3	Gestionar usuarios.	Media	Medio	1	2	2
4	Gestionar Categoría Investigativa.	Media	Bajo	0.5	2	2
5	Gestionar Categoría	Media	Bajo	0.5	2	2

	Profesional.					
6	Gestionar Especialidades	Media	Bajo	0.5	2	2
7	Gestionar Profesores	Alta	Medio	1	3	3
8	Gestionar tipos de actividades	Medio	Medio	0.5	4	5
9	Gestionar Facultades	Media	Bajo	0.5	4	5
10	Gestionar Departamento.	Media	Bajo	0.5	4	5
11	Gestionar Carrera.	Media	Media	0.5	4	5
12	Gestionar Tipos de Cursos	Alta	Media	0.5	4	5
13	Gestionar semestres	Media	Media	0.5	4	5
14	Gestionar Grupo.	Media	Bajo	0.5	4	5
15	Gestionar Asignatura.	Media	Bajo	0.5	4	5
16	Gestionar Control.	Alta	Alto	3	5	6
17	Generar estadísticas	Media	Bajo	1	6	7
18	Generar Excel del control por curso y semestre	Media	Media	1	7	7
<b>Totales</b>					14	7

**Planificación de Iteraciones**

Las iteraciones fueron planificadas mediante entrevistas al cliente, se incluye en cada una la realización de una o más historias de usuarios y pueden realizarse tantas como sean necesarias. Al terminar cada iteración se recomienda realizar la entrega correspondiente. Para realizar esta planificación se tuvo en cuenta que las iteraciones no fuesen muy largas y no excedieran las tres semanas.

**Plan de entrega**

Las entregas del sistema se realizan al finalizar una iteración, aunque pueden estar compuestas por una o más. Realizar una planificación para las entregas hace que se pueda realizar un control por parte del plan acordado con el cliente. Debe ser negociado y elaborado en forma conjunta entre el cliente y el equipo de desarrollado durante las reuniones de planificación de entregas, la idea es hacer entregas frecuentes para obtener una mayor retroalimentación.

A continuación, se muestra en el plan de entrega definido para el ciclo de desarrollo:

Tabla 2. Plan de entrega

Iteración	Historia de Usuario	Fecha de entrega
1	2	1 de julio del 2023
2	3	22 de julio del 2023
3	7	12 de agosto del 2023
4	10	2 de septiembre del 2023
5	16	23 de septiembre del 2023
6	17	7 de octubre del 2023
7	18	4 de noviembre del 2023

## Historia de Usuario

Historia de Usuario	
<b>Número:</b> 1	<b>Nombre Historia:</b> Diseño y creación de la Base de Datos
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 1.5	<b>Iteración Asignada:</b> 1
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se diseña e implementa la base de datos en el gestor MySQL.	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 2	<b>Nombre Historia:</b> Autenticarse
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 1
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
Se inicia cuando el usuario, administrador (coordinadora) del sistema intenta acceder para realizar las actividades asignadas de acuerdo con los permisos asignados	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 3	<b>Nombre Historia:</b> Gestionar usuario
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 1	<b>Iteración Asignada:</b> 2
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar usuario	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 4	<b>Nombre Historia:</b> Gestionar categoría investigativa
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Media	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 2
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar categoría investigativa	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 5	<b>Nombre Historia:</b> gestionar categoría profesional
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Media	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 2
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar categoría profesional	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 6	<b>Nombre Historia:</b> Gestionar Especialidades
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Baja
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 2
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar Especialidades	
<b>Observaciones:</b> Confirmado por el cliente	



--

Historia de Usuario	
<b>Número:</b> 7	<b>Nombre Historia:</b> Gestionar Profesor
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Media
<b>Puntos estimados:</b> 1	<b>Iteración Asignada:</b> 3
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar Profesor	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 8	<b>Nombre Historia:</b> Gestionar tipos de actividades
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 4
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar tipos de actividades	
<b>Observaciones:</b> Confirmado por el cliente	

Historia de Usuario	
<b>Número:</b> 9	<b>Nombre Historia:</b> Gestionar Facultades
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Media	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 4
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	

<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar Facultades
<b>Observaciones:</b> Confirmado por el cliente

Historia de Usuario	
<b>Número:</b> 10	<b>Nombre Historia:</b> Gestionar Departamento
<b>Usuario:</b> Administrador	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Bajo
<b>Puntos estimados:</b> 0.5	<b>Iteración Asignada:</b> 4
<b>Programador responsable:</b> Yariel Ortiz Rodríguez	
<b>Descripción:</b> Se inicia cuando el usuario, administrador del sistema desea añadir, listar, modificar y eliminar Departamento	
<b>Observaciones:</b> Confirmado por el cliente	

### Tarea de Ingeniería

Las Tareas de Ingeniería son actividades que los programadores conocen que el sistema debe hacer. Deben ser estimables, su tiempo de implementación debe ser corto, aproximadamente entre uno y tres días, su objetivo es resolver las Historias de Usuario. Una Historia de Usuario puede tener una o varias Tareas de Ingeniería en dependencia de la funcionalidad a desarrollar. Pueden existir también tareas de ingeniería técnicas, que son aquellas que, aunque no derivan directamente de una Historia de Usuario, es necesaria su consideración para que el sistema funcione (Behutiye et al. 2020).

### Tareas a desarrollar

Cada Historia de Usuario se transformará en Tareas de Ingeniería (TI) según lo muestra la siguiente tabla

No	Nombre HU	Tarea de Ingeniería	Iteración
1	Diseño y Creación de la base de datos.	1-Diseño de la base de datos. 2-Creación de la base de datos.	1

2	Autenticarse .	3- Autenticarse .	1
3	Gestionar usuarios .	4- Añadir usuario 5-Listar usuario . 6-M odificar usuario 7- Elim inar usuario .	2
4	Gestionar Categorías Investigativas .	8- Añadir Categoría Investigativa 9-Listar Categoría Investigativa 10-M odificar Categoría Investigativa 11-E lim inar Categoría Investigativa	2
5	Gestionar Categorías Profesionales	12 añadir Categoría Profesional 13-Listar Categoría Profesional 14-M odificar Categoría Profesional 15-E lim inar Categoría Profesional	2
6	Gestionar Especialidades	16- Añadir Especialidad 17-Listar Especialidad 18-M odificar Especialidad 19- Elim inar Especialidad	2
7	Gestionar Profesores	20- Añadir Profesor 21-Listar Profesor 22-M odificar Profesor 23- Elim inar Profesor	3
8	Gestionar tipos de actividades	24- Añadir tipos de actividades 25-Listar tipos de actividades 26-M odificar tipos de actividades 27- Elim inar tipos de actividades	4

9	Gestionar Facultades	28 - Añadir Facultad 29 - Listar Facultad 30 - Modificar Facultad 31 - Eliminar Facultad	4
10	Gestionar Departamento	32 - Añadir Departamento 33 - Listar Departamento 34 - Modificar Departamento 35 - Eliminar Departamento	4
11	Gestionar Carreras	36 - Añadir Carrera 37 - Listar Carrera 38 - Modificar Carrera 39 - Eliminar Carrera	4
12	Gestionar Tipos de Cursos	40 - Añadir Tipos cursos 41 - Listar Tipos cursos 42 - Modificar Tipos cursos 43 - Eliminar Tipos cursos	4
13	Gestionar semestres	44 - Añadir semestre 45 - Listar semestre 46 - Modificar semestre 47 - Eliminar semestre	4
14	Gestionar Grupo	48 - Añadir Grupo 49 - Listar Grupo 50 - Modificar Grupo 51 - Eliminar Grupo	4
15	Gestionar Asignatura	52 - Añadir Asignatura 53 - Listar Asignatura 54 - Modificar Asignatura 55 - Eliminar Asignatura	4

16	Gestionar Tipos de Clases	56- Añadir Tipo de Clase 57- Listar Tipo de clase 58- Modificar Tipo de Clase 59- Eliminar Tipo de Clase	4
17	Gestionar Control	60- Crear Control 61- Actualizar Control 62- Crear Evaluación 63- Crear resumen 64- Crear reporte de incumplimiento	5
18	Generar estadísticas		6
19	Generar Excel del control por curso y semestre		7

**Tareas para cada HU.**

Tarea de Ingeniería	
Número de Tarea: 1	Número de Historia de Usuario: 1
Nombre de Tarea: Diseño de la base de datos.	
Tipo de Tarea: Diseño.	Puntos Asignados: 1
Programador responsable: Yariel Ortiz Rodríguez	
Descripción: Analizar con profundidad el negocio referente a la gestión de la información de los proyectos de investigación de la Universidad de Matanzas y luego diseñar la base de datos que permita almacenar dicha información.	

Tarea de Ingeniería	
Número de Tarea: 2	Número de Historia de Usuario: 1
Nombre de Tarea: Creación de la base de datos.	
Tipo de Tarea: Desarrollo.	Puntos Asignados: 0.5
Programador responsable: Yariel Ortiz Rodríguez	

Crear en MySQL la base de datos con sus respectivas relaciones y la integridad correspondiente entre las tablas.

#### Tarea de Ingeniería

**Número de Tarea:** 3

**Número de Historia de Usuario:** 2

**Nombre de Tarea:** Autenticarse

**Tipo de Tarea:** Desarrollo.

**Puntos Asignados:** 0.5

**Programador responsable:** Yariel Ortiz Rodríguez

**Descripción:** Se diseña la interfaz para introducir los datos de acceso. Después verificarlos en la base de datos. Se procede a autenticar el usuario en el sistema. Se le concede los permisos según su rol

#### Tarea de Ingeniería

**Número de Tarea:** 4

**Número de Historia de Usuario:** 3

**Nombre de Tarea:** Añadir usuario

**Tipo de Tarea:** Desarrollo.

**Puntos Asignados:** 0.2

**Programador responsable:** Yariel Ortiz Rodríguez

**Descripción:** Se mostrará un formulario donde el usuario con los permisos necesarios podrá añadir usuario.

#### Tarea de Ingeniería

**Número de Tarea:** 5

**Número de Historia de Usuario:** 3

**Nombre de Tarea:** Listar usuario.

**Tipo de Tarea:** Desarrollo.

**Puntos Asignados:** 0.3

**Programador responsable:** Yariel Ortiz Rodríguez

**Descripción:** Se mostrará un formulario donde el usuario con los permisos necesarios podrá Listar usuario.

#### Tarea de Ingeniería

<b>Número de Tarea: 6</b>	<b>Número de Historia de Usuario: 3</b>
<b>Nombre de Tarea: Modificar usuario</b>	
<b>Tipo de Tarea: Desarrollo.</b>	<b>Puntos Asignados: 0.2</b>
<b>Programador responsable: Yariel Ortiz Rodríguez</b>	
<b>Descripción: Se mostrará un formulario donde el usuario con los permisos necesarios podrá modificar usuario.</b>	

Tarea de Ingeniería	
<b>Número de Tarea: 7</b>	<b>Número de Historia de Usuario: 3</b>
<b>Nombre de Tarea: Eliminar usuario.</b>	
<b>Tipo de Tarea: Desarrollo.</b>	<b>Puntos Asignados: 0.3</b>
<b>Programador responsable: Yariel Ortiz Rodríguez</b>	
<b>Descripción: Se mostrará un formulario donde el usuario con los permisos necesarios podrá eliminar usuario.</b>	

#### **-Pruebas**

La prueba de software es el proceso de evaluar y verificar que un producto o aplicación de software hace lo que se supone que debe hacer. Los beneficios de las pruebas incluyen la prevención de errores, la reducción de los costos de desarrollo y la mejora del rendimiento (Dehlaghi-Ghadim et al. 2023).

#### **Pruebas de aceptación.**

Las pruebas de aceptación significan la satisfacción o no, del cliente con el producto desarrollado y el final de una iteración y el comienzo de la siguiente. Se elaboran a lo largo de la iteración, en paralelo con el desarrollo del sistema, y adaptándose a sus cambios. El objetivo de las pruebas de aceptación es validar que un sistema cumpla con el funcionamiento esperado, lo que permite al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento (Abegg et al. 2023).

Las pruebas de aceptación son consideradas como "pruebas de caja negra". Los clientes son responsables de verificar que los resultados de estas pruebas son correctos. Así mismo, en caso de que fallen varias pruebas, deben indicar el orden de prioridad de resolución. Una historia de usuario no se puede

considerar terminada hasta tanto pase correctamente todas las pruebas de aceptación (Mellor 2023; Peng et al. 2023).

**Tabla: Caso de prueba de la creación de la base de datos. Fuente:**

**Elaboración del autor**

Caso de Prueba	
Número Caso de Prueba: 1	Número Historia de Usuario: 1
Nombre Caso de Prueba: Creación de la Base de Datos.	
Descripción: Se comprueba que haya sido diseñada y creada correctamente la base de datos.	
Condiciones de ejecución: Solo pueden realizar las acciones crear, editar o eliminar procesos el rol permitido.	
Entradas: Datos de Prueba	
Resultado esperado: Todos los conjuntos de datos probados han sido correctamente procesados.	
Evaluación: Prueba satisfactoria	

**Tabla: Caso de prueba Autenticación. Fuente: Elaboración del autor**

Caso de Prueba	
Número Caso de Prueba: 2	Número Historia de Usuario: 2
Nombre Caso de Prueba: Autenticación	
Descripción: Se realizan intentos de entrar al sistema con conjuntos de usuarios y contraseñas incorrectos, campos en blanco y al final con un usuario registrado y habilitado	
Condiciones de ejecución: Usuario válido en el sistema.	
Entradas: Usuario y contraseña.	



<p>Resultado esperado: la aplicación debe alertar al usuario cuando se inserten datos erróneos. Cuando se inserten los datos correctamente, el usuario debe entrar y hacer uso de las acciones que le están permitidas según los permisos asociados al mismo.</p>
<p>Evaluación: Prueba satisfactoria</p>

**Tabla: Caso de prueba Gestionar usuario. Fuente: Elaboración del autor**

Caso de Prueba	
Número Caso de Prueba: 3	Número Historia de Usuario: 3
Nombre Caso de Prueba: Gestionar usuario	
<p>Descripción: Se realizan intentos de autenticación en el sistema con diferentes conjuntos de usuarios y contraseñas, incluyendo casos de datos incorrectos y campos en blanco. Finalmente, se realiza una autenticación exitosa con un usuario registrado y habilitado.</p>	
<p>Condiciones de ejecución: Usuario válido en el sistema con permisos para gestionar usuarios.</p>	
<p>Entradas: Nombre de usuario y contraseña.</p>	
<p>Resultado esperado: la aplicación debe alertar al usuario cuando se inserten datos erróneos. Cuando se inserten los datos correctamente, el usuario debe entrar y hacer uso de las acciones que le están permitidas según los permisos asociados al mismo.</p>	
<p>Evaluación: Prueba satisfactoria</p>	

**Tabla: Caso de prueba Gestionar Categoría investigativa. Fuente:**

**Elaboración del autor**

<b>Caso de Prueba</b>	
<b>Número Caso de Prueba: 4</b>	<b>Número Historia de Usuario: 4</b>
<b>Nombre Caso de Prueba:</b> Gestión de Categoría Investigativa	
Descripción: Este caso de prueba comprueba que los usuarios con el rol correspondiente puedan agregar, editar y eliminar categorías investigativas en el sistema.	
<b>Condiciones de ejecución:</b> Usuario válido en el sistema con permisos para gestionar categorías investigativas.	
<b>Entradas:</b> Nueva categoría investigativa para agregar, categoría investigativa existente para editar o eliminar.	
<b>Resultado esperado:</b> El sistema debe permitir agregar, editar y eliminar categorías investigativas según los permisos del usuario. La información de la categoría investigativa debe ser actualizada correctamente en el sistema.	
<b>Evaluación:</b> Prueba satisfactoria	

**Tabla: Caso de prueba Gestionar Categoría Profesional. Fuente:**

**Elaboración del autor**

<b>Caso de Prueba</b>	
<b>Número Caso de Prueba: 5</b>	<b>Número Historia de Usuario: 5</b>

<b>Nombre Caso de Prueba:</b> Gestionar Categoría Profesional.
<b>Descripción:</b> Este caso de prueba comprueba que los usuarios con el rol correspondiente puedan agregar, editar y eliminar categorías profesionales en el sistema.
<b>Condiciones de ejecución:</b> Usuario válido en el sistema con permisos para gestionar categorías profesionales.
<b>Entradas:</b> Nueva categoría profesional para agregar, categoría profesional existente para editar o eliminar.

#### **Resultados obtenidos por las pruebas de aceptación**

Después de desarrollar todo un proceso de pruebas se obtuvieron resultados satisfactorios. Sirvieron para ver si las validaciones están hechas correctamente, para que todos los errores que se encontraron en la revisión del sistema con el cliente fueran corregidos. Se solucionaron varios problemas que impedían el cumplimiento de los requisitos fundamentales del software, para ganar en credibilidad y aceptación por el cliente. El desarrollo de la aplicación cumple las expectativas trazadas al inicio del proyecto y satisface al cliente en su totalidad.

#### **Pruebas de compatibilidad**

Se asegura que la aplicación web funcione correctamente en distintos navegadores web. No todos interpretan el código HTML y CSS de la misma manera por lo que estas diferencias en ocasiones provocan un mal funcionamiento del sitio o la pérdida de visualización (Córdoba, 2015). Para realizar dichas pruebas existen diversas técnicas ya definidas. Entre estas se encuentran las siguientes:

**Verificación del cumplimiento de estándares:** consiste en analizar los componentes gráficos del sitio web en diferentes navegadores, para verificar que sigan los lineamientos y especificaciones de los estándares (Mascheroni, Cogliolo e Irrazabal, 2016),

**Pruebas de interfaz de usuario:** es la más común de las técnicas. Puede ser realizada de forma manual o con software especializado (pruebas automatizadas). Su objetivo es revisar el contenido visual del sitio web a través

de la navegación de sus páginas en los diferentes navegadores (Memon, Banerjee y Nagarajan, 2003), (Xie y Memon, 2007),

**Análisis del modelo de objetos del documento (Document Object Model DOM):** es una técnica dinámica que consiste en comparar el comportamiento de una aplicación web en diferentes navegadores, identificando las diferencias como defectos (Choudhary, Versee y Orso, 2010),

**Comparación de imágenes:** se basa en tomar una captura de pantalla del sitio en un tipo de navegador, y compararla con otra captura del sitio en otro navegador diferente del primero. Si ambas imágenes coinciden, entonces el sitio será compatible entre ambos navegadores (Hori y col., 2015).

#### **Estudio de factibilidad**

Uno de los factores importantes que se debe tener en cuenta en el desarrollo de un software, es si las ventajas de la aplicación propuesta justifican o no su costo. Además de realizar una estimación del esfuerzo y el tiempo de desarrollo es necesario llevar a cabo un análisis de los beneficios tangibles e intangibles que reportará el proyecto (Birchler et al. 2023). El tiempo que se empleará en cada historia de usuario se estima basado en la experiencia del programador en el trabajo con el lenguaje de programación, el entorno de desarrollo, el conocimiento sobre el tema de investigación y las técnicas de programación necesarias para resolver el problema. La estimación por lo tanto es de forma empírica y desde el comienzo se conoce el tiempo de duración total estimado del proyecto.

Se muestra un total de 8.5 semanas de trabajo de implementación y si a esto se suma el tipo de realización de pruebas de aceptación, pruebas de factibilidad; resulta un total de 15 semanas. Para un total de 4 meses aproximadamente.

**Utilizando la fórmula de Boehm:**  $\text{Costo} = \text{CH} * \text{SM} * \text{TD}$

CH: es la cantidad de hombres

SM: es el salario semanal por persona

TD: es el tiempo de desarrollo total estimado para la iteración

$$\text{Costo} = 1 * 2000.00 * 4$$

$$\text{Costo} = 8000.00$$

Considerando como valor para SM de 2000.00 pesos (salario mínimo), donde la cantidad de hombres es 1 y la duración estimada es de 5 meses, entonces al

sustituir en la fórmula y calcular se obtiene un costo de \$ 8000.00 MN. Teniendo en cuenta que no es un proyecto costoso, resulta provechoso el empleo de tiempo y esfuerzo en la realización de este.

**Descripción de los resultados obtenidos.**

Posteriormente de desarrollar todo un proceso de pruebas de aceptación y estudio de factibilidad, obteniendo resultados satisfactorios. Se logró obtener un software cuyas funciones se encuentra en correspondencia con las especificaciones acordadas por el cliente y que además cumple con los requerimientos de rendimiento.

**Conclusiones del capítulo.**

Como conclusiones del capítulo se muestran los elementos de las pruebas de aceptación. Estas pruebas realizadas fueron de gran importancia para demostrar el buen funcionamiento del software y el cumplimiento de los requisitos planteados por el cliente.

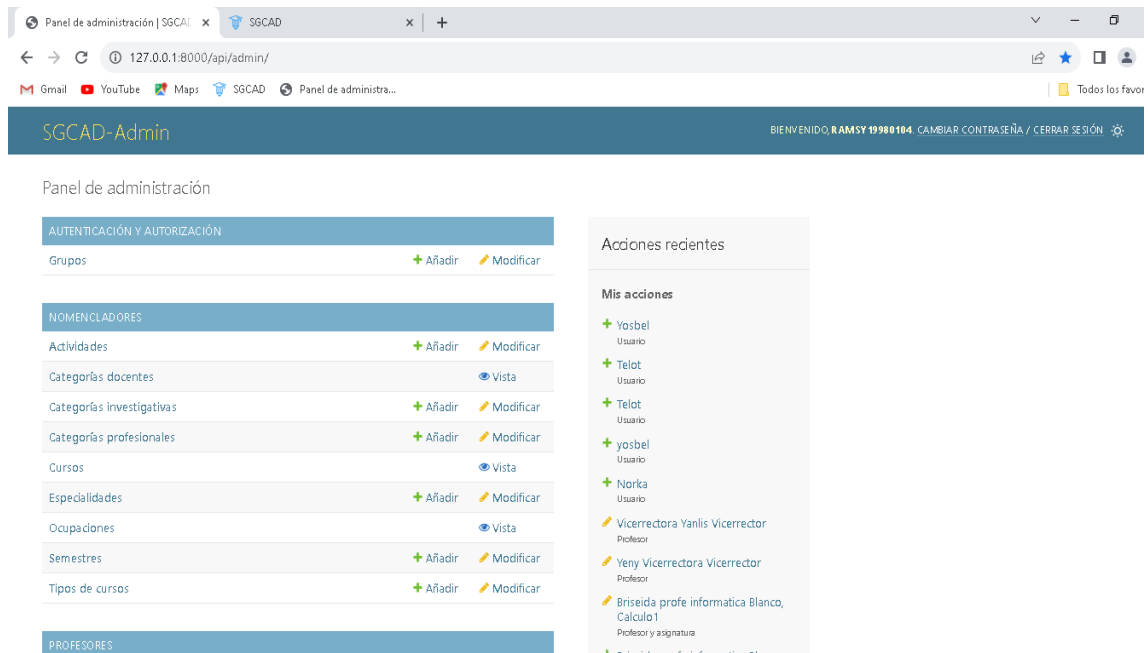
Se plantean las etapas necesarias para desarrollar el software según la metodología XP, con la excepción de las pruebas funcionales. Se definió el equipo de trabajo. Se crearon las historias de usuarios y las tareas de ingeniería planificadas en cada una de las iteraciones.

Se obtiene una aplicación web con una interfaz amigable, de alto rendimiento, seguridad, confiabilidad y usabilidad que contribuyen al proceso objeto de estudio

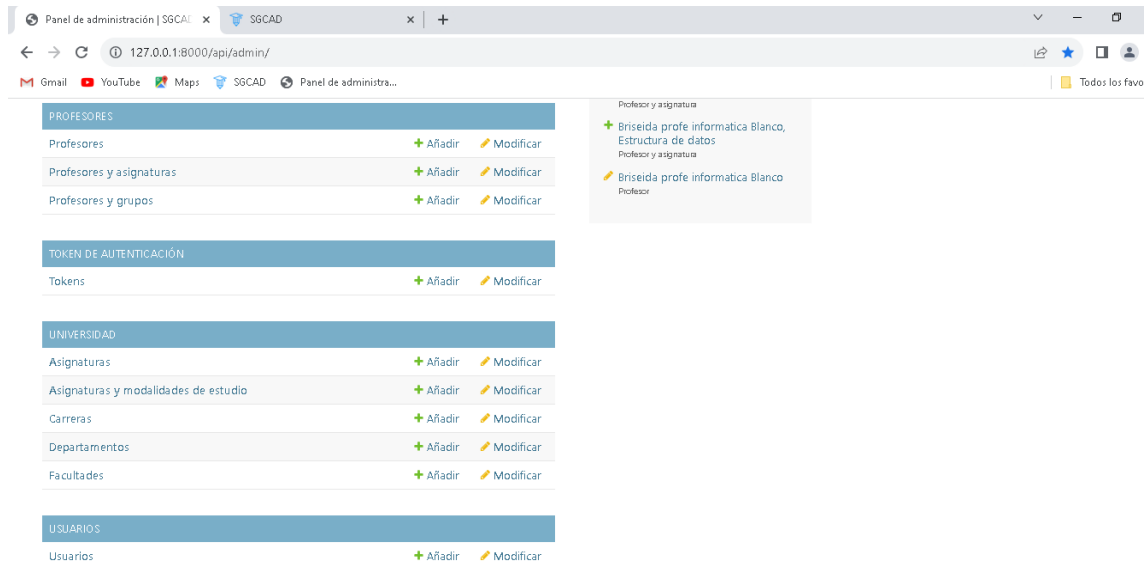
### Capítulo 3: Análisis de los resultados. Validación de la solución propuesta.

#### Introducción.

En este capítulo se realiza la validación de la solución propuesta, con el objetivo de comprobar el grado de cumplimiento de los requisitos funcionales iniciales y se verifica que no exista ninguna característica en el programa que no haya sido probada con la intención de mostrar un error no descubierto hasta entonces.



Panel de administración de la aplicación



Panel de administración | SGCAD

127.0.0.1:8000/api/admin/

Profesores

- Profesores + Añadir Modificar
- Profesores y asignaturas + Añadir Modificar
- Profesores y grupos + Añadir Modificar

TOKEN DE AUTENTICACIÓN

- Tokens + Añadir Modificar

UNIVERSIDAD

- Asignaturas + Añadir Modificar
- Asignaturas y modalidades de estudio + Añadir Modificar
- Carreras + Añadir Modificar
- Departamentos + Añadir Modificar
- Facultades + Añadir Modificar

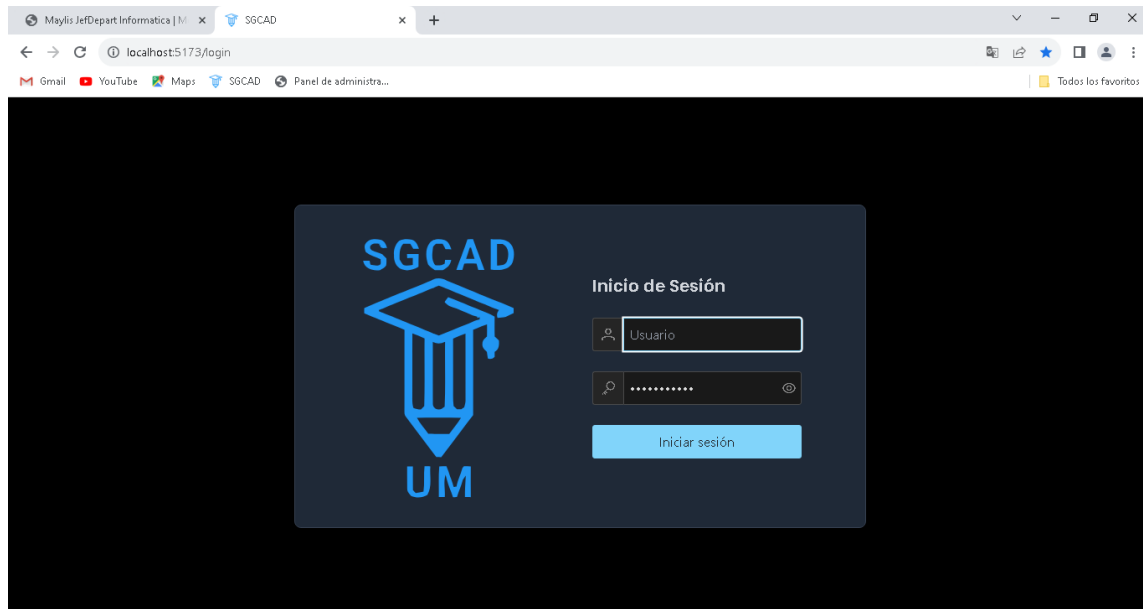
USUARIOS

- Usuarios + Añadir Modificar

Profesor y asignatura


- + Briseida profe informatica Blanco, Estructura de datos Profesor y asignatura
- Briseida profe informatica Blanco Profesor

### Panel de administración de la aplicación



Maylis JefDepart.Informatica | M x SGCAD

localhost:5173/login



SGCAD

UM

#### Inicio de Sesión

Usuario

Iniciar sesión

## Panel de autenticación

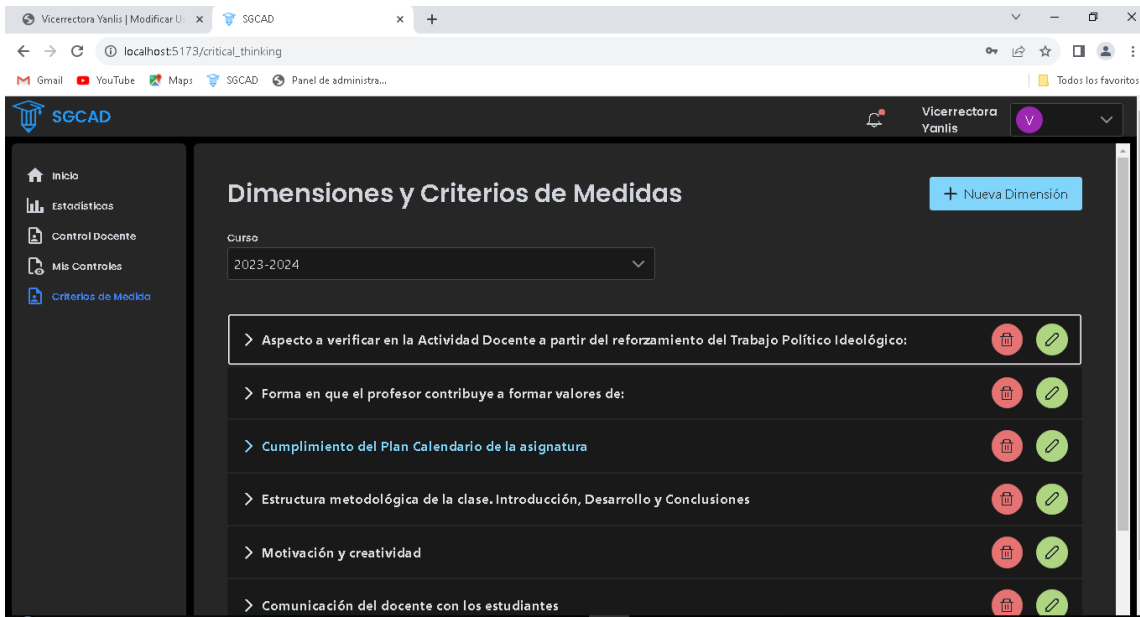
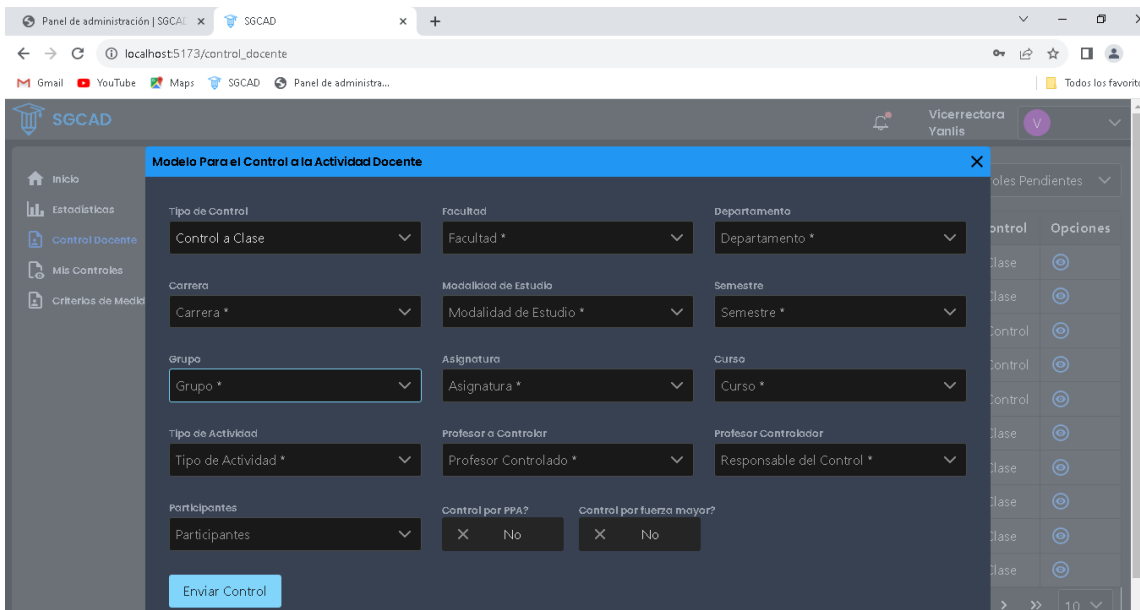
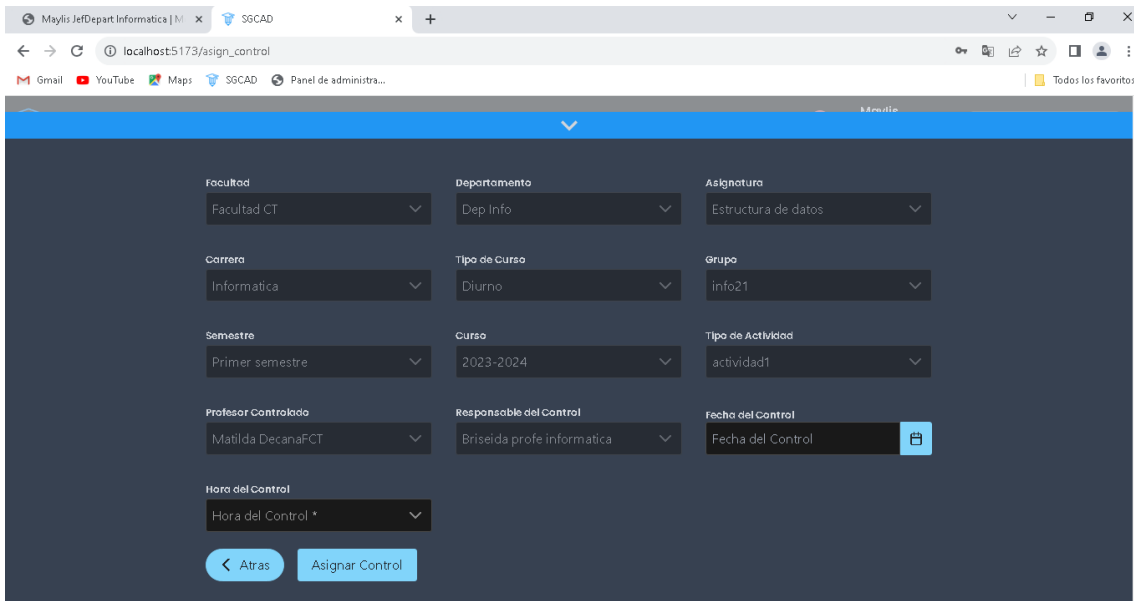


Imagen 1. Formulario Dimensiones, atributos y criterios de medida





**Imagen 2. Formulario del modelo para el control a la actividad docente**

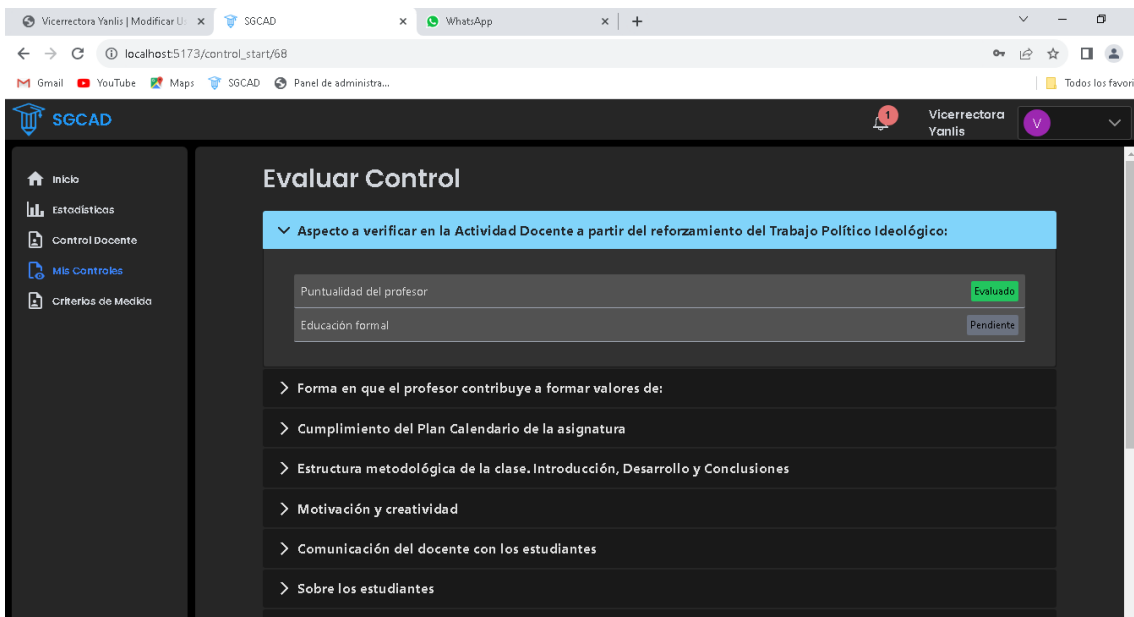


The screenshot shows a web browser window with the URL `localhost:5173/assign_control`. The page contains a form with the following fields:

- Facultad:** Facultad CT
- Departamento:** Dep. Info
- Asignatura:** Estructura de datos
- Carrera:** Informática
- Tipo de Curso:** Diurno
- Grupo:** info21
- Semestre:** Primer semestre
- Curso:** 2023-2024
- Tipo de Actividad:** actividad1
- Profesor Controlado:** Matilda DecanaFCT
- Responsable del Control:** Briseida profe informatica
- Fecha del Control:** Fecha del Control
- Hora del Control:** Hora del Control \*

At the bottom of the form, there are two buttons: **< Atras** and **Asignar Control**.

**Imagen 3 Formulario donde el jefe de departamento asigna la fecha y hora del control**



The screenshot shows a web browser window with the URL `localhost:5173/control_start/68`. The page is titled **Evaluar Control** and includes a sidebar with the following navigation items:

- Inicio
- Estadísticas
- Control Docente
- Mis Controles
- Criterios de Medición

The main content area displays a checklist under the heading **Aspecto a verificar en la Actividad Docente a partir del reforzamiento del Trabajo Político Ideológico:**

- Puntualidad del profesor: Evaluado
- Educación formal: Pendiente

Below the checklist, there are several expandable sections:

- Forma en que el profesor contribuye a formar valores de:
- Cumplimiento del Plan Calendario de la asignatura
- Estructura metodológica de la clase. Introducción, Desarrollo y Conclusiones
- Motivación y creatividad
- Comunicación del docente con los estudiantes
- Sobre los estudiantes

Imagen 4. Formulario para evaluar el control docente

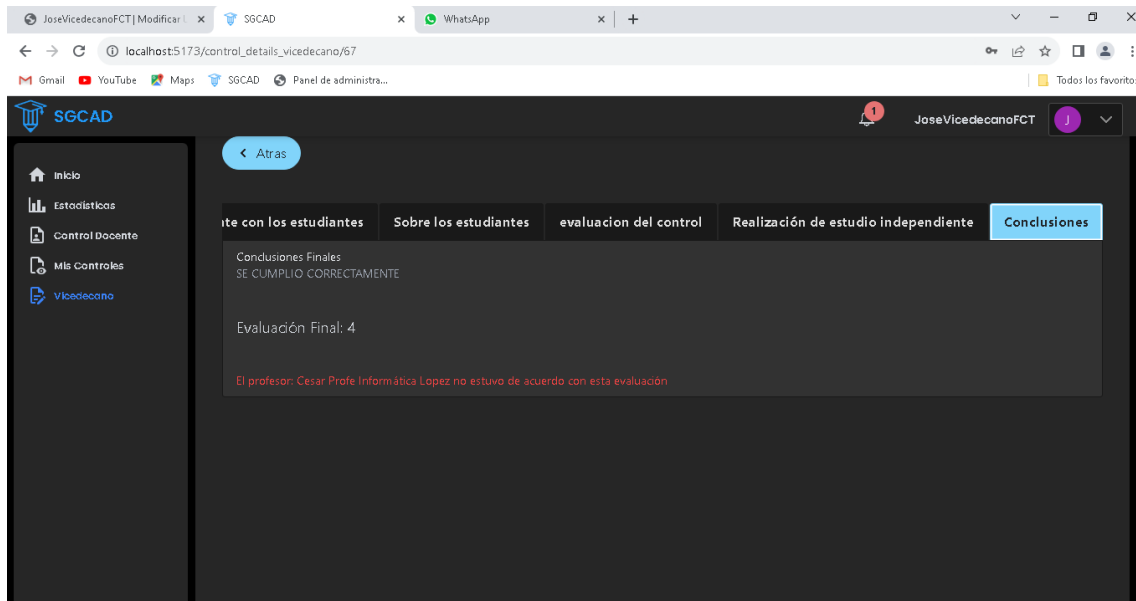


Imagen 5. Formulario para, observar cada ítem evaluado en el control con su evaluación final

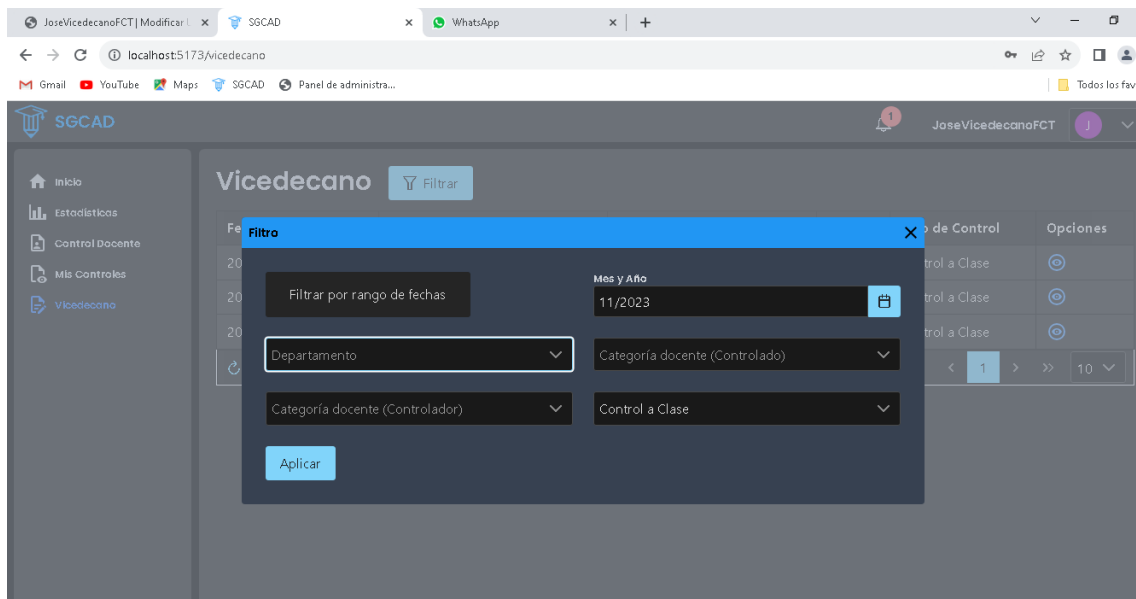


Imagen 6. Formulario para filtrar el control

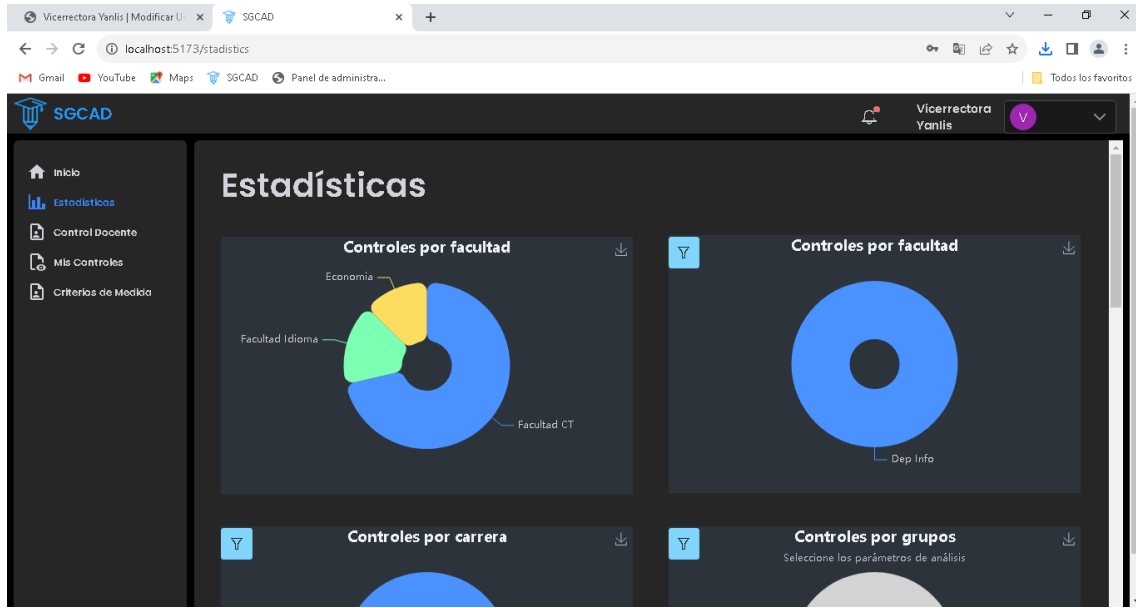
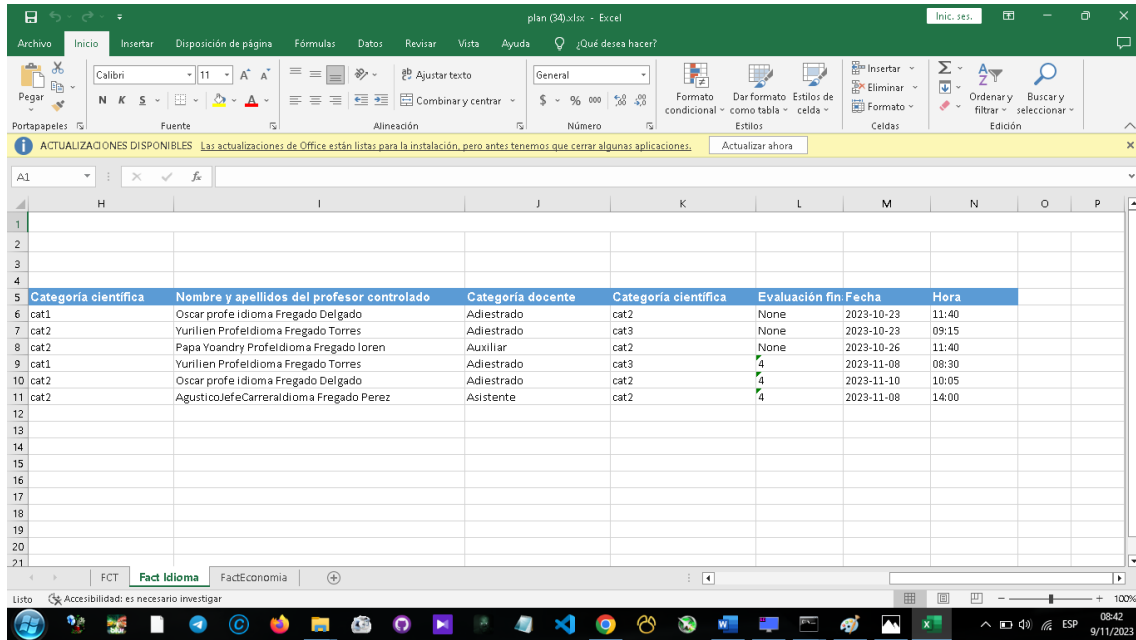


Imagen 7. Estadísticas

The Excel spreadsheet contains the following data:

Carrera	Tipo de curso	Año	Asignatura	Nombre y apellidos del profesor controlador	Categoría docente	Categoría científica
Idioma	Diurno	Primer año	Introducción a idioma	YarielJefDepartIdioma Ortiz Rodriguez	Auxiliar	cat1
Idioma	Diurno	Primer año	Introducción a idioma	Papa Yoandry Profefidioma Fregado loren	Auxiliar	cat2
Idioma	Diurno	Primer año	Introducción a idioma	YoandryDecan Idioma Fregado loren	Titular	cat2
Idioma	Diurno	Primer año	Introducción a idioma	YarielJefDepartIdioma Ortiz Rodriguez	Auxiliar	cat1
Idioma	Diurno	Primer año	Introducción a idioma	YoandryDecan Idioma Fregado loren	Titular	cat2
Idioma	Diurno	Primer año	Introducción a idioma	YoandryDecan Idioma Fregado loren	Titular	cat2
Total de controles realizados: 6						

Imagen 8. Formulario resumen general del proceso



Categoría científica	Nombre y apellidos del profesor controlado	Categoría docente	Categoría científica	Evaluación fin	Fecha	Hora
cat1	Oscar profe idioma Fregado Delgado	Adiestrado	cat2	None	2023-10-23	11:40
cat2	Yurilien Profefidioma Fregado Torres	Adiestrado	cat3	None	2023-10-23	09:15
cat2	Papa Yoandry Profefidioma Fregado Ioren	Auxiliar	cat2	None	2023-10-26	11:40
cat1	Yurilien Profefidioma Fregado Torres	Adiestrado	cat3	4	2023-11-08	08:30
cat2	Oscar profe idioma Fregado Delgado	Adiestrado	cat2	4	2023-11-10	10:05
cat2	AgusticoJefeCarreralidioma Fregado Perez	Asistente	cat2	4	2023-11-08	14:00

Imagen9. Formulario resumen general del proceso

**Conclusiones**

En esta investigación quedaron satisfechos los objetivos planteados y de esta forma se determinaron las siguientes conclusiones:

- o Se normalizaron los fundamentos teóricos que permitan la creación de un sistema web que gestione la información asociada a los controles a clase.
- o Se determinó la metodología Extreme Programming (XP) para el desarrollo del sistema web debido a sus características.
- o Se determinaron como herramientas más viables para la creación del proyecto el gestor de base de datos MySQL, lenguaje Python, como servidor web, *framework* Django,
- o Se implementó una aplicación web que facilite la gestión de los controles a las actividades docentes.
- o Se aplicó un sistema de pruebas que permitió detectar errores cometidos durante los procesos anteriores del desarrollo del sistema web para la gestión de los controles a las actividades docentes y corregirlos.

De forma general, se concluye que la aplicación web desarrollada es una herramienta confiable y fácil de utilizar pues eleva la calidad de la gestión de los controles a las actividades docentes, reduce los errores a cometer,

facilita y perfecciona el proceso y brinda reportes de interés a los directivos. Por estas razones, queda respondida la pregunta científica planteada en la introducción.

#### **Recomendaciones:**

Desde el punto de vista del alcance del presente trabajo y teniendo en cuenta el momento de desarrollo del mismo, se proponen las siguientes recomendaciones:

1. Aprovechar las posibilidades de información almacenadas en la aplicación e implementar funcionalidades como la notificación de nuevos controles realizados y la gestión de controles asignados a los profesores controlados y controladores.
2. Agregarle nuevos reportes que sean de interés para los usuarios que interactúen con la aplicación.
3. También se recomienda poner en funcionamiento el sistema web para automatizar la gestión del control a clase en la universidad de Matanzas

#### **Bibliografía**

- Álvarez, R. (2012). Introducción al HTML. Retrieved from <http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>.
- Cillero, M. (25 de junio de 2022). Pruebas de aceptación. Obtenido de Pruebas de aceptación: [manuel.cillero.es](http://manuel.cillero.es).
- Córdoba, J. (2015). Desarrollo y reutilización de componentes software y multimedia mediante lenguajes de guión. IFCD0210. IC Editorial.
- Dinahosting. (2022). Ayuda dinahosting. Obtenido de: <https://dinahosting.com/ayuda/queesmysql/#:~:text=M y S Q L% 20es% 20u n% 20sistem a% 20de,instalaciones% 20en% 20todo% 20el% 20m undo>.
- García D, C. (2016). Aplicación informática para gestión de reservas y estancias Hostal Azul Matanzas. Matanzas.
- GómezGarcía D, E. (2018). Desarrollo del sistema de requisiciones para la empresa hidroeléctrica AbanicoSA Aplicando el entorno de programación Node.js". B.S. thesis. Escuela Superior Politécnica de Chimborazo.
- Hori, A. (2015). An Oracle based on Image Comparison for Regression Testing of Web Applications. 639-645.

- IBM. (25 de noviembre de 2022). ¿Qué es la prueba de software? Obtenido de ¿Qué es la prueba de software?: [www.ibm.com](http://www.ibm.com).
- Jiménez Ortega, R. (2017). Curso de HTML5 desde cero (HTML5, CSS3, JavaScript).
- Joskowicz, J. (2008). Reglas y prácticas en Extreme Programming. Universidad de Vigo 22.
- Louden, K, C. (s.f.). Lenguajes de programación: principios y práctica. Cengage Learning Latin America. (2004, Entrevistador)
- Peña M, A. (2007). Arquitectura cliente-servidor. Retrieved from <http://www.desarrolloweb.com/articulos/arquitectura-cliente-servidor.html>.
- Pérez, J. (2019). Introducción a JavaScript.
- RocketTheme. (Metodologías de desarrollo de software). Metodologías de desarrollo de software.
- Rodríguez, F. (2018). *La gestión del control de las actividades docentes en la universidad: estado actual y perspectivas*. doi:45
- Saavedra E, L. (2009). Framework para el desarrollo de aplicaciones. Catalyst.
- Sommerville, I. (2016)). Software Engineering. 10th. Harlow, UK: Pearson.
- Valladarez Gaitan, R. (2016). Universidad Nacional Autónoma de Nicaragua, Managua UNAN-MANAGUA Recinto Universitario Rubén Darío (RURD) Facultad de Ciencias e Ingeniería Departamento de Computación En.
- Winesett, J. (2010). Agile web application development with Yii1. .
- Winesett, J. (2010). Agile web application development with Yii1. 1 and PHP5. Packt Publishing Ltd.
- You, E. (27 de 09 de 2022). Obtenido de Vuejs: <https://vuejs.org/>.
- Zanotti, A. (2016). El software libre su difusión en Argentina. Argentina.

ABEGG, M., R. BÖSCH, D. KÜKENBRINK AND F. MORSDORF Tree volume estimation with terrestrial laser scanning — Testing for bias in a 3D virtual environment. Agricultural and Forest Meteorology, 2023/03/15/ 2023, 331, 109348.

BEHUTIYE, W., P. KARHAPÄÄ, L. LÓPEZ, X. BURGUÉS, et al. Management of quality requirements in agile and rapid software

development: A systematic mapping study. *Information and Software Technology*, 2020/07/01/ 2020, 123, 106225.

BEHUTIYE, W., P. RODRÍGUEZ, M. OIVO, S. AARAMAA, et al. Towards optimal quality requirement documentation in agile software development: A multiple case study. *Journal of Systems and Software*, 2022/01/01/ 2022, 183, 111112.

BIRCHLER, C., N. GANZ, S. KHATIRI, A. GAMBI, et al. Cost-effective simulation-based test selection in self-driving cars software. *Science of Computer Programming 2023*, 226, 1-12.

BOUIHIA, B. AND M. BAHAJA 2019. An UML to OWL based approach for extracting Moodle's Ont. In *Proceedings of the Second International Conference on Intelligent Computing in Data Sciences (ICDS 2018) Second International Conference on Intelligent Computing in Data Sciences (ICDS 2018)* 2019 *Procedia Computer*, 313-322.

CASANOVA, M. A. Gestionar la Autonomía Pedagógica: Un Factor de Calidad Reconocido. REICE. *Revista Iberoamericana sobre Calidad, Eficacia y Cambio en Educación*, 2021, 19(2), 9-22.

CURCIO, K., R. SANTANA, S. REINEHR AND A. MALUCELLI Usability in Agile Software Development: A Tertiary Study. *Computer Standards & Interfaces*, 2019.

DEHLAGHI-GHADIM, A., A. BALADOR, M. H. MOGHADAM, H. HANSSON, et al. ICSSIM — A framework for building industrial control systems security testbeds. *Computers in Industry*, 2023/06/01/ 2023, 148, 103906.

EGGEN, H. O. AND K. A. HOLE. An evaluation of join-strategies in a distributed MySQL plugin architecture. *Norwegian University of Science and Technology* 2019.

FOKIDES, E. Development and testing of a scale for examining factors affecting the learning experience in the Metaverse. *Computers & Education: X Reality*, 2023/01/01/ 2023, 2, 100025.

GURSEL, E., B. REDDY, A. KHOJANDI, M. MADADI, et al. Using artificial intelligence to detect human errors in nuclear power plants: A case in operation and maintenance. *Nuclear Engineering and Technology*, 2023/02/01/ 2023, 55(2), 603-622.

- HASAN, D. A., B. K. HUSSAN, S. R. ZEEBAREE, D. M. AHMED, et al. The Impact of Test Case Generation Methods on the Software Performance: A Review. *International Journal of Science and Business*, 2021, 5(6), 33-44.
- KARAMBIR AND A. SHARMA A review of agile methodology in software development. *International Research Journal of Engineering and Technology (IRJET)*, 2016, 03(03), 1325-1329.
- LÓPEZ, L., X. BURGUÉS, S. MARTÍNEZ-FERNÁNDEZ, A. M. VOLLMER, et al. Quality measurement in agile and rapid software development: A systematic mapping. *Journal of Systems and Software*, 2022/04/01/ 2022, 186, 1-26.
- MELLOR, A. *Test-Driven Development with Java. Create higher-quality software by writing tests first with SOLID and hexagonal architecture*. Edition ed. Birmingham -UK: Packt Publishing, 2023. 348 p.
- MENGLUN KUO, T., T. CHIN-CHUNG AND W. JYUN-CHENG Linking web-based learning self-efficacy and learning engagement in MOOCs: The role of online academic hardiness. *The Internet and Higher Education*, 2021, 51, 1-15.
- OZKAYA, M. AND F. ERATA A survey on the practical use of UML for different software architecture viewpoints. *Information and Software Technology*, 2020, 121, 1-27.
- PATIÑO-TORO, O. N., A. VALENCIA-ARIAS, A. FERNÁNDEZ-TORO, A. JIMÉNEZ-GUZMÁN, et al. Proposed methodology for designing and developing MOOCs for the deaf community. *Heliyon*, 2023/10/01/ 2023, 9(10), e20456.
- PENG, Y., J. HAN, Z. ZHANG, L. FAN, et al. The Tong Test: Evaluating Artificial General Intelligence Through Dynamic Embodied Physical and Social Interactions. *Engineering*, 2023/08/09/ 2023.
- SALAHIRAD, A., G. GAY AND E. MOHAMMADI Mapping the structure and evolution of software testing research over the past three decades. *The Journal of Systems & Software*, 2023, 195, 1-25.
- SALDÍAS KIEFER, L. AND D. REYES-LILLO Evaluación de usabilidad de sitios web de bibliotecas de universidades públicas chilenas a partir del método SIRIUS. *e-Ciencias de la Información*, 2021, 11(1), 1-18.



- SERLING-BOYD, N. AND E. M. MILOSLAVSKY Enhancing the Inpatient Consultation Learning Environment to Optimize Teaching and Learning. *Rheum Dis Clin North Am*, Feb 2020, 46(1), 73-83.
- SFAXI, L. AND M. M. BEN AISSA DECIDE: An Agile event-and-data driven design methodology for decisional Big Data projects. *Data & Knowledge Engineering*, 2020, 130, 101862.
- STEINRÜCKE, J., B. P. VELDKAMP AND T. DE JONG The effect of self-reflection on information usage and information literacy in a digital serious game. *Computers and Education Open*, 2023/12/01/ 2023, 4, 100133.
- STRADOWSKI, S. AND L. MADEYSKI Exploring the challenges in software testing of the 5G system at Nokia: A survey. *Information and Software Technology*, 2023, 153, 1-18.
- TULLO, D., J. FAUBERT AND A. BERTONE Distinctive patterns of Multiple Object-Tracking performance trajectories in youth with deficits in attention, learning, and intelligence. *Research in Developmental Disabilities*, 2023/02/01/ 2023, 133, 104402.
- WAMEMA, J., A. ALUNYU, M. AMIYO AND J. NABUKENYA Enterprise architecture requirements for standardising digital health in Uganda's health system. *Health Policy and Technology*, 2023/09/15/ 2023, 100805.
- WIDI ANGGORO, P. D. Design of Web Virtual Reality for Job Interview Preparation Simulation. *Lontar Komputer*, 2020, 11(3), 132-143.
- WU, S., A. HADACHI, C. LU AND D. VIVET MOTT: A new model for multi-object tracking based on green learning paradigm. *AI Open*, 2023/01/01/ 2023, 4, 145-153.
- WU, Y. 2020. Anti-patterns for Java Automated Program Repair Tools. In *Proceedings of the 2020 35th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, Australia 2020 Association for Computing Machinery, 1-10.
- YANG, T., Z. JIANG, Y. SHANG AND M. NOROUZI Systematic review on next-generation web-based software architecture clustering models. *Computer Communications*, 2021, 167, 63-74.
- ZHI, Q., L. GONG, J. REN, M. LIU, et al. Element quality indicator: A quality assessment and defect detection method for software requirement specification. *Heliyon*, 2023/05/01/ 2023, 9(5), e16469.

ZNEIKA, M. Querying Semantic Web/Linked Data Graphs Using Summarization. University of Paris Seine, 2019.