

PORTADA

**UNIVERSIDAD TÉCNICA DE AMBATO**

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL**

**CARRERA DE SOFTWARE**

**Tema:**

**IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DOCENTE EN LA UNIDAD EDUCATIVA RUMIÑAHUI UBICADA EN EL CANTÓN AMBATO, PROVINCIA DE TUNGURAHUA**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del título de Ingeniero en Software

**ÁREA:** Gestión de las tecnologías de la información

**LÍNEA DE INVESTIGACIÓN:** Tecnologías de la información y sistemas de control

**AUTOR:** Andrés Patricio Tapia González

**TUTOR:** Ing. Hernán Fabricio Naranjo Ávalos Mg.

**Ambato - Ecuador**

**Agosto – 2024**

# APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DOCENTE EN LA UNIDAD EDUCATIVA RUMIÑAHUI UBICADA EN EL CANTÓN AMBATO, PROVINCIA DE TUNGURAHUA, desarrollado bajo la modalidad Proyecto de Investigación por el/la señor/señorita Andrés Patricio Tapia González, estudiante de la Carrera de Ingeniería en Software de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, agosto 2024.

------------------------------------------

Ing. Hernán Fabricio Naranjo Ávalos, Mg.

TUTOR

# AUTORÍA

El presente trabajo de titulación con el tema: IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DOCENTE EN LA UNIDAD EDUCATIVA RUMIÑAHUI UBICADA EN EL CANTÓN AMBATO, PROVINCIA DE TUNGURAHUA es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2024.

------------------------------------------

Andrés Patricio Tapia González

C.C. 1803857646

AUTOR

# DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, agosto 2024.

------------------------------------------

Andrés Patricio Tapia González

C.C. 1803857646

AUTOR

# APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el/la señor/señorita \*(nombres y apellidos), estudiante de la Carrera de , de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado \*(COLOCAR TEMA EN MAYÚSCULAS), nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2023.

(debe ir el mes que se presenta papeles para defensa)

------------------------------------------

Ing. Elsa Pilar Urrutia Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

(siempre va la Decana)

------------------------------- -------------------------------

Ing.\* Ing. Nombres y apellidos, Mg./PhD./MSc.

PROFESOR CALIFICADOR PROFESOR CALIFICADOR

# DEDICATORIA

*Redactar dedicatoria en no más de una hoja.*

# AGRADECIMIENTO

*Redactar el agradecimiento en no más de una hoja.*

# ÍNDICE GENERAL DE CONTENIDOS

[PORTADA i](#_Toc164979792)

[APROBACIÓN DEL TUTOR ii](#_Toc164979793)

[AUTORÍA iii](#_Toc164979794)

[DERECHOS DE AUTOR iv](#_Toc164979795)

[APROBACIÓN DEL TRIBUNAL DE GRADO v](#_Toc164979796)

[DEDICATORIA vi](#_Toc164979797)

[AGRADECIMIENTO vii](#_Toc164979798)

[ÍNDICE GENERAL DE CONTENIDOS viii](#_Toc164979799)

[ÍNDICE DE TABLAS xi](#_Toc164979800)

[ÍNDICE DE FIGURAS xii](#_Toc164979801)

[ÍNDICE DE ANEXOS xiv](#_Toc164979802)

[RESUMEN EJECUTIVO xvi](#_Toc164979803)

[ABSTRACT xvii](#_Toc164979804)

[CAPÍTULO I. MARCO TEÓRICO 1](#_Toc164979805)

[1.1 Tema de investigación 1](#_Toc164979806)

[1.1.1 Planteamiento del problema 1](#_Toc164979807)

[1.2 Antecedentes investigativos 2](#_Toc164979808)

[1.3 Fundamentación teórica 8](#_Toc164979809)

[1.4 Objetivos 14](#_Toc164979810)

[1.4.1 Objetivo general 14](#_Toc164979811)

[1.4.2 Objetivos específicos 14](#_Toc164979812)

[CAPÍTULO II. METODOLOGÍA 15](#_Toc164979813)

[2.1 Materiales 15](#_Toc164979815)

[2.2 Métodos 15](#_Toc164979816)

[2.2.1 Modalidad de la investigación 15](#_Toc164979817)

[2.2.2 Población y muestra 16](#_Toc164979818)

[2.2.3 Recolección de información 16](#_Toc164979819)

[2.2.4 Procesamiento y análisis de datos 16](#_Toc164979820)

[CAPÍTULO III. RESULTADOS Y DISCUSIÓN 22](#_Toc164979821)

[3.1 Descripción de procesos de la institución 22](#_Toc164979823)

[3.2 Estudio comparativo de metodologías de desarrollo ágiles 36](#_Toc164979824)

[3.3 Estudio comparativo de Frameworks de desarrollo Frontend 39](#_Toc164979825)

[3.4 Estudio comparativo de Frameworks de desarrollo Backend 41](#_Toc164979826)

[3.5 Descripción de dispositivos biométricos 45](#_Toc164979827)

[3.6 Diseño de arquitectura de la aplicación 46](#_Toc164979828)

[3.7 Desarrollo 52](#_Toc164979829)

[3.7.1 Fases de la metodología Crystal Clear 48](#_Toc164979830)

[3.7.2 Primera iteración 60](#_Toc164979831)

[CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES 67](#_Toc164979832)

[4.1 Conclusiones 67](#_Toc164979834)

[4.2 Recomendaciones 67](#_Toc164979835)

[REFERENCIAS BIBLIOGRÁFICAS 68](#_Toc164979836)

[ANEXOS 74](#_Toc164979837)

# ÍNDICE DE TABLAS

[Tabla 1. Requerimientos funcionales del sistema 18](#_Toc164979838)

[Tabla 2. Requerimientos no funcionales 20](#_Toc164979839)

[Tabla 3. Cuadro comparativo de metodologías ágiles 37](#_Toc164979840)

[Tabla 4. Cuadro comparativo de Frameworks Frontend 40](#_Toc164979841)

[Tabla 5. Cuadro comparativo de Frameworks Backend 41](#_Toc164979842)

[Tabla 6. Roles del equipo de trabajo 49](#_Toc164979843)

[Tabla 7. Matriz de tareas 51](#_Toc164979844)

[Tabla 8. Matriz de prueba 52](#_Toc164979845)

[Tabla 9. Matriz de la tarea 01 **¡Error! Marcador no definido.**](#_Toc164979846)

[Tabla 10. Matriz de la tarea 02 63](#_Toc164979847)

[Tabla 11. Matriz de prueba 01 64](#_Toc164979848)

[Tabla 12. Matriz de prueba 02 65](#_Toc164979849)

[Tabla 13. Matriz de prueba 03 65](#_Toc164979850)

# ÍNDICE DE FIGURAS

[Figura 1. Diagrama de proceso de registro de asistencia 22](#_Toc164979851)

[Figura 2. Diagrama de proceso de marcaje de asistencia no presencial 23](#_Toc164979852)

[Figura 3. Diagrama de proceso de justificación de no marcaje de asistencia 27](#_Toc164979853)

[Figura 4. Diagrama de proceso de generación de reportes de asistencia 29](#_Toc164979854)

[Figura 5. Diagrama de proceso de generación de reportes de personal 33](#_Toc164979855)

[Figura 6. Arquitectura del sistema 46](#_Toc164979856)

[Figura 7. Diseño de la pantalla de inicio de sesión 63](#_Toc164979857)

[Figura 8. Diseño de la pantalla de inicio del módulo de empleados 64](#_Toc164979858)

# ÍNDICE DE ANEXOS

[Anexo A. Entrevista de levantamiento de requerimientos 74](#_Toc162466726)

[Anexo B. Entrevista de refinamiento de requerimientos 78](#_Toc162466727)

[Anexo C. Documento de formalización de requerimientos - IEEE 830 87](#_Toc162466728)

[Anexo D. Diagrama de proceso de registro de asistencia 88](#_Toc162466729)

[Anexo E. Diagrama de proceso de reporte de asistencia de personal 89](#_Toc162466730)

[Anexo F. Diagrama de proceso de registro de asistencia no presencial 90](#_Toc162466731)

[Anexo G. Diagrama de proceso de justificación de inasistencia 91](#_Toc162466732)

[Anexo H. Diagrama de proceso de generación de reportes de personal 92](#_Toc162466733)

# RESUMEN EJECUTIVO

El resumen ejecutivo constará de máximo 250 palabras.

**Palabras claves:** Primera letra de la primera palabra en mayúscula, las siguientes palabras con minúsculas, a excepción de nombres propios que inician con letra mayúscula.

# ABSTRACT

The executive summary will consist of a maximum of 250 words.

**Keywords:** First letter of the first word in capital letters, the following words with lowercase, except for proper names that start with a capital letter.

# CAPÍTULO I. MARCO TEÓRICO

## Tema de investigación

IMPLEMENTACIÓN DE UN SISTEMA DE CONTROL DOCENTE EN LA UNIDAD EDUCATIVA RUMIÑAHUI UBICADA EN EL CANTÓN AMBATO, PROVINCIA DE TUNGURAHUA

### Planteamiento del problema

En la Sociedad de la Información en Ecuador se observa un avance significativo en la adopción de tecnologías que mejoran la calidad de vida de la población y optimizan la gestión administrativa en instituciones públicas. Según [1] los sistemas web utilizados en el ámbito educativo representan una herramienta que mejora la calidad de vida de la población ecuatoriana y optimiza la administración de la información de las instituciones públicas. En el informe de [2] se detalla que la implementación de sistemas automatizados ha potenciado la agilidad y productividad de las instituciones. La integración de estos sistemas posibilita a las instituciones trabajar con mayor eficiencia, proporcionando agilidad y versatilidad a los procesos. Esta característica ha generado resultados más eficientes en términos de la fuerza laboral de la institución.

En [3] el Ministerio de Educación Ecuatoriano informa que durante el periodo lectivo entre los años 2022-2023, existen 12341 instituciones educativas fiscales. Estos centros educativos, además de llevar a cabo sus actividades pedagógicas de forma ordenada y sistemática, también deben administrar eficientemente todas sus actividades administrativas. En la provincia de Tungurahua, hay actualmente 185 instituciones educativas bajo el régimen fiscal que enfrentan las mismas necesidades mencionadas anteriormente. La gestión de la asistencia se destaca como una actividad administrativa fundamental en toda institución. Según [4] es imprescindible que toda empresa cuente con un control adecuado que supervise el tiempo de asistencia de su personal. Por lo tanto, resulta esencial establecer mecanismos para detectar cualquier irregularidad en la presencia de los trabajadores. En el documento de [5] se señala que la puntualidad y la asistencia de un empleado pueden influir en la productividad de la fuerza laboral de la empresa dado que, si un empleado llega a tiempo a su lugar de trabajo, puede cumplir con eficiencia sus responsabilidades asignadas. Esta noción cobra mayor relevancia en una institución educativa, donde los docentes representan figuras clave en la formación de la juventud ecuatoriana. Cualquier fallo en el cumplimiento de sus responsabilidades afectaría significativamente a toda la comunidad educativa.

La carencia de automatización en el proceso administrativo de control docente en una institución educativa puede ocasionar un control ineficiente del tiempo de trabajo del personal que labora en la institución. Según [6] el control manual de asistencia requiere la participación de personal designado para llevar a cabo los procedimientos de verificación de la asistencia, lo que vuelve el proceso de supervisión propenso a errores involuntarios y genera actividades adicionales. Esta situación resulta en ineficiencia y falta de seguridad en el control docente. En la Unidad Educativa Rumiñahui, ubicada en la ciudad de Ambato, Tungurahua, hay actualmente tres relojes biométricos distribuidos en cada campus. Estos dispositivos no están conectados a un sistema específico, lo que impide la integración de datos y requiere la recolección manual de información y generación de reportes, volviendo ineficiente el proceso de supervisión de la asistencia del personal docente que labora en la institución.

## Antecedentes investigativos

En [7] se presenta una solución que reduce la evasión de marcaciones de registro de asistencia del personal docente de la Universidad Metropolitana del Ecuador. El trabajo se elabora bajo un ambiente web de software libre. Para la construcción de la solución se utilizó PHP (Hypertext Preprocessor) como lenguaje de programación utilizando Laravel como Framework y MariaDB como gestor de bases de datos. Adicionalmente el autor detalla que se utilizó Xampp como mecanismo para la implementación de un servidor web. El autor asegura que mediante la utilización de un sistema de alertas tempranas el control de asistencia de personal de la institución será más efectivo. Además, concluye que un sistema debe ser escalable en caso de que en un futuro cambie el sistema de marcaciones o surjan requerimientos adicionales por parte del cliente.

En [8] se plantea una solución que permite reducir el tiempo de control de asistencia, permisos y licencias en la Escuela Superior Militar de Aviación Cosme Rennella Barbato. El enfoque de la investigación es de tipo exploratorio. Como herramienta de adquisición de la información se utilizó la encuesta y la entrevista. En cuanto a las herramientas de desarrollo utilizadas el autor detalla que utilizo como gestor de base de datos MySQL (My Structured Query Language). Además, indica que utilizo HTML (HyperText Markup Language), JavaScript y CSS (Cascading Style Sheets) para construir la interfaz web, en conjunto con JQuery y Ajax como librerías de desarrollo. El autor concluye que la implementación de un sistema web reduce notablemente el tiempo utilizado para la gestión de permisos, justificaciones y licencias de vacaciones de la institución.

En [9] se presenta una solución web mediante la utilización de huellas dactilares para el control de la asistencia del personal de la empresa SAC. Presenta una metodología aplicada, descriptiva y no experimental. Como metodología de desarrollo se utilizó XP (Extreme Programming). El autor utiliza el Modelo Vista Controlador utilizando PHP como lenguaje de programación y Bootstrap como librería de estilos. Concluye que la utilización de una aplicación web de gestión de asistencia de personal brinda rapidez, claridad, seguridad y comodidad en el procesamiento de la información relacionada con el control de asistencia.

En [10] se presenta una solución integrada para la gestión de horarios y control de asistencia del personal docente del preuniversitario de la UNAMAD. El autor indica que la metodología de investigación que utilizo para este proyecto es aplicada preexperimental. Se utilizo como método de adquisición de información la encuesta aplicada a 3 administrativos y 14 docentes de la institución. Como metodología de desarrollo utilizo Scrum para la construcción del sistema. En cuanto a las tecnologías utilizadas se utilizó MySQL como gestor de base de datos. Para la construcción de la interfaz web de la aplicación el autor opto por utilizar NodeJS y AngularJS bajo el lenguaje de programación JavaScript. Adicional a esto utilizo el lenguaje de programación Java para tareas puntuales. El autor concluye que el sistema presentado mejoro la gestión de horarios y del personal docente de la institución.

En [11] se detalla un proyecto de control de asistencia mediante la utilización de datos biométricos en un almacén de electrodomésticos. Este proyecto trabajo bajo una metodología de proyecto fiable. Como arquitectura trabaja bajo un esquema de elaboración propia en el cual se tiene el punto de registro donde el autor indica que se va a encontrar instalado el sistema, panel de control donde se encontraran conectados los periféricos, sistema de detección que es el dispositivo mediante el cual el empleado registra su asistencia y host y sistema de software donde se aloja la base de datos y/o la aplicación que controla el sistema. En cuanto a las tecnologías utilizadas el autor detalla que utilizó MySQL como gestor de base de datos y Java como lenguaje de programación Concluye que la utilización de tecnologías biométricas son una mediad de captación de datos recomendable debido a su fiabilidad y rapidez.

En [12] se presenta el diseño e implementación de un prototipo de marcación de control dactilar utilizando hardware de bajo costo y administrado mediante una aplicación web. La metodología de investigación aplicada al presente proyecto es deductiva y experimental. Como herramientas de recolección de información el autor utilizo la observación, entrevista y recolección manual de información. Como lenguajes de programación el autor utilizo Python, MongoDB como gestor de base de datos y VueJS para la construcción de la interfaz de usuario. El autor construye un lector biométrico de bajo costo para el registro de la asistencia del personal. Concluye que un sistema de control de personal utilizando un biométrico construido de cero presenta un sistema organizado, dinámico e innovador.

En [13] se presenta un sistema de control de asistencia para la empresa Construcciones SAC. Se detalla como metodología de investigación un tipo aplicado, descriptiva correlacional y no experimental. Como método de adquisición de información se utilizó la encuesta aplicada a 40 trabajadores de la empresa de Construcciones SAC. Se realizo un sistema web con la utilización de los lenguajes de programación PHP y C#. El autor concluye que el 100% de la muestra encuestada está totalmente de acuerdo en contar con un instrumento que se adapte a las necesidades empresariales de control de asistencia. De igual manera detalla que la utilización de un sistema de control de asistencia se evitan errores en los pagos y multas del personal que labora en la empresa en cuestión.

En [14] se presenta un sistema de control de asistencia para la Incubadora de Microempresas Productivas Impro IFD. Se detalla como metodología de investigación un tipo interpretativo. Como método de adquisición de información el autor detalla que utilizó la entrevista, pero no detalla la población a la cual se la aplicó. Para la construcción del sistema se utilizó la metodología XP, PostgreSQL como gestor de base de datos y los lenguajes de programación Visual Basic con el Framework .Net y C++. El autor concluye que la aplicación de este sistema a la institución cumplió con los requerimientos de la empresa cumpliendo el objetivo general de la investigación. Sin embargo, detalla que existió inconvenientes a la hora de definir ciertos procesos de recursos humanos que no existían dentro de la normativa de la institución.

En [15] presenta un sistema de control de asistencia basado en marcaje biométrico aplicado en la Gerencia Regional de Educación Moquegua. Esta investigación es de tipo aplicada preexperimental. Para la construcción del sistema se utiliza el lenguaje de programación PHP. Adicionalmente se utiliza Ajax para la construcción del Frontend de la aplicación. Como base de datos se utiliza PostgreSQL. En cuanto a los resultados se realizó un estudio descriptivo e inferencial en el cual se realiza pre y post la implementación del sistema a una muestra conformada por 60 trabajadores administrativos. Concluye que un sistema de control de asistencia biométrico contribuye en la mejora de procesos y en la disminución de tiempo efectuado para la generación de reportes de tiempos de trabajo de los empleados de la institución.

En [16] se presenta una solución de control de asistencia para la empresa HGV. En cuanto a la metodología de investigación el autor detalla que es de tipo descriptiva y cuantitativa. Además, señala que no es experimental. Como medio de recolección de la información se utilizó la encuesta, la cual fue aplicada a una muestra de 15 trabajadores ubicados en una sede de la empresa. El autor utilizo el marco de trabajo Scrum para la construcción de la solución. Como tecnologías se detalla que se utilizó MySQL como gestor de base de datos y PHP como lenguaje de programación utilizando Bootstrap como Framework HTML, CSS y JavaScript. Como dispositivo biométrico el autor utilizo el ZKTeco K14 el cual permite una comunicación IP/TCP (Internet Protocol / Transmission Control Protocol). Concluye que un sistema web de control de asistencia permite mejorar el control de asistencia de una institución.

En [17] se detalla a un sistema biométrico que utiliza una aplicación móvil para el control de la asistencia de los estudiantes en el Instituto Superior Tecnológico Americana del Cusco en Perú. El estudio aplicado fue una investigación aplicada experimental y cuantitativo. El proyecto se ejecutó bajo un modelo de arquitectura cliente-servidor. Se utilizo el lenguaje de programación Python en el Frontend de la aplicación Web mediante la utilización de Django. Para la construcción Frontend de la aplicación móvil se utiliza el Framework Flutter el cual trabaja bajo el lenguaje de programación Dart. El estudio indica que se logró disminuir el índice de ausentismo en un promedio de los estudiantes de un 28.54% a un 13.5%.

En [18] presenta un sistema web para mejorar el control de asistencia en la dirección de educación de Ancash. En cuanto a la metodología se detalla que trabaja bajo un marco aplicado y descriptivo, preexperimental. Para la obtención de los resultados se aplicó una encuesta pre y post la implementación de la propuesta donde se utilizó una prueba de Chi Cuadrado para el análisis de los datos obtenidos. Para la construcción del sistema se utilizó Java Server Pages en la aplicación web, Python para el análisis de los datos biométricos, MySQL como motor de base de datos y Apache Tomcat y Glasfish Server como mecanismos para la implantación del sistema. Adicionalmente se detalla que se utilizó una arquitectura cliente servidor. El autor concluye que la implantación de un sistema web representa una mejora en el control de asistencia de la institución. Así mismo menciona que un sistema web de control de personal logra reducir el tiempo en la emisión de reportes relacionados con esta actividad.

En [19] se presenta un sistema que permita el registro de la asistencia del personal administrativo y docente del Instituto Superior Tecnológico San Gabriel. El autor utiliza la entrevista como método de adquisición de datos. Para la construcción del sistema se utilizaron los lenguajes de programación PHP y C# además de MySQL como gestor de base de datos. Para el registro de los datos biométricos el autor indica que trabaja con dispositivos Hikvision mediante la utilización de su SDK para desarrolladores el cual expone las funcionalidades necesarias para la finalidad de la investigación. Adicionalmente el autor indica que trabajo bajo una metodología de desarrollo XP. También indica que la arquitectura que utiliza es la de cliente-servidor. Concluye que la implementación de un sistema de asistencia en la web permite acceder en tiempo real a la información generada por el dispositivo biométrico, lo cual posibilita la generación de reportes de forma eficiente.

En [20] se presenta un sistema web que permita controlar de mejor manera la asistencia del personal administrativo del Tecnológico Publico Eleazar Guzmán Barrón. La investigación presenta una metodología aplicada, descriptiva y no experimental. Como mecanismo para la recolección de la información se utilizó la encuesta y las entrevistas aplicadas a los 31 administrativos. Para la construcción del sistema se utilizó el Framework Laravel bajo el lenguaje de programación PHP y MySQL como gestor de base de datos. Se concluye que un sistema de control de asistencia mejora notablemente la gestión del personal de la institución.

En [21] se presenta un aplicativo móvil mediante tecnología biométrica para el control de asistencia del personal docente de la Institución Educativa San Antonio de Padua. Esta investigación se elaboró bajo una metodología cuantitativa aplicada. Para la construcción del sistema se utilizó React Native como Framework de desarrollo Frontend bajo el lenguaje de programación TypeScript y como gestor de base de datos utiliza Firebase. Como mecanismo de recolección de datos biométricos se utiliza el lector de huellas que se encuentra en los dispositivos móviles de los docentes. El autor concluye que un aplicativo móvil disminuye notablemente el tiempo en la generación de reportes y facilita el control de asistencia para el personal docente de la institución.

## Fundamentación teórica

A continuación, se presenta la base teórica que sustenta el presente trabajo investigativo, ofreciendo contexto y argumentos válidos para la consecución de los objetivos planteados.

**Ingeniería de Software**

Según [22] la ingeniería de software se define como una disciplina que comprende todos los aspectos de la producción de software desde sus etapas iniciales de especificación del sistema hasta el mantenimiento del producto terminado. En este contexto el autor destaca dos definiciones clave:

Disciplina de la ingeniería: Según [22] hace referencia a la capacidad de los ingenieros para aplicar teorías, métodos y herramientas que sean convenientes de manera selectiva para la resolución de un problema en específico.

Aspectos de producción de software: En este sentido [22] destaca que la ingeniería de software no es solamente los procesos técnicos de desarrollo de software, sino, también comprende actividades como la gestión de actividades en un proyecto de software y aplicación y desarrollo de metodologías que brinden apoyo al proceso de producción de software.

En síntesis, los ingenieros de software tienen un enfoque técnico y organizado en la construcción de sistemas, enfatizando la creación de software de calidad guiándose en metodologías y técnicas para la creación de estos. Adicionalmente, se involucran en el proceso de creación desde fases de planificación hasta mantenimiento del producto terminado.

**Aplicación Web**

Según [23] es un producto que se aloja en un servidor y se ejecuta en un navegador web. Este tipo de aplicaciones permiten la portabilidad de un sistema, ya que, no necesita de un sistema operativo en específico para poderse ejecutar de manera correcta. La arquitectura más conocida en una aplicación es la de cliente – servidor donde el cliente es el usuario que realiza peticiones de información al servidor que es donde se encuentra alojado el sistema.

**Frontend**

Según [24] el término *Frontend* hace referencia a la interfaz gráfica que el usuario utiliza para interactuar de forma directa con las opciones de una aplicación web. En otras palabras, el Frontend de una aplicación es la cual permite al usuario interactuar con los datos de un sistema de forma sencilla y amigable.

**Backend**

Según [24] el término *Backend* o también llamado *Servidor* es donde se maneja la funcionalidad de un sistema web. Este se comunica con el Frontend de la aplicación mediante la utilización de procedimientos HTTP. El Backend es el encargado de procesar la solicitud del Frontend de un sistema web de manera eficiente y segura.

**Base de Datos**

En [25] se define a una base de datos como una recopilación estructurada de datos, que usualmente se almacena de manera electrónica en un sistema informático. Los que se suelen utilizar usualmente organizados en filas y columnas pertenecientes a diferentes tablas aumentando de esta forma la eficacia en la consulta de datos. De esta forma estos datos se pueden acceder, gestionar, modificar, controlar y eliminar utilizando un Lenguaje de Consulta Estructurada (SQL). Este lenguaje permite realizar acciones en la base de datos de manera eficiente y ordenada.

**Framework**

Según [26] un *Framework* es un marco de trabajo que brinda una estructura base para la elaboración de un proyecto. En síntesis, es una plantilla que sirve como punto de partida para la organización y desarrollo de un proyecto de software. Generalmente los Frameworks son utilizados por los desarrolladores para acelerar el proceso de codificación de un sistema siguiendo una estructura determinada y fomentando la reutilización de código.

**API Rest**

En [27] se define como una Interfaz de Programación de Aplicaciones (API) que se ajusta a los límites de la arquitectura REST que es un conjunto de límites de arquitectura. API hace referencia al conjunto de protocolos y definiciones que se utilizan para integrar piezas de software en un sistema, suele ser considerada como el contrato entre el proveedor de la información y el usuario final de una aplicación. Una API se puede considerar REST cuando tiene una arquitectura cliente – servidor que se comunica mediante la utilización de peticiones HTTP.

**React**

En [21] se utilizó React como librería de desarrollo. React simplifica la construcción de interfaces de usuario dinámicas y reactivas al ofrecer una biblioteca de JavaScript eficiente y modular. Esta biblioteca se basa en componentes reutilizables que se pueden utilizar en diferentes partes del sistema. Su comunidad brinda soporte a los desarrolladores facilitando el entendimiento de la biblioteca. Adicionalmente es destacable que React es mantenido en la actualidad por Facebook lo cual hace que su documentación sea abundante.

**Next JS**

En [28] se define como una plataforma de desarrollo basada en React de código abierto destinada a la creación de software de alto rendimiento. Se caracteriza por su enfoque en el rendimiento y la facilidad de uso y funciones de optimización de carga mejorando la experiencia del usuario.

**Node JS**

En [10] se utilizó Node JS para la construcción del Backend del sistema. Node.js es un entorno de ejecución de JavaScript del lado del servidor. Permite a los desarrolladores utilizar JavaScript para construir aplicaciones de red escalables y de alto rendimiento. Node.js está diseñado para ser eficiente y orientado a eventos, lo que lo hace ideal para aplicaciones que manejan una gran cantidad de operaciones de entrada/salida (E/S) simultáneas y en tiempo real. Node JS se emplea para la construcción de un Backend basado en eventos.

**Nest JS**

Según su documentación oficial [29] es un marco para crear aplicaciones del lado del servidor escalables y eficientes. Para su funcionamiento utiliza Express, pero se puede utilizar también Fastify como marco de servidor HTTP. Proporciona un marco de abstracción adicional respecto a Express para la construcción de aplicaciones del lado del servidor.

**Object Relational Mapper (ORM)**

Según [30] un Object Relational Mapper por sus siglas en inglés (ORM) es una técnica de programación que permite trabajar con bases de datos relacionales de forma natural para un programador utilizando métodos predefinidos en lugar de sentencias SQL. Cuyo principal objetivo es simplificar el manejo de una base de datos desde el Backend de una aplicación.

**Prisma**

Según [31] es un ORM encargado de transformar los datos en objetos para su mejor manejo mediante la utilización de métodos definidos para gestionar una base de datos relacional. Utiliza un sistema de tipado asegurado, de esta manera evita los errores.

**PostgreSQL**

En [15] define a PostgreSQL como un SGBD de tipo relacional reconocido por su robustez, flexibilidad y capacidad para almacenar, organizar y recuperar datos de manera eficiente en diversos entornos de aplicaciones empresariales y web. La elección de PostgreSQL como SGBD se justifica por su capacidad para gestionar datos de forma eficiente y confiable, proporcionando un sólido respaldo para el almacenamiento y recuperación de información. PostgreSQL se ocupa como un SGBD para el almacenamiento de la información de sistemas web. Esto permite el manejo de la información de los aplicativos desarrollados de manera transparente para el usuario.

**Docker**

En [31] es una herramienta de despliegue de aplicaciones en contenedores diseñada para facilitar la creación y ejecución de servidores. Un contenedor es similar a una máquina virtual, pero evita la creación de un sistema operativo independiente. De igual manera Docker maneja de manera eficiente los recursos computacionales del host en el cual se ejecuta evitando sobre carga computacional en los componentes de este último.

**Arquitectura de tres capas**

Según [32] la arquitectura de tres capas es una arquitectura utilizada en el desarrollo de software bien definida que organiza los productos en tres capas bien definidas. La capa de presentación o la interfaz de usuario donde el cliente final interactúa con el sistema, la capa de aplicación que es donde se encuentra la lógica de negocio y procesan los datos y finalmente la capa de datos que es donde se almacenan todos los datos asociados con la aplicación.

**Crystal Clear**

Según [33] es una metodología de desarrollo ágil que se centra en la simplicidad y la transparencia. Se enfoca en cuatro ejes que son la comunicación constante, el enfoque en los aspectos esenciales del proyecto, entrega temprana y continua de software funcional y mejora continua del proceso. Esta metodología promueve los equipos pequeños dando claridad a los roles de cada miembro.

## Objetivos

### Objetivo general

Implementar un sistema mediante la utilización de una arquitectura web para el registro de asistencia docente en la Unidad Educativa Rumiñahui, ubicada en el cantón Ambato, provincia de Tungurahua, con un enfoque en la optimización del tiempo empleado en la generación de reportes de asistencia.

### Objetivos específicos

* Desarrollar un sistema de control docente mediante la utilización de una arquitectura web en la Unidad Educativa Rumiñahui en el cantón Ambato, provincia de Tungurahua.
* Evaluar la eficiencia en el proceso de generación de reportes de control de asistencia docente en la Unidad Educativa Rumiñahui en el cantón Ambato, provincia de Tungurahua.
* Implantar un sistema web de control docente que mejore la eficiencia en la generación de reportes de asistencia en la Unidad Educativa Rumiñahui en el cantón Ambato, provincia de Tungurahua.

# CAPÍTULO II. METODOLOGÍA



## Materiales

* Laptop
* Impresora
* IDE’s de programación
* Software Development Kit (SDK)
* Frameworks de desarrollo
* Software de Virtual Private Network (VPN)
* Sistema operativo
* Software de control de versiones
* Hojas de papel bond A4
* Esferos
* Cuaderno
* Tinta de impresora

## Métodos

### Modalidad de la investigación

La modalidad de la investigación es de tipo cuantitativo. Este enfoque permitirá definir valores numéricos precisos y medibles al proceso de control docente en la Unidad Educativa Rumiñahui. La presentación de datos numéricos brinda un enfoque objetivo de los resultados de la investigación, lo cual permite verificar la validez de esta.

### Población y muestra

La población utilizada para la presente propuesta de proyecto de titulación son los miembros del personal administrativo, docentes y personal de logística de la Unidad Educativa Rumiñahui ubicada en el cantón Ambato, provincia de Tungurahua en territorio ecuatoriano. Sumando el personal administrativo, docentes y personal de logística dan como resultado una población de 106 empleados dentro de la institución. Esta decisión se toma basado en una revisión de literatura la cual se ha detallado en secciones anteriores del presente documento. Para esto se tomará como muestra el 100% del personal de la Unidad Educativa Rumiñahui en vista que ellos serían los principales beneficiaros de este proyecto y se busca tener el margen de error más bajo posible.

### Recolección de información

Para la recolección de la información se utilizará una entrevista técnica de levantamiento de requerimientos. Adicional a esto se realizará una encuesta que estará conformada por preguntas en escala de Likert. Esta encuesta será validada mediante una prueba de consistencia interna con la finalidad de verificar la veracidad de la información recolectada.

### Procesamiento y análisis de datos

Las entrevistas mostradas en el Anexo A y Anexo B dirigidas al personal involucrado en el sistema de la Unidad Educativa Rumiñahui servirán para la definición clara y concisa de los requerimientos funcionales y operacionales del sistema los cuales se detallan en la formalización de los requerimientos del sistema bajo el estándar IEEE 830 que se muestra en el Anexo C. Estos requerimientos se documentarán debidamente con la finalidad de verificar el cumplimiento de estos cuando se entregue el proyecto a la institución.

En lo que respecta a las encuestas aplicadas al personal administrativo se realizará un análisis estadístico con la finalidad de medir la eficiencia del proceso de control docente antes y después de la aplicación del sistema. A esta técnica se la conoce como test y re-test. Con los resultados obtenidos de la encuesta se podrá evidenciar el aumento o disminución de la eficiencia en los procesos de control docente de la Unidad Educativa Rumiñahui.

En base a la recolección de información proveniente de las entrevistas detalladas en el Anexo A y el Anexo B se han obtenido los siguientes requerimientos funcionales del sistema descritos en la Tabla 1, los cuales se formalizaron en el Anexo C.

Tabla 1. Requerimientos funcionales del sistema

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Identificador** | **Requerimiento** | **Descripción** |
| RF01 | Inicio de sesión | El usuario puede utilizar los módulos del sistema de acuerdo con su rol dentro de la institución una vez iniciado sesión. |
| RF02 | Gestión de los datos del personal institucional | El administrador puede agregar, registrar el usuario y contraseña, listar, modificar y eliminar a un empleado dentro del sistema. Además, puede generar un reporte de todos los empleados de la institución con su información base y un reporte por cada tipo de empleado. |
| RF03 | Gestión de los datos de las sedes de la institución | El administrador puede agregar, listar, modificar y eliminar a una sede dentro del sistema. Además, puede generar reportes por cada sede y un reporte unificado. |
| RF04 | Registro de asistencia no presencial. | El personal de la institución puede registrar su asistencia dentro del sistema de manera no presencial cuando las disposiciones institucionales así lo requieran. |
| RF05 | Gestión de asistencia. | El administrador puede visualizar la asistencia del personal de la institución, visualizar las inconsistencias, sincronizar la información de manera manual, registrar la asistencia de un docente en específico de manera manual y generar reportes de asistencia tanto unificado siguiendo el formato establecido o individual por cada empleado. |
| RF06 | Gestionar hoja de vida del empleado. | El administrador puede visualizar, editar, agregar y eliminar información de un empleado incluida su hoja de vida con la información dada en la matriz Excel provista por la institución. Además, puede gestionar la información de cursos y certificaciones de los empleados. Finalmente podrá generar un reporte por empleado el cual contenga toda la información de la hoja de vida del empleado. |
| RF07 | Justificación de no registro de asistencia | El personal de la institución puede llenar un formulario para realizar la justificación de una inasistencia en su jornada de trabajo. |
| RF08 | Gestión de justificaciones de personal | El administrador puede listar las solicitudes de justificación del personal de la institución, cambiar de estado, agregar observaciones, eliminar una solicitud. Además, puede agregar documentos que sustenten la justificación como adjuntos, estos documentos se deben entregar de forma manual al administrador en primera instancia. |

De igual forma con la información recabada se han obtenido los siguientes requerimientos no funcionales del sistema:

Tabla 2. Requerimientos no funcionales

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Identificador** | **Requerimiento** | **Descripción** |
| RNF01 | Encriptación de la información | El sistema encriptará la información delicada dentro del sistema. |
| RNF02 | Sincronización de asistencia | El sistema se sincronizará de forma periódica cada 5 minutos para obtener los registros de asistencia presencial y no presencial. |
| RNF03 | Envió de recordatorios de registro de asistencia | El sistema enviará notificaciones al correo electrónico de los empleados que no registren su asistencia luego de pasar 5 minutos de la hora límite de registro. |
| RNF04 | Sincronización de solicitudes de justificación | El sistema sincronizará periódicamente cada día a una hora dada la información relacionada a las solicitudes de justificación de inasistencia del personal |
| RNF05 | Envío de notificaciones de cambio de estado de solicitud de justificación | Se envía un correo electrónico al empleado de la institución cuando una solicitud de justificación de inasistencia cambie de estado por acción del administrador |

En este sentido con la información recopilada se procedió a realizar la diagramación de los procesos que estarán inmersos en el sistema. En el Anexo D se puede evidenciar el proceso que se lleva a cabo para marcar la asistencia dentro de la institución. En el Anexo E se visualiza el proceso de generación de reportes de asistencia que se encuentra vigente actualmente en la institución. El Anexo F describe el proceso de registro de asistencia no presencial que se lleva a cabo dentro de la institución cuando se lo requiere. En el Anexo G se evidencia el proceso que se lleva a cabo cuando un docente desea justificar alguna inasistencia en la institución. Finalmente, en el Anexo H se detalla el proceso que se lleva a cabo para la generación de reportes de personal dentro de la institución.

# CAPÍTULO III. RESULTADOS Y DISCUSIÓN



## Descripción de procesos de la institución

La Unidad Educativa Rumiñahui no cuenta con diagramas de proceso que describan los procedimientos que se llevan a cabo para gestionar al personal docente y administrativo de la institución. Sin embargo, en base a las entrevistas realizadas al personal de la institución se ha elaborado los siguientes diagramas que pretenden describir de forma clara los protocolos que se llevan a cabo dentro de la institución.

Diagrama

Descripción generada automáticamente

Figura 1. Diagrama de proceso de registro de asistencia

Elaborado por el autor

Para la comprensión del diagrama el cual describe el proceso de registro de asistencia que actualmente se utiliza en la institución y presentado en la Figura 1 se ha realizado la siguiente tabla descriptiva del proceso llevado a cabo descrita en la Tabla 3.

Tabla 3. Descripción de proceso de registro de asistencia

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Registro de asistencia | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Registrar asistencia | Empleado | El empleado registra su asistencia utilizando sus datos biométricos. |
| Almacenar información | Reloj biométrico | El dispositivo biométrico almacena el registro de asistencia. |

El proceso descrito en la Tabla 3 es inalterable, es decir con la implementación del sistema no se modificará el procedimiento que llevan a cabo los empleados para registrar su asistencia de manera presencial.

Diagrama

Descripción generada automáticamente

Figura 2. Diagrama de proceso de marcaje de asistencia no presencial

Elaborado por el autor

De igual forma para una mejor comprensión del diagrama descrito en la Figura 2 el cual detalla el registro de asistencia no presencial en la institución se ha elaborado la siguiente tabla que describe el proceso a detalle.

Tabla 4. Descripción del proceso de registro de asistencia no presencial

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Registro de asistencia no presencial | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Crear formulario | Inspector general | El inspector general de la institución genera un formulario para registrar la asistencia del personal |
| Mostrar formulario | Google Forms | Se utiliza la plataforma Google Forms para poder compartir el formulario de asistencia generado |
| Compartir formulario | Inspector general | El inspector general comparte el enlace del formulario de asistencia no presencial al personal de la institución |
| Llenar formulario | Empleado | El personal de la institución llena el formulario con los datos solicitados |
| Almacenar datos | Google Forms | La plataforma Google Forms almacena las respuestas de los empleados |
| Elaborar informe | Inspector general | El inspector general elabora un informe de asistencia con los datos recopilados por la plataforma Google Forms proveniente del formulario creado |

En la Tabla 4 se describe a detalle el procedimiento que se lleva a cabo actualmente para el registro de asistencia no presencial dentro de la Unidad Educativa Rumiñahui. Este proceso en base a las entrevistas realizadas se debe optimizar para lo cual se ha estructurado el siguiente proceso de registro de asistencia no presencial el cual se describe en la Figura 3.

Diagrama

Descripción generada automáticamente

Figura 3. Diagrama de proceso de registro de asistencia no presencial propuesto

Elaborado por el autor

Para una mejor comprensión del diagrama ilustrado en la Figura 3 se ha elaborado la siguiente tabla que detalla los procedimientos descritos en este gráfico de procesos.

Tabla 5. Descripción del proceso de registro de asistencia no presencial propuesto

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Registro de asistencia no presencial propuesto | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Habilitar opción | Inspector general | El inspector general de la institución habilita la opción de registro de asistencia no presencial dentro del sistema |
| Notificar empleados | Inspector general | Se notifica al personal de la institución que se ha habilitado la función de asistencia no presencial dentro del sistema |
| Iniciar sesión | Empleado | Inicia sesión con sus credenciales asignadas |
| Autenticar usuario | SGD Rumiñahui | El sistema verifica que las credenciales sean correctas y autentica al usuario dentro del sistema |
| Mostrar opciones | SGD Rumiñahui | El sistema presenta las opciones disponibles al usuario |
| Seleccionar opción | Empleado | El empleado selecciona la opción correspondiente al registro de asistencia no presencial |
| Mostrar pantalla de registro | SGD Rumiñahui | El sistema presenta la pantalla donde el empleado registrara su asistencia |
| Registrar asistencia | Empleado | El empleado registra su ingreso o salida de la jornada laboral seleccionando la opción pertinente |
| Almacenar información | SGD Rumiñahui | El sistema almacena la información para posteriormente sincronizarla con los demás registros de asistencia del mes |

El proceso propuesto pretende reducir tiempos en la generación de reportes de asistencia, puesto que en el proceso actual de asistencia no presencial descrito en la Figura 4 y la Tabla 4 se dependía de la creación de un formulario de asistencia por parte del inspector general de la institución y de la generación de un reporte de asistencia manual para usarlo en el futuro en el conglomerado mensual de asistencia.

Diagrama

Descripción generada automáticamente

Figura 4. Diagrama de proceso de justificación de inasistencia

Elaborado por el autor

En la Figura 4 se describe el proceso mediante el cual el personal de la institución realiza la justificación de un no registro de asistencia a la institución. Para su mejor comprensión se ha elaborado la siguiente tabla que describe el proceso.

Tabla 6. Descripción del proceso de justificación de inasistencia

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Registro de asistencia no presencial propuesto | | |
| **Procedimiento** | **Actor** | **Descripción** |
| No registrar asistencia | Empleado | El empleado omite su registro de asistencia |
| Generar informe | Inspector general | El inspector general realiza el informe de asistencia mensual |
| Notificar empleado | Inspector general | El inspector general notifica al empleado de la falta de su registro de asistencia en un determinado día |
| Justificar inasistencia | Empleado | El empleado entrega la documentación respectiva anexada al formulario respectivo de justificación de inasistencia |
| Validar justificación | Inspector general | El inspector general valida que la información entregada por el empleado sea real |
| Aprobar justificación | Inspector general | El inspector general puede o no aprobar la justificación, en caso de que esta presente inconsistencias se regresa al proceso de notificar al empleado |
| Modificar informe | Inspector general | El inspector general procede a modificar el informe mensual de asistencia |
| Entregar informe | Inspector general | El inspector general entrega el informe mensual de asistencia con las correcciones correspondientes a la autoridad competente |

En el presente proceso se visualiza claramente el procedimiento a llevar a cabo para la justificación de una inasistencia en la institución. Esto se lleva a acabo siguiendo los lineamientos del Ministerio de Educación por lo cual es inalterable.

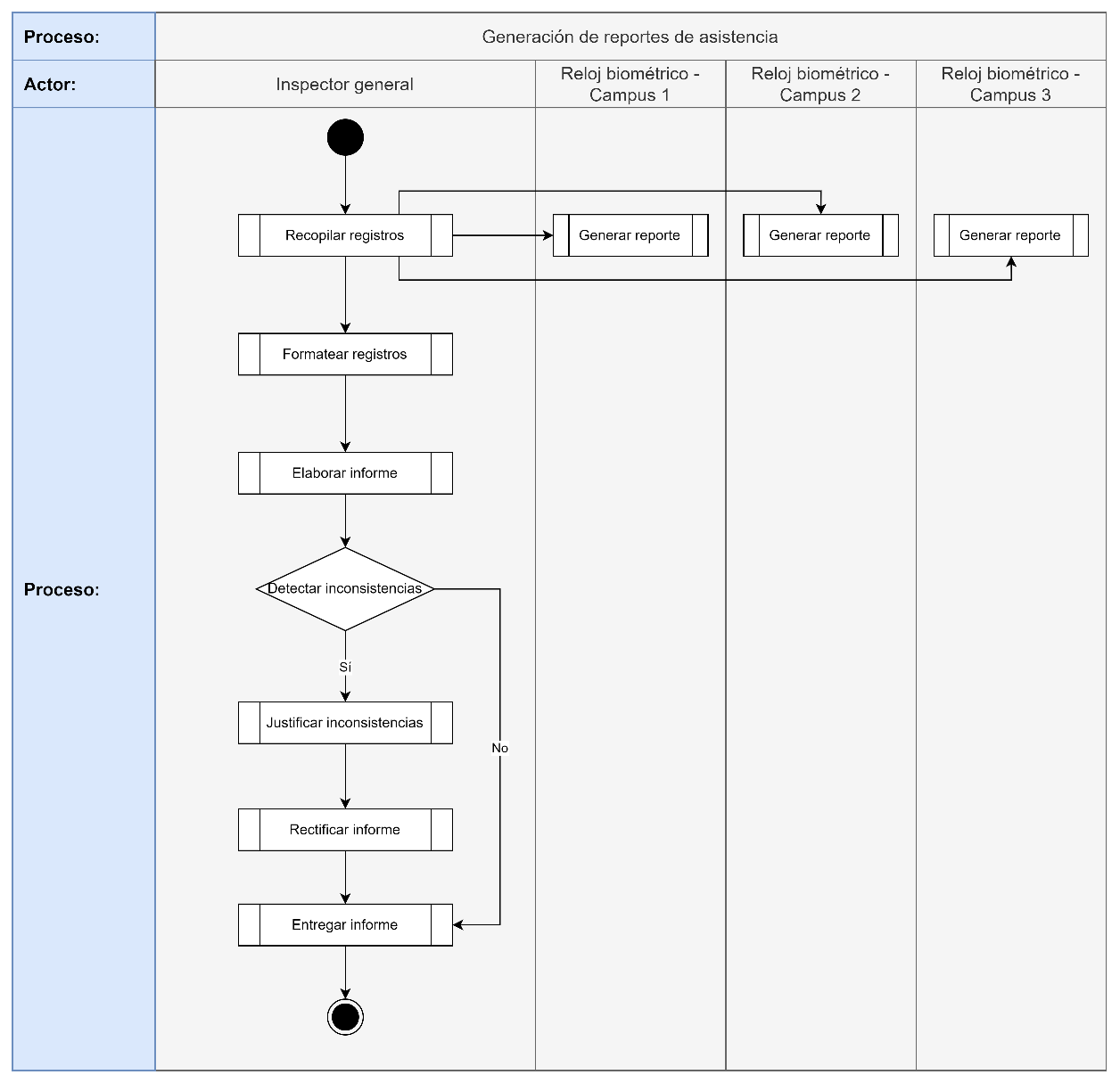


Figura 5. Diagrama de proceso de generación de reportes de asistencia

Elaborado por el autor

En la Figura 5 se evidencia el proceso que lleva a cabo el Inspector General de la institución para la elaboración del informe mensual de asistencia de personal remitido al Ministerio de Educación. El proceso se explicará a detalle en la tabla presentada a continuación.

Tabla 7. Descripción del proceso de generación de reportes de asistencia

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Generación de reportes de asistencia | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Recopilar registros | Inspector general | El inspector general de la institución recopila de forma manual y dirigiéndose a cada institución los registros de asistencia de cada uno de los relojes biométricos |
| Generar reporte | Reloj biométrico – Campus 1  Reloj biométrico – Campus 2  Reloj biométrico – Campus 3 | El dispositivo biométrico genera un archivo de texto plano con los registros de asistencia |
| Formatear registros | Inspector general | El inspector general da formato a los archivos de texto plano utilizando Microsoft Excel |
| Elaborar informe | Inspector general | El inspector general elabora el informe de asistencia mensual |
| Detectar inconsistencias | Inspector general | El inspector general verifica de forma manual que no existan incongruencias en los registros si no existieran se entrega el informe y si existieran se ingresa al proceso de justificación de inasistencia |
| Justificar inconsistencias | Inspector general | Ingresa al proceso de justificación de inasistencia descrito en la Figura 4 y la Tabla 6 del presente documento |
| Modificar informe | Inspector general | El inspector general procede a modificar el informe mensual de asistencia |
| Entregar informe | Inspector general | El inspector general entrega el informe mensual de asistencia con las correcciones correspondientes a la autoridad competente |

El proceso descrito en la Tabla 7 se lo considera ineficiente. Esto debido a que el inspector general de la institución debe dirigirse de manera física a cada campus para recopilar la información de los dispositivos biométricos lo que genera la pérdida de uno y en ocasiones de hasta dos días de trabajo. Este procedimiento se pretende mejorar mediante el proceso descrito en la Figura 7 descrita a continuación.

Diagrama

Descripción generada automáticamente

Figura 6. Diagrama de proceso de generación de reportes de asistencia propuesto

Elaborado por el autor

Este proceso pretende optimizar el proceso de generación de reportes de asistencia mediante la implementación del sistema propuesto. Este diagrama se lo explica de forma detallada en la Tabla 8 detallada a continuación.

Tabla 8. Descripción del proceso de generación de reportes de asistencia propuesto

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Generación de reportes de asistencia propuesto | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Mostrar asistencia | SGD Rumiñahui | El sistema muestra la información de asistencia en pantalla |
| Sincronizar registros | Inspector general | El inspector general selecciona la opción de sincronizar los registros para tener la información actualizada |
| Actualizar registros | SGD Rumiñahui | El sistema sincroniza los registros de asistencia con la información que proveen los dispositivos biométricos ubicados en los diferentes campus |
| Proveer registros | Reloj biométrico – Campus 1  Reloj biométrico – Campus 2  Reloj biométrico – Campus 3 | Los dispositivos biométricos proveen la información de registros de asistencia al sistema |
| Crear reporte | Inspector general | El inspector general selecciona la opción de generación de reportes de asistencia |
| Mostrar reporte | SGD Rumiñahui | El sistema muestra en pantalla el reporte generado |
| Guardar reporte | Inspector general | El inspector guarda el reporte generado por el sistema |
| Entregar informe | Inspector general | El inspector general entrega el informe mensual de asistencia con las correcciones correspondientes a la autoridad competente |

El proceso descrito en la Figura 6 y en la Tabla 8 pretende optimizar el tiempo que toma la generación de un reporte de asistencia mensual. En este procedimiento se elimina la necesidad de movilidad física que tiene el inspector general descrito en el diagrama definido en la Figura 5 y Tabla 7, lo cual permite un ahorro de tiempo significativo en la jornada laboral del inspector general de la institución.

Diagrama

Descripción generada automáticamente

Figura 7. Diagrama de proceso de generación de reportes de personal

Elaborado por el autor

Finalmente, en la Figura 7 se detalla el proceso que se lleva a cabo para la elaboración de informes de personal que labora en la institución educativa. El cual será descrito en la Tabla 9 detallada a continuación.

Tabla 9. Descripción del proceso de generación de reportes de personal

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Generación de reportes de personal | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Solicitar reporte | Autoridad | Una autoridad solicita un reporte de personal al inspector general de la institución |
| Recopilar información | Inspector general | El inspector general recopila la información del personal del cual se requiere el reporte |
| Elaborar informe | Inspector general | El inspector realiza un informe de manera manual utilizando un software de ofimática con la información requerida |
| Enviar informe | Inspector general | El inspector general remite el informe a la autoridad que lo requiere |

Con la finalidad de reducir la interacción manual del inspector general en este proceso se ha propuesto el siguiente procedimiento descrito en la Figura 8.

Diagrama

Descripción generada automáticamente

Figura 8. Diagrama de proceso de generación de reportes de personal propuesto

Elaborado por el autor

El presente proceso pretende reducir la interacción manual del inspector general en la generación de reportes de personal, dando la responsabilidad de la generación de reporte al sistema planteado. Para una mejor comprensión de este proceso se ha elaborado la Tabla 10 que se muestra a continuación.

Tabla 10. Descripción del proceso de generación de reportes de personal propuesto

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Proceso:** Generación de reportes de personal propuesto | | |
| **Procedimiento** | **Actor** | **Descripción** |
| Solicitar reporte | Autoridad | Una autoridad solicita un reporte de personal al inspector general de la institución |
| Crear reporte | Inspector general | El inspector general selecciona la opción dentro del sistema correspondiente a la generación de reportes de personal |
| Generar reporte | SGD Rumiñahui | El sistema genera el reporte bajo los criterios dados |
| Visualizar reporte | Inspector general | El inspector general visualiza el reporte |
| Enviar informe | Inspector general | El inspector general remite el informe a la autoridad que lo requiere |

## Estudio comparativo de metodologías de desarrollo ágiles

Para la elección de una metodología de desarrollo ágil que se adapte a las necesidades del presente proyecto se han analizado los siguientes marcos de trabajo: Scrum, Extreme Programming y Crystal. La información para la elaboración del siguiente cuadro comparativo descrito en la Tabla 3 se ha tomado de [34], [35], [36], [37] en la cual se describen las características de las metodologías seleccionadas.

Tabla 11. Cuadro comparativo de metodologías ágiles

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Aspecto** | **Scrum** | **Extreme Programming** | **Crystal Clear** |
| **Tamaño del equipo** | Recomendado para equipos medianos (hasta 10 personas). | Funciona bien con equipos pequeños a medianos. | Ideal para equipos pequeños (1-6 personas). |
| **Tamaño del proyecto** | Adecuado para proyectos de mediana escala. | Adecuado para proyectos de pequeña a mediana escala. | Adecuado para proyectos de pequeña escala. |
| **Flexibilidad** | Menos flexible, con sprints fijos y roles definidos. | Muy flexible, con iteraciones cortas y liberación continua. | Alta adaptabilidad a las necesidades del equipo y del proyecto. |
| **Ciclo de vida** | Basado en iteraciones de desarrollo. | Basado en iteraciones y liberaciones continuas. | Basado en iteraciones flexibles y adaptables a las necesidades del proyecto. |
| **Roles** | Roles bien definidos (Product Owner, Scrum Master, Equipo de Desarrollo). | Roles definidos, pero con énfasis en la colaboración y habilidades cruzadas. | Roles menos definidos, se enfoca en las personas y su interacción. |
| **Planificación** | Planificación fija para cada sprint. | Planificación basada en iteraciones con historias de usuario y liberaciones frecuentes. | Planificación continua y adaptable. |
| **Entregas** | Entregas al final de cada sprint. | Entregas muy frecuentes, a veces varias veces al día. | Entregas frecuentes y ajustes basados en retroalimentación. |
| **Cambios durante el proyecto** | No permite cambios durante el sprint. | Alienta cambios incluso tarde en el desarrollo. | Permite cambios en cualquier momento. |
| **Comunicación** | Comunicación estructurada en torno a reuniones de sprint. | Comunicación continua y extrema programación en parejas. | Comunicación constante y directa entre los miembros del equipo. |
| **Enfoque** | Se centra en la colaboración y la transparencia del proceso. | Se centra en la calidad técnica y el diseño del software. | Se centra en la seguridad personal y la mejora continua. |

**Metodología seleccionada:**

Basándose en el análisis comparativo presentado en la Tabla 3, que examina las metodologías ágiles más prominentes en el desarrollo de software, se ha tomado la decisión de seleccionar Crystal Clear como la metodología preferida para este proyecto. Esta elección se fundamenta en las características distintivas que ofrece esta metodología, las cuales se alinean de manera óptima con las necesidades específicas de la investigación en curso.

En primer lugar, destaca el tamaño recomendado del equipo, que oscila entre 1 y 6 personas, lo que se ajusta adecuadamente a la estructura y dimensiones del equipo actualmente disponible para el proyecto. Esta configuración favorece una comunicación más directa y una gestión más eficiente de recursos. Asimismo, la naturaleza flexible y adaptable de la gestión de cambios en Crystal Clear brinda la libertad necesaria para ajustar y adaptar el enfoque del proyecto conforme a las exigencias cambiantes del entorno, lo que fomenta la innovación y la mejora continua.

El enfoque centrado en la seguridad personal y la mejora continua no solo promueve un clima laboral propicio, sino que también garantiza un nivel óptimo de calidad en el producto final. La frecuencia de las entregas y los ajustes basados en la retroalimentación proporcionan una visibilidad temprana del progreso del proyecto y facilitan una rápida iteración y adaptación. En resumen, Crystal Clear emerge como la elección más adecuada para este estudio, ofreciendo una combinación única de flexibilidad, calidad y adaptabilidad que se alinean perfectamente con los objetivos y requisitos de la investigación en curso.

## Estudio comparativo de Frameworks de desarrollo Frontend

Para la elección de un Framework de desarrollo para el Frontend se han seleccionado los Frameworks Next JS, Vue JS y Angular JS ya que según [38] actualmente son los más utilizados en el mundo del desarrollo web. Adicionalmente la información para la elaboración del cuadro comparativo que se detalla en la Tabla 4 se ha tomado de [39], [40], [41], [42], [43], [44].

Tabla 12. Cuadro comparativo de Frameworks Frontend

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Aspecto** | **Next.js** | **Angular** | **Vue.js** |
| **Lenguaje** | JavaScript (basado en React) | TypeScript (basado en Angular) | JavaScript (basado en Vue) |
| **Popularidad** | En crecimiento | Ampliamente utilizado | En crecimiento |
| **Comunidad** | Activa y en constante expansión | Gran comunidad de desarrolladores | Comunidad activa y creciente |
| **Curva de aprendizaje** | Baja | Moderada | Baja |
| **Rendimiento** | Buen rendimiento en aplicaciones SSR (Server-Side Rendering) | Buen rendimiento en aplicaciones SPA (Single-Page Applications) | Buen rendimiento en aplicaciones SPA y SSR |
| **Enfoque** | Reactivo (componentes funcionales) | Reactivo (componentes basados en clases) | Reactivo (componentes funcionales) |
| **Integración con otros** | Fácil integración con otras bibliotecas y herramientas | Integración con herramientas de Google (por ejemplo, Firebase) | Integración sencilla con otras bibliotecas y herramientas |
| **Herramientas** | Amplia variedad de herramientas disponibles | Amplia gama de herramientas y módulos | Buena selección de herramientas |
| **Experiencia** | Alta | Baja | Nula |

**Framework seleccionado:**

Para la selección de un Framework de desarrollo Frontend se ha tomado como parámetros críticos para su selección la experiencia del desarrollador, la curva de aprendizaje, el soporte y la tendencia de crecimiento del Framework. En este contexto, Next JS se presenta como una selección idónea, considerando sus características como una curva de aprendizaje baja, una amplia comunidad y documentación y sobre todo la experiencia del desarrollador.

## Estudio comparativo de Frameworks de desarrollo Backend

Para la selección de un Framework de desarrollo para Backend se han seleccionado los Frameworks Express, ASP.NET Core y Nest JS los cuales se encuentran en el listado de Frameworks de desarrollo Backend más utilizados según [38]. De igual forma la información para la elaboración del cuadro comparativo detallado en la Tabla 5 se ha tomado de [45], [46], [47], [48], [49].

Tabla 13. Cuadro comparativo de Frameworks Backend

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Aspecto** | **Express** | **ASP.NET Core** | **Nest.js** |
| **Lenguaje** | JavaScript (basado en Node.js) | C# (basado en .NET Core) | TypeScript (basado en Node.js) |
| **Popularidad** | Ampliamente utilizado | Ampliamente utilizado | En rápido crecimiento |
| **Estructura del proyecto** | Menos estructura, mayor flexibilidad | Estructura modular y organizada | Estructura organizada similar a Angular |
| **Rendimiento** | Buen rendimiento en aplicaciones web | Buen rendimiento en aplicaciones web | Buen rendimiento en aplicaciones web |
| **Escalabilidad** | Escalable, pero la estructura depende del desarrollador | Escalable y estructura modular | Escalable y estructura modular |
| **Curva de aprendizaje** | Baja | Moderada | Baja |
| **Comunidad** | Gran comunidad de desarrolladores | Gran comunidad de desarrolladores | Comunidad activa y creciente |
| **Integración con otros** | Fácil integración con otras bibliotecas y herramientas | Integración con herramientas de Microsoft (por ejemplo, Azure) | Integración sencilla con otras bibliotecas y herramientas |
| **Documentación** | Buena documentación oficial | Buena documentación oficial | Documentación detallada y amigable |
| **Experiencia** | Media | Baja | Alta |

**Framework seleccionado:**

En este contexto y siendo vital para un desarrollo veloz en un proyecto de software la elección se basa sobre el parámetro de la experiencia, por lo cual se escoge Nest JS como Framework de desarrollo Backend. Este Framework trabaja bajo el lenguaje de Programación TypeScript el cual al ser un lenguaje tipado ayuda a evitar errores en momento de codificación. Adicionalmente es de recalcar que este Framework al trabajar bajo Node JS se puede hacer uso de todas las librerías que este entorno ofrece, simplificando varias de las funciones de un sistema.

## Estudio comparativo de gestores de bases de datos

En lo que respecta la selección de un sistema gestor de base de datos se debe considerar en primera instancia los tipos de bases de datos, los cuales son relacionales y no relacionales. En este sentido, para la elaboración del siguiente cuadro comparativo se ha tomado información de [50], [51], [52] la cual se representará de forma resumida en la Tabla 6.

Tabla 14. Cuadro comparativo de bases de datos SQL y No SQL

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Aspecto** | **SQL** | **No SQL** |
| **Modelo de datos** | Utiliza tablas para almacenar datos con relaciones definidas entre ellas | Utiliza documentos, columnas, grafos u otros modelos para almacenar datos, sin necesidad de relaciones predefinidas |
| **Estructura de datos** | Esquemas rígidos con estructuras de datos predefinidas y relaciones claras entre tablas | Esquemas flexibles que permiten almacenar datos no estructurados o semiestructurados |
| **Transaccionalidad** | Soporta transacciones ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad) | Algunas bases de datos NoSQL pueden soportar transacciones ACID, pero otras sacrifican consistencia para lograr mayor escalabilidad (teorema CAP) |
| **Consultas** | Lenguaje estructurado de consultas (SQL) para realizar consultas complejas | Las consultas pueden variar dependiendo del tipo de base de datos NoSQL, algunos tienen sus propios lenguajes de consulta mientras que otros pueden tener limitaciones en comparación con SQL |
| **Escalabilidad** | Escalabilidad vertical (escalado añadiendo más recursos a una máquina) limitada. Escalabilidad horizontal (escalamiento distribuido en múltiples servidores) puede ser complicada | Escalabilidad horizontal más sencilla debido a la naturaleza distribuida y flexible de muchas bases de datos NoSQL |
| **Enfoque** | Ideal para aplicaciones donde la estructura de los datos es clara y las transacciones son críticas | Ideal para aplicaciones con grandes volúmenes de datos, alta escalabilidad, y donde la flexibilidad y la velocidad de acceso a los datos son prioritarias |

Según la naturaleza del sistema y en función de la recolección de datos llevada a cabo en las etapas preliminares de este proyecto de desarrollo, se considera que una base de datos de tipo SQL es la opción idónea para la aplicación a ser construida. Esto se fundamenta en la naturaleza de los datos, los cuales requieren una estructuración que permita establecer relaciones detalladas entre dos o más entidades.

En este sentido, para la elección de un sistema gestor de bases de datos relacional se ha seleccionado como candidatos a PostgreSQL, MySQL y SQL Server siendo las opciones más populares de acuerdo con [38]. Adicionalmente se ha estructurado la siguiente tabla comparativa detallada en la Tabla 7 en la cual se compararán las principales características de cada candidato con información tomada de[53], [54], [55], [56], [57], [58].

Tabla 15. Cuadro comparativo de Sistemas de Gestión de Bases de Datos SQL

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Aspecto** | **PostgreSQL** | **MySQL** | **SQL Server** |
| **Lenguaje** | SQL estándar con algunas extensiones específicas | SQL estándar con algunas extensiones específicas | T-SQL (Transact-SQL), con funciones exclusivas de Microsoft |
| **Popularidad** | Ampliamente utilizado en aplicaciones empresariales | Muy popular en aplicaciones web y pequeñas empresas | Ampliamente utilizado en empresas y organizaciones |
| **Comunidad** | Comunidad activa y creciente | Gran comunidad de desarrolladores | Gran comunidad de usuarios y soporte de Microsoft |
| **Rendimiento** | Buen rendimiento en aplicaciones complejas | Buen rendimiento en aplicaciones de lectura/escritura | Buen rendimiento en aplicaciones empresariales |
| **Características** | Soporte para tipos de datos avanzados y funciones | Amplia variedad de motores de almacenamiento | Integración con herramientas y servicios de Microsoft |
| **Escalabilidad** | Escalabilidad horizontal y vertical | Escalabilidad horizontal y vertical | Escalabilidad horizontal y vertical |
| **Seguridad** | Fuerte encriptación, control de acceso y auditoría | Encriptación y control de acceso básico | Encriptación y funciones de seguridad avanzadas |
| **Lenguaje** | SQL estándar con algunas extensiones específicas | SQL estándar con algunas extensiones específicas | T-SQL (Transact-SQL), con funciones exclusivas de Microsoft |
| **Popularidad** | Ampliamente utilizado en aplicaciones empresariales | Muy popular en aplicaciones web y pequeñas empresas | Ampliamente utilizado en empresas y organizaciones |
| **Comunidad** | Comunidad activa y creciente | Gran comunidad de desarrolladores | Gran comunidad de usuarios y soporte de Microsoft |
| **Costo** | Gratis y de Código Libre | Gratis y de Código Libre | Opción gratuita con Código Propietario |

Con base a la Tabla 7 y considerando como criterio principal el costo del sistema se ha seleccionado PostgreSQL como sistema de gestión de base de datos para la construcción de este sistema. PostgreSQL dispone una amplia comunidad y es ampliamente utilizado en el mundo de desarrollo de software por su seguridad y rendimiento.

## Descripción de dispositivos biométricos

Los dispositivos biométricos que se utilizarán para el registro de asistencia del personal de la institución son los mismos que han venido utilizando en la Unidad Educativa en años anteriores. Estos son un dispositivo biométrico de asistencia de la marca Hikvision cuyo modelo es el AGREGAR EL MODELO y dos dispositivos biométricos de la marca ZKTeko cuyo modelo es el AGREGAR EL MODELO.

En este contexto en la Tabla 6 se detalla la información técnica del dispositivo Hikvision y en la Tabla 7 del dispositivo ZKTeko.

AGREGAR TABLAS

## Diseño de arquitectura de la aplicación

Para la implementación del presente proyecto de desarrollo se ha elaborado la siguiente arquitectura del sistema, la cual se encuentra basada en las necesidades de la institución y las restricciones a nivel de costos de esta última.

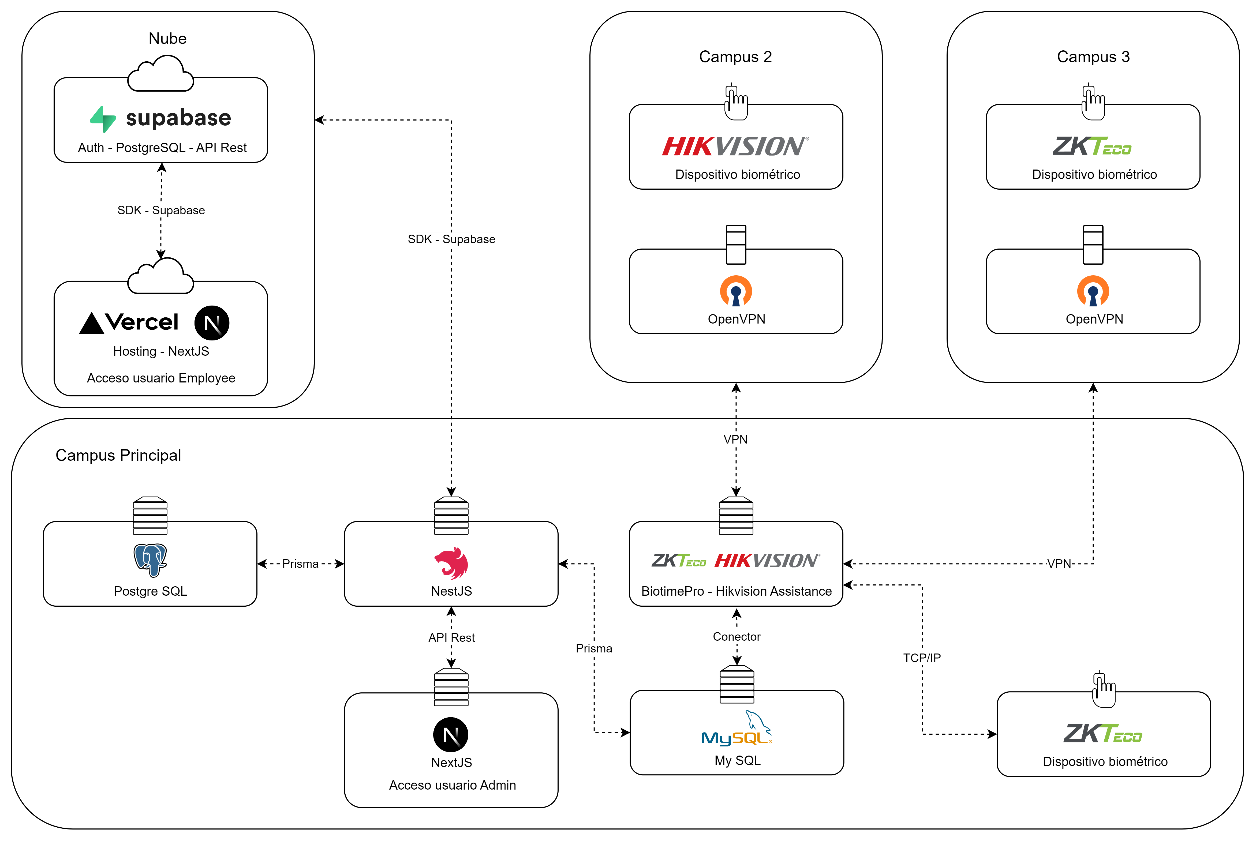


Figura 9. Arquitectura del sistema

Elaborado por el autor

En la Figura 9 se puede evidenciar la arquitectura bajo la cual se implementará la solución dentro de la institución. Se tiene una división de los campus de la institución y adicionalmente la descripción de la arquitectura en la nube bajo la cual funcionará el módulo al cual van a tener acceso todos los empleados de la institución para realizar su registro de asistencia no presencial descrito en el RF04 de la Tabla 1 y el RF07 descrito en la Tabla 1 que consisten en la justificación de no registros de asistencia en un determinado día.

Se tiene una división en lo que respecta las sedes de la institución en vista que cada sede dispone de un reloj biométrico diferente para el registro de la asistencia del personal. El servidor principal del sistema se encontrará en el campus principal de la institución el cual aloja tanto el Frontend, el Backend y las Bases de Datos del sistema. En cada una de las sedes se coloca una PC que servirá para enrutar el tráfico de los dispositivos biométricos hacia el servidor ubicado en el campus principal mediante la utilización de Open VPN.

Es de recalcar que para la recolección de la información de los biométricos no se utilizará el SDK de desarrollo de estos por motivo que en el caso de los dispositivos ZKTeko este es de pago y excede el límite presupuestario dado por la institución, por lo cual se ha visto en la necesidad de utilizar el Software propietario de ZKTeko para obtener los registros de asistencia de los dispositivos biométricos. Este Software provee una opción de configuración la cual permite utilizar una base de datos propia para el almacenamiento de los registros de asistencia, como se evidencia en la Figura 6 se utilizará MySQL para esto.

De igual forma en el caso del dispositivo Hikvision se utilizará el Software propietario de Hikvision para el manejo de los registros de asistencia del personal, el mismo que se encontrará vinculado a la Base de Datos MySQL que utilizará el Software de ZKTeko.

## Ciclo de vida de la metodología Crystal Clear

Para una mejor comprensión del ciclo de vida de la metodología de desarrollo ágil Crystal Clear se ha elaborado el siguiente esquema denotado en la Figura 10 en base a la información presentada en [33], [59], [60], [61], [62].

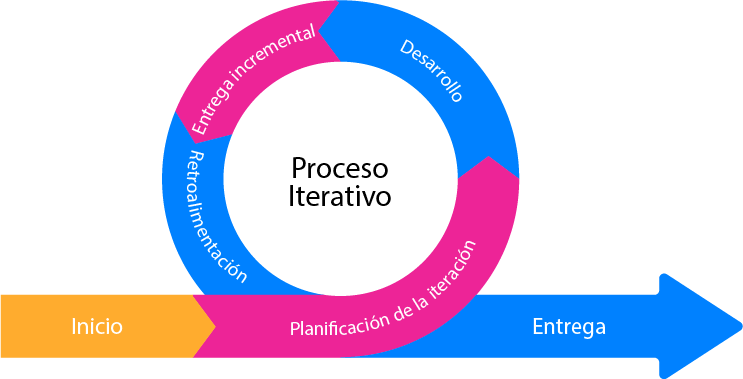


Figura 10. Esquema de la metodología Crystal Clear

Elaborado por el autor

### Inicio

En la etapa inicial de esta metodología se ejecutan acciones que definen el alcance del proyecto y su alcance. Aquí se define el equipo del trabajo designando los roles de cada miembro del grupo, se define la línea de comunicación, se identifican las características claves del sistema definiendo su alcance y finalmente se hace una primera planificación estableciendo las estimaciones para cada función del sistema.

**Establecimiento del equipo de trabajo**

Según [33], [59], [60], [61], [62] como primer punto a seguir en esta metodología es vital establecer los roles que va a tener cada uno de los integrantes del equipo de trabajo involucrado en el proyecto. En este sentido, se han definido los siguientes roles: tutor académico, responsable por parte de la institución y el desarrollador del sistema. Estos roles se encuentran detallados en la Tabla 16 que se encuentra a continuación.

Tabla 16. Roles del equipo de trabajo

Elaborado por el autor

|  |  |
| --- | --- |
| **Rol** | **Integrante** |
| Tutor académico | Ing. Hernán Naranjo Mg. |
| Responsable por parte de la institución | Lcdo. Cristian Paredes Mg. |
| Desarrollador | Andrés Tapia |

**Establecimiento de una línea de comunicación**

En base a [33], [59], [60], [61], [62] para implementar efectivamente la metodología Crystal Clear, es esencial establecer una línea de contacto que fomente la confianza y el compañerismo entre los miembros del equipo. Se ha elegido WhatsApp como el principal medio de interacción para todo el equipo, especialmente entre el responsable de la institución y el desarrollador. Asimismo, para la conexión entre el tutor académico y el desarrollador, se ha asignado el correo electrónico institucional como la vía principal, reservando WhatsApp como un recurso secundario. Esta disposición facilita una interacción constante entre los integrantes del equipo, lo cual se espera que contribuya al éxito del proyecto de desarrollo.

**Identificación de las características claves**

Para la definición de las características clave del sistema se ejecutaron dos entrevistas al Inspector General de la Unidad Educativa Rumiñahui, siendo este el responsable por parte de la institución. Estas entrevistas se realizaron con una separación de 15 días con la finalidad de que los requerimientos funcionales y no funcionales del sistema se definan de forma clara. Con esta información recabada se elaboró un documento técnico de levantamiento de requerimientos siguiendo el estándar IEEE 830 de 1998 el cual se encuentra en el Anexo N. Este documento pretende realizar una formalización de los requisitos del sistema.

Si bien es cierto en la metodología Crystal Clear no se realiza un levantamiento exhaustivo de requerimientos en etapas tempranas de desarrollo, se consideró que el documento de requerimientos permitirá tanto a la institución como al desarrollador de este sistema comprender el alcance del proyecto de manera global. El uso de este estándar pretende disminuir en cierta forma los cambios que se puedan implementar en el proyecto, los cuales pueden incurrir en gastos adicionales.

**Planificación inicial y estimación de tiempos**

Según [59] una estimación de tiempos inicial ayuda al proceso de esta metodología. Para ello se pueden utilizar mecanismos de estimación provenientes de otras metodologías de desarrollo ágiles. Para esta etapa el autor utilizó una estimación por puntos de historia, la cual según [63] se utiliza para estimar los tiempos de ejecución de un proyecto de desarrollo. Esta técnica consiste en definir puntos de historia a cada requerimiento del sistema y en base a esto realizar un cálculo del tiempo estimado. Este mecanismo toma un requerimiento en el cual el equipo de desarrollo tenga experiencia y lo toma como base para estimar el resto de las funcionalidades del sistema. Finalmente se estructuran las tareas a llevarse a cabo en base a esta planificación y estimación inicial.

### Proceso iterativo

En esta etapa de la metodología se realizan las iteraciones consideradas necesarias para la consecución de los objetivos planteados en la etapa inicial. Según CITA en cada una de las iteraciones se realiza una planificación, una entrega de una parte funcional del sistema y en base a esta se realiza una retroalimentación al equipo de desarrollo por parte de los interesados del sistema.

**Planificación de la iteración**

En este sentido y para facilitar la estructuración de la iteración se ha elaborado la siguiente matriz descrita en la Tabla 17 que pretende facilitar el seguimiento de las tareas de la iteración.

Tabla 17. Matriz de tareas

Elaborado por el autor

|  |  |
| --- | --- |
| **TAREA** | |
| **Tarea:** | *Identificador de la tarea* |
| **Descripción:** | *Descripción global de la tarea a realizar* |
| **Estimación:** | *Estimación del tiempo de desarrollo en horas* |
| **Responsable:** | *Miembro responsable de la tarea* |
| **Criterios de aceptación:** | *Objetivos de la tarea* |

**Realización de pruebas y retroalimentación continua**

Para finalizar la iteración [33], [59], [60], [61], [62] detalla que esta fase constituye un componente crucial en cada iteración. Durante esta etapa se pretende probar la funcionalidad del componente de software funcional. Estas pruebas pueden ser tanto manuales como automáticas. Para facilitar el registro de las pruebas y su respectiva retroalimentación se ha elaborado la siguiente matriz descrita en la Tabla 18.

Tabla 18. Matriz de prueba

Elaborado por el autor

|  |  |
| --- | --- |
| **PRUEBA** | |
| **Identificador:** | *Identificador numérico de la prueba.* |
| **Tarea relacionada:** | *Identificador de la tarea donde se encuentra la descripción del componente funcional de software.* |
| **Criterio de aceptación:** | *Criterio bajo el cual se aprobará o rechazará la prueba realizada.* |
| **Retroalimentación:** | *Retroalimentación por parte de las partes.* |
| **Estado:** | *Estado de aceptación de la prueba, el cual puede ser Aprobado o Rechazado.* |

De acuerdo con el resultado de la prueba se corrige la funcionalidad de la iteración o se procede con la entrega de la pieza de Software del sistema. De esta forma se finaliza la iteración.

### Entrega

Según [33], [59], [60], [61], [62] en esta etapa final del proceso de desarrollo se realiza la entrega del sistema final y se inicia una etapa de mantenimiento y correcciones. Esta etapa da por finalizado el proceso de desarrollo del sistema.

## Desarrollo

En esta sección del documento se describirá cada una de las iteraciones realizadas en la etapa de desarrollo del sistema, en base a la descripción de la metodología planteada. Como punto inicial se definirá una estimación del tiempo necesario para la finalización del proyecto descrita en la etapa de *Planificación inicial y estimación de tiempos* en secciones anteriores del presente documento.

### Planificación inicial

En este contexto como primer punto se establece el tiempo semanal disponible por el desarrollador para la ejecución del presente proyecto el cual se define en la Tabla 11 descrita a continuación.

Tabla 19. Tiempo disponible del desarrollador

Elaborado por el autor

|  |  |  |
| --- | --- | --- |
| **Días a la semana** | **Total, de horas diarias** | **Total, de horas semanales** |
| 5 días | 4 horas | 20 horas |

En base al tiempo disponible se ha realizado la siguiente estimación de tiempos en base a los requerimientos funcionales y no funcionales del sistema descrita en la Tabla 20 detallada a continuación.

Tabla 20. Estimaciones de las tareas

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidad** | **Tareas** | **Tiempo estimado** | **Prioridad** |
| **Horas** |
| Inicio de sesión | Generar diseño de la pantalla de inicio de sesión | 3 | Alta |
| Diseño del modelo entidad relación del usuario | 1 | Media |
| Implementación de la lógica para el manejo de la autenticación del usuario | 3 | Alta |
| Creación de métodos HTTP para autenticar al usuario | 2 | Alta |
| Codificación de la interfaz del módulo de inicio de sesión | 4 | Alta |
| Pruebas de inicio de sesión | 1 | Baja |
| Interfaz de usuario | Generar barra lateral para la navegación | 4 | Alta |
| Generar el esquema para las páginas | 1 | Alta |
| Pruebas de la interfaz de usuario | 1 | Baja |
| Registro de asistencia no presencial | Generar diseño de pantalla para el registro de asistencia no presencial | 2 | Alta |
| Generar el modelo entidad relación del módulo de asistencia no presencial | 1 | Media |
| Creación de métodos HTTP para el marcaje no presencial de un empleado de la institución | 1 | Alta |
| Codificación de la interfaz del módulo de registro de asistencia no presencial | 4 | Alta |
| Despliegue del módulo de registro de asistencia no presencial | 1 | Media |
| Pruebas del módulo de registro de asistencia no presencial | 1 | Baja |
| Gestión de datos de las sedes de la institución | Generar diseño de la pantalla para la visualización de las sedes de la institución | 2 | Alta |
| Generar diseño pantalla con el formulario para la inserción y edición de las sedes institucionales | 2 | Alta |
| Generar el modelo entidad relación de una sede institucional | 1 | Media |
| Creación de métodos HTTP para el manejo de las sedes institucionales | 2 | Alta |
| Codificación de la pantalla de visualización de las sedes de la institución | 2 | Alta |
| Codificación de la interfaz del formulario de gestión de sede institucional | 5 | Alta |
| Creación de métodos HTTP para la generación de reportes de las sedes institucionales | 4 | Baja |
| Pruebas del módulo de gestión de datos de las sedes institucionales | 1 | Baja |
| Gestión de datos del personal de la institución | Generar diseño de la pantalla para la visualización del personal de la institución | 2 | Alta |
| Generar diseño de la pantalla para la visualización de un empleado de la institución | 2 | Alta |
| Generar el diseño de la pantalla de formulario para la inserción y edición de datos generales del empleado | 2 | Alta |
| Generar diseño de la pantalla para la gestión los títulos de un empleado | 2 | Alta |
| Generar diseño de pantalla del formulario para la gestión de los títulos de un empleado | 2 | Alta |
| Generar diseño de pantalla para la gestión de los certificados de un empleado | 2 | Alta |
| Generar diseño de pantalla del formulario para la gestión de los certificados de un empleado | 2 | Alta |
| Generar diseño de pantalla para la gestión de los horarios de un empleado | 2 | Alta |
| Generar diseño de pantalla del formulario para la gestión de los horarios de un empleado | 2 | Alta |
| Generar el modelo entidad relación para el módulo de gestión de personal | 1 | Media |
| Creación de métodos HTTP para la gestión del personal de la institución | 6 | Alta |
| Creación de métodos HTTP para la generación de reportes de personal | 9 | Alta |
| Codificación de la pantallade visualización de personal de la institución | 2 | Alta |
| Codificación de la pantalla para la visualización de un empleado de la institución | 1 | Alta |
| Codificación del formulario de inserción y edición de datos generales del empleado | 2 | Alta |
| Codificación de la pantalla de gestión de certificados de un empleado | 1 | Alta |
| Codificación del formulario de inserción y edición de certificados de un empleado | 2 | Alta |
| Codificación de la pantalla de gestión de títulos de un empleado | 1 | Alta |
| Codificación del formulario de inserción y edición de títulos de un empleado | 2 | Alta |
| Codificación de la pantalla de gestión de horarios de un empleado | 1 | Alta |
| Codificación del formulario de inserción y edición de los horarios de un empleado | 2 | Alta |
| Pruebas del módulo de gestión de personal | 3 | Media |
| Gestión de asistencia de personal | Generar diseño de la pantalla para la visualización de la asistencia del personal de la institución | 2 | Alta |
| Generar diseño del formulario para la inserción manual de un registro de asistencia | 2 | Alta |
| Generar el modelo entidad relación del módulo de asistencia | 1 | Media |
| Creación de métodos HTTP para la gestión de la asistencia del personal | 3 | Alta |
| Creación de métodos HTTP para la generación de reportes de asistencia de personal | 4 | Alta |
| Codificación de la pantalla de visualización de la asistencia del personal de la institución | 2 | Alta |
| Codificación del formulario de inserción manual de un registro de asistencia | 2 | Media |
| Pruebas del módulo de gestión de asistencia | 1 | Media |
| Justificaciones de inasistencia | Generar diseño de la pantalla de visualización de justificaciones enviadas | 2 | Alta |
| Generar diseño de la pantalla de los formularios para la justificación de inasistencias de personal | 2 | Alta |
| Generar modelo entidad relación del módulo de justificaciones de personal | 1 | Media |
| Creación de métodos HTTP para generar y listar las justificaciones de inasistencia | 2 | Alta |
| Creación de métodos HTTP para la generación del documento físico de justificación de inasistencia | 2 | Alta |
| Codificación de la pantalla de visualización de justificaciones enviadas | 2 | Alta |
| Codificación de los formularios para enviar una justificación de inasistencia | 2 | Alta |
| Pruebas del módulo de justificación de inasistencias | 1 | Media |
| Gestión de justificaciones recibidas | Generar diseño de la pantalla de visualización de justificaciones enviadas por el personal de la institución | 2 | Alta |
| Generar el diseño de visualización específica de una justificación recibida | 2 | Alta |
| Generar el diseño de la pantalla para la carga de documentos que sustenten la justificación | 2 | Alta |
| Generar el modelo entidad relación del módulo de gestión de justificaciones | 1 | Media |
| Creación de métodos HTTP para la gestión de las justificaciones receptadas | 2 | Alta |
| Creación de métodos HTTP para la generación de reportes de justificaciones receptadas | 4 | Alta |
| Codifcació de la pantalla de visualización de justificaciones recibidas | 2 | Alta |
| Codificación de la pantalla de visualización de una justificación en específico | 2 | Alta |
| Codificación de la pantalla de gestión de documentación que sustenta la justificación | 7 | Alta |
| Pruebas del módulo de gestión de justificaciones | 2 | Media |

Bajo estas estimaciones se procede a realizar la planificación de cada una de las iteraciones, con base al tiempo disponible por el desarrollador detallado en la Tabla 19 y las estimaciones descritas en la Tabla 20.

En este sentido, para la ejecución de cada iteración se ha determinado un tiempo de duración de dos semanas iniciando el lunes y terminando el día domingo de la siguiente semana dando como resultado una disponibilidad de tiempo de 40 horas para cada uno de los procesos cíclicos de desarrollo. En base a esto se ha procedido a realizar las planificaciones de cada una de las iteraciones.

### Primera iteración

**Planificación**

En base a la planificación general se han seleccionado las siguientes funcionalidades del sistema para su desarrollo en la iteración número 1: Inicio de sesión, Interfaz de usuario, Registro de asistencia no presencial. Las tareas designadas para completar estas funcionalidades, así como el tiempo se encuentran descritas en la Tabla 21 y 22 respectivamente, las cuales se encuentran a continuación.

Tabla 21. Detalle de la macro tarea Inicio de sesión

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidad:** | Inicio de sesión | | |
| **Tarea** | **Tiempo** | **Prioridad** | **Criterios de aceptación** |
| Generar diseño de la pantalla de inicio de sesión | 3 | Alta | **Como** usuario **quiero** iniciar sesión utilizando un correo electrónico y una contraseña |
| Diseño del modelo entidad relación del usuario | 1 | Media | **Como** usuario **quiero** poder iniciar sesión de forma segura dentro de la aplicación |
| Implementación de la lógica para el manejo de la autenticación del usuario | 3 | Alta |  |
| Creación de métodos HTTP para autenticar al usuario | 2 | Alta |  |
| Codificación de la interfaz del módulo de inicio de sesión | 4 | Alta |  |
| Pruebas de inicio de sesión | 1 | Baja |  |
| **Tiempo total:** | 14 |  |  |

Tabla 22. Detalle de la macro tarea Interfaz de usuario

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidad:** | Interfaz de usuario | | |
| **Tarea** | **Tiempo** | **Prioridad** | **Criterios de aceptación** |
| Generar barra lateral para la navegación | 4 | Alta | **Como** usuario **quiero** poder ver y acceder a todas las secciones principales de la aplicación desde la barra lateral. |
| Generar el esquema para las páginas | 1 | Alta | **CRITERIO DE ACEPTACION** VA ENFOCADO EN FUNCION DE LO QUE QUIERO PROBAR  DADO QUE ME ENCUENTRO EN EL FORMULARIO DE LOGIN CUANDO INGRESO ENTONCES  DADO, CUANDO, ENTONCES |
| Pruebas de la interfaz de usuario | 1 | Baja |  |
| **Tiempo total:** | 6 |  |  |

Tabla 23. Detalle de la macro tarea Registro de asistencia no presencial

Elaborado por el autor

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidad:** | Registro de asistencia no presencial | | |
| **Tarea** | **Tiempo** | **Prioridad** | **Criterios de aceptación** |
| Generar diseño de pantalla para el registro de asistencia no presencial | 2 | Alta | **Como** empleado **quiero** poder registrar mi asistencia de manera no presencial  **Como** empleado **quiero** que la interfaz de usuario para registrar la asistencia no presencial sea sencilla  **Como** empleado **quiero** poder registrar mi asistencia de manera no presencial utilizando mi celular |
| Generar el modelo entidad relación del módulo de asistencia no presencial | 1 | Media |
| Creación de métodos HTTP para el marcaje no presencial de un empleado de la institución | 1 | Alta |
| Codificación de la interfaz del módulo de registro de asistencia no presencial | 4 | Alta |
| Despliegue del módulo de registro de asistencia no presencial | 1 | Media |  |
| Pruebas del módulo de registro de asistencia no presencial | 1 | Baja |
| **Tiempo total:** | 10 |  |  |

La presente iteración tiene un tiempo de 30 horas en total para su ejecución, con una holgura de 10 horas para que se pueda solventar cualquier imprevisto dentro del desarrollo de las funcionalidades asignadas.

**Desarrollo**

**PENDIENTE DE AGREGAR MÁS HISTORIAS DE USUARIO**

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Figura 11. Diseño de la pantalla de inicio de sesión

Elaborado por el autor

AGREGAR DESCRIPCION A CADA FIGURA Y TABLA SIEMPRE ARRIBA

DESGLOSAR CADA MACROTAREA EN SUBTAREAS CON SUN RESPECTIVA ESTIMACION. EN TODO SE INICIA CON EL MOCKUP

Tabla 24. Matriz de la tarea 02

Elaborado por el autor

|  |  |
| --- | --- |
| **TAREA** | |
| **Tarea:** | 02 |
| **Descripción:** | Desarrollar la pantalla de “*Inicio*” del módulo al que tienen acceso los empleados |
| **Estimación:** | 0.5 horas |
| **Responsable:** | Desarrollador |
| **Criterios de aceptación:** | **Como** empleado **quiero** poder visualizar las funcionalidades a las cuales tengo acceso |

Captura de pantalla de un celular con letras

Descripción generada automáticamente

Figura 12. Diseño de la pantalla de inicio del módulo de empleados

Elaborado por el autor

**Fase de pruebas**

Para la fase de pruebas y retroalimentación de la iteración se ha tomado como modelo a seguir la matriz detallada en la Tabla 8.

Tabla 25. Matriz de prueba 01

Elaborado por el autor

|  |  |
| --- | --- |
| **PRUEBA** | |
| **Identificador:** | 01 |
| **Tarea relacionada:** | 01 |
| **Criterio de aceptación:** | **Como** usuario **quiero** poder iniciar sesión de forma segura dentro de la aplicación |
| **Retroalimentación:** | **Tutor académico:**  Ninguna  **Responsable de la institución:**  Ninguna  **Desarrollador:**  Ninguna |
| **Estado:** | Aprobado |

Tabla 26. Matriz de prueba 02

Elaborado por el autor

|  |  |
| --- | --- |
| **PRUEBA** | |
| **Identificador:** | 02 |
| **Tarea relacionada:** | 01 |
| **Criterio de aceptación:** | **Como** usuario **quiero** iniciar sesión utilizando un correo electrónico y una contraseña |
| **Retroalimentación:** | **Tutor académico:**  Ninguna  **Responsable de la institución:**  Ninguna  **Desarrollador:**  Ninguna |
| **Estado:** | Aprobado |

Tabla 27. Matriz de prueba 03

Elaborado por el autor

|  |  |
| --- | --- |
| **PRUEBA** | |
| **Identificador:** | 03 |
| **Tarea relacionada:** | 02 |
| **Criterio de aceptación:** | **Como** empleado **quiero** poder visualizar las funcionalidades a las cuales tengo acceso |
| **Retroalimentación:** | **Tutor académico:**  Ninguna  **Responsable de la institución:**  Ninguna  **Desarrollador:**  Ninguna |
| **Estado:** | Aprobado |

# CAPÍTULO IV. CONCLUSIONES Y RECOMENDACIONES



## Conclusiones

* Texto
* Texto

## Recomendaciones

* Texto
* Texto

# REFERENCIAS BIBLIOGRÁFICAS

[1] Ministerio de Telecomunicaciones y Sociedad de la Información MINTEL, *Libro Blanco de la Sociedad de la Información y del Conocimiento*. Quito: MINTEL, 2018.

[2] R. Chillcce, “Sistema de Información Biométrico para la Gestión del Control de Asistencia del Personal Administrativo y Docente de la UDEA, 2020,” Universidad Para el Desarrollo Andino, Huancavelica, 2021. [Online]. Available: https://orcid.org/0000-

[3] Ministerio de Educación, “Datos abiertos del Ministerio de Educación del Ecuador.”

[4] J. Albarrán and B. Marquez, *Operaciones administrativas de recursos humanos*. Madrid: Editorial RA-MA, 2015.

[5] K. Sanchez, “Desarrollo de un Sistema Web y Control de Asistencia del Personal en la Institución Educativa Privada Triolet - Huaura, 2019,” Universidad Nacional José Faustino Sánchez Carrión, Huacho, 2019.

[6] M. Arias, “Desarrollo de una aplicación web para la mejora del control de asistencia de personal en la Escuela Tecnológica Superior de la Universidad Nacional de Piura,” Universidad Inca Garcilaso de la Vega, Lima, 2018.

[7] T. De La Rosa and G. Garcia Rubio, “Sistema para la Gestión y Control de marcaciones del personal de la Universidad Metropolitana del Ecuador,” *Sistema para la Gestión y Control de Marcaciones del Personal de la Universidad Metropolitana del Ecuador*, 2019, [Online]. Available: http://remca.umet.edu.ec/index.php/REMCA

[8] S. Alvarado, “Implementación de un sistema web para reducir el tiempo en la gestión y control de asistencia, permisos y licencias de vacaciones en la Escuela Superior Militar de Aviación Cosme Rennella Barbato, Salinas.,” Universidad Estatal Península de Santa Elena, La Libertad, 2019.

[9] C. Vivanco and A. Díaz, “Sistema web con identificación a través de huella digital para la mejora de la eficacia del registro y control de asistencia en la empresa Induamerica Servicios Logísticos SAC,” Universidad Nacional Pedro Ruiz Gallo, Lambayeque, 2019.

[10] L. Del Alcázar, “Desarrollo de un sistema integrado de gestión de horarios y control de asistencia docente para el centro Preuniversitario de la UNAMAD,” Universidad Nacional Amazónica de Madre de Dios, Puerto Maldonado, 2019.

[11] E. Barrios and G. Villadiego, “Sistema para el registro y control de personal con tecnología biométrica para el Almacén Mini Precios de Sahagún - Córdoba,” Universidad de Córdova, Córdoba, 2020.

[12] J. Escandón and R. Coronel, “Diseño e Implementación de un Prototipo de Marcación con Control Dactilar Utilizando Hardware de Bajo Costo y Visualizado a Través de Una Aplicación Web,” Guayaquil, 2020.

[13] K. Micha and J. Medina, “Sistema Bioasis y el proceso de control de asistencia en las obras de la empresa Cna Construcciones S.A.C. – Lima,” Universidad Cesar Vallejo, Callao, 2020.

[14] J. Sanchez, “Sistema de Control y Administración de Recursos Humanos para la Incubadora de Microempresas Productivas,” Universidad Pública De El Alto, El Alto, 2020.

[15] P. Sullo, “Propuesta de implementación del sistema biométrico para el control de asistencia administrativa de la Gerencia Regional de Educación Moquegua-2019,” Universidad Autónoma del Perú, Lima, 2021.

[16] A. Romero, “Implementación de un sistema Web con reloj biométrico para la empresa HGV contratistas generales SAC - Chimbote; 2021,” Universidad Católica Los Ángles Chimbote, Chimbote, 2021.

[17] T. Sulla, “Sistema biométrico basado en aplicaciones móviles para el control de asistencia de estudiantes del Instituto Superior Tecnológico Americana del Cusco,” Universidad Cesar Vallejo, Lima, 2022.

[18] Y. Ardiles, “Implementación de un sistema web para mejorar el control de asistencia en la direccion regional de educación de Ancash, 2022,” Universidad Nacional Santiago Atúnez de Mayolo, Huaraz, 2022.

[19] L. Paucar, “Implementación de un Sistema de Registro para el Personal Administrativo y Docente del Instituto Superior Tecnológico San Gabriel Utilizando un Lector de Huellas con el Framework Bootstrap, PHP y Base de Datos MySQL en el Período 2020,” Instituto Superior Universitario “San Gabriel,” Riobamba, 2022.

[20] C. Purificación, “Sistema web para mejorar el proceso de control de asistencia del personal administrativo en el instituto de educación Superior Tecnológico Público Eleazar Guzmán Barrón, de la ciudad de Huaraz, 2022,” Universidad Nacional Santiago Antúnez de Mayolo, Huaraz, 2022.

[21] O. Carbajal and A. Minaya, “Aplicativo Móvil para el control biométrico de asistencia del personal docente en la institución educativa San Antonio de Padua,” Universidad Cesar Vallejo, Lima, 2023.

[22] I. Somerville, *Ingeniería del Software 7ma. Ed.*, Séptima edición. Madrid: Pearson Educación S.A, 2005.

[23] Amazon Web Services, “¿Qué es una aplicación web? - Explicación de las aplicaciones web - AWS,” Amazon Web Services, Inc. Accessed: Mar. 18, 2024. [Online]. Available: https://aws.amazon.com/es/what-is/web-application/

[24] Amazon Web Services, “Front End frente a back-end: diferencia entre el desarrollo de aplicaciones - AWS,” Amazon Web Services, Inc. Accessed: Mar. 18, 2024. [Online]. Available: https://aws.amazon.com/es/compare/the-difference-between-frontend-and-backend

[25] Oracle, “¿Qué es una base de datos?,” Oracle | Cloud Applications and Cloud Platform. Accessed: Mar. 18, 2024. [Online]. Available: https://www.oracle.com/mx/database/what-is-database/

[26] UNIR, “Framework: qué es, para qué sirve y algunos ejemplos | UNIR FP,” UNIR FP. Accessed: Mar. 18, 2024. [Online]. Available: https://unirfp.unir.net/revista/ingenieria-y-tecnologia/framework/

[27] Red Hat, “¿Qué es una API REST?,” Red Hat - We make open source technologies for the enterprise. Accessed: Mar. 18, 2024. [Online]. Available: https://www.redhat.com/es/topics/api/what-is-a-rest-api

[28] N. Chico, C. Paúl, P. Rodríguez, and D. Iván, “Plataforma Web para la gestión integrada de reservas de laboratorios, aulas y otros espacios para la Universidad de las Fuerzas Armadas ESPE,” Universidad de las Fuerzas Armadas - ESPE, 2023.

[29] NestJS, “Documentation | NestJS - A progressive Node.js framework,” Documentation | NestJS - A progressive Node.js framework. Accessed: Mar. 18, 2024. [Online]. Available: https://docs.nestjs.com/

[30] M. Fersen, “Estudio Comparativo entre SQL Directo y un ORM,” Babahoyo, 2023.

[31] S. Fernández Fraga, “Desarrollo de una aplicación de gestión documental: frontal React y backend GraphQL con almacenamiento git,” Universidad Da Coruña, 2021.

[32] IBM, “¿Qué es la arquitectura de tres niveles? | IBM,” IBM in Deutschland, Österreich und der Schweiz . Accessed: Mar. 18, 2024. [Online]. Available: https://www.ibm.com/mx-es/topics/three-tier-architecture

[33] R. Barroso Abreu, Y. Oliveros Guntín, Y. Álvarez Alfonso, J. Coello Mena, and L. García Álvarez de la Campa, “Metodología ágil Crystal Clear. Un caso de estudio,” *Serie Científica de la Universidad de las Ciencias Informáticas*, vol. 5, 2012.

[34] L. Montoya, J. Sepúlveda, and L. Jiménez, “Análisis comparativo de las metodologías ágiles en el desarrollo de software aplicadas en Colombia,” *Gestión del Talento Humano: Enfoques y Modelos*, pp. 449–463, 2017.

[35] A. Shrivastava, I. Jaggi, N. Katoch, D. Gupta, and S. Gupta, “A Systematic Review on Extreme Programming,” in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jul. 2021. doi: 10.1088/1742-6596/1969/1/012046.

[36] C. Técnicas Aplicadas, M. I. Vinicio Estrada-Velasco, J. I. Alexandra Núñez-Villacis, and W. I. Clemente Cunuhay-Cuchipe, “Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software,” vol. 7, pp. 434–447, 2021, doi: 10.23857/dc.v7i4.2429.

[37] K. Schwaber and J. Sutherland, “La Guía de Scrum,” 2020.

[38] Stack Overflow, “Stack Overflow Developer Survey 2023,” Stack Overflow. Accessed: Apr. 23, 2024. [Online]. Available: https://survey.stackoverflow.co/2023/

[39] E. Saks, “JavaScript frameworks: Angular vs React vs Vue,” Hagaa-Helia University of Applied Sciences, 2019.

[40] J. Cincović and M. Punt, “Comparison: Angular vs. React vs. Vue. Which framework is the best choice?,” *Information Society of Serbia - ISOS*, 2020.

[41] M. Fariz, S. Lazuardy, and D. Anggraini, “Modern Front End Web Architectures with React.Js and Next.Js,” *International Research Journal of Advanced Engineering and Science*, vol. 7, no. 1, pp. 132–141, 2022.

[42] VueJS, “VueJS Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://vuejs.org/

[43] Angular, “Angular Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://angular.io/docs

[44] NextJS, “NextJS Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://nextjs.org/docs

[45] V. Azkarin, R. G. Guntara, and O. Herdiana, “Development of a REST API for Human Resource Information System for Employee Referral Management Domain Using the Express JS Framework and Node.js,” 2023.

[46] P. Szymon Rodzik, “Comparative Analysis of the Net 6 and NestJS Programming Frameworks in Terms of their Suitability for User Authentication and Authorization,” 2023.

[47] Microsoft, “ASP Net Core Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-8.0

[48] Express, “Express Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://expressjs.com/

[49] NestJS, “NestJS Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://docs.nestjs.com/

[50] S. Binani, A. Gutti, and S. Upadhyay, “SQL vs. NoSQL vs. NewSQL- A Comparative Study,” *Communications on Applied Electronics (CAE)*, vol. 6, no. 1, 2016, [Online]. Available: www.caeaccess.org

[51] A. Vatjalainen, “SQL versus NoSQL Comparison case: MySQL versus MongoDB,” 2023.

[52] R. Aghi, S. Mehta, R. Chauhan, S. Chaudhary, and N. Bohra, “A comprehensive comparison of SQL and MongoDB databases,” *International Journal of Scientific and Research Publications*, vol. 5, no. 2, 2015, [Online]. Available: www.ijsrp.org

[53] M. Gines, “Análisis Comparativo del Desempeño de los Sistemas Gestores de Bases de Datos: PostgreSQL y MySQL con los Datos Académicos de la Universidad Nacional Jorge Basadre Grohmann - 2019,” 2022.

[54] J. León, “Análisis Comparativo de Sistemas Gestores de Bases de Datos PostgreSQL y MySQL en Proceso CRUD,” 2020.

[55] A. Solarz and T. Szymczyk, “Oracle 19c, SQL Server 2019, Postgresql 12 and MySQL 8 database systems comparison,” 2020.

[56] Microsoft, “SQL Server Technical Documentation.” Accessed: Apr. 27, 2024. [Online]. Available: https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver16

[57] MySQL, “MySQL Documentation,” MySQL Documentation. Accessed: Apr. 27, 2024. [Online]. Available: https://dev.mysql.com/doc/

[58] PostgreSQL, “PostgreSQL Documentation,” PostgreSQL Documentation. Accessed: Apr. 27, 2024. [Online]. Available: https://www.postgresql.org/docs/

[59] C. Chillagana, “Diseño e Implementación de un Sistema Desktop para la Administración y Gestión Eclesiástica: Aplicación de la Metodología Crystal en la Parroquia San Buenaventura, Ciudad de Latacunga, Provincia de Cotopaxi (2013),” Universidad Técnica de Cotopaxi, 2015.

[60] M. García, “Estudio comparativo entre las metodologías ágiles y las metodologías tradicionales para la gestión de proyectos de software,” Universidad de Oviedo, 2015.

[61] A. Calderón, S. Dámaris, and J. Valverde, *Metodologías Ágiles*. 2007.

[62] A. P. Chicaiza Ayala, “Desarrollo de Software de Nómina de Empleados Utilizando la Metodología Crystal,” Escuela Politécnica del Ejercito, Sangolqui, 2007.

[63] M.-H. Hugo and O.-M. Edgar, “Estimation and Control in Agile Methods for Software Development: a Case Study Lemus-Olalde Cuauhtémoc,” *Ingeniería Investigación y Tecnología*, vol. 9, no. 3, pp. 403–418, 2014.

# ANEXOS

Anexo A. Entrevista de levantamiento de requerimientos

|  |  |  |
| --- | --- | --- |
| N° | Pregunta | Respuesta |
| 01 | ¿Qué funciones específicas espera que realice el sistema o la aplicación que estamos desarrollando? | Claro, me encantaría. Primero y, ante todo, necesitamos un sistema que nos permita gestionar eficientemente la información de nuestros docentes, incluyendo su registro, edición y eliminación, así como también la posibilidad de adjuntar sus currículos vitae. Además, requerimos un módulo de gestión de sedes para administrar los detalles de nuestras tres sedes escolares.  También necesitamos asegurarnos de que los docentes estén marcando su asistencia adecuadamente. Por lo tanto, el sistema debe enviar notificaciones tanto al personal docente como a mí mismo en caso de ausencias no registradas. Además, necesitamos un proceso para justificar atrasos y faltas de los docentes, con la opción de que tanto el personal como los docentes puedan agregar sus justificaciones, las cuales serán validadas por mí o por otro administrador designado.  Además, debemos tener una manera de sincronizar los registros de asistencia de los relojes biométricos de nuestras sedes para alimentar la base de datos del sistema. Esto es crucial para generar informes de asistencia precisos para cada docente, por sede, por jornadas y en casos especiales como docentes que trabajan en múltiples jornadas.  Por último, necesitamos un sistema de inicio de sesión seguro para mí y para otros administradores, así como la capacidad de gestionar el personal administrativo y logístico de la institución. |
| 02 | ¿Qué acciones o tareas debería poder realizar el sistema para satisfacer sus necesidades? | Por supuesto. El sistema debe permitir la gestión completa de los datos de los docentes y del personal administrativo y logístico, incluyendo agregar, ver, editar y eliminar información, como mencioné anteriormente. Además, debe ser capaz de enviar notificaciones automáticas de ausencias no registradas y gestionar el proceso de justificación de atrasos y faltas.  También necesitamos que el sistema sincronice los registros de asistencia de los relojes biométricos de nuestras sedes y genere informes detallados sobre la asistencia de los docentes, incluyendo faltas, atrasos y otros datos relevantes. |
| 03 | ¿Cuáles son las características principales que espera ver en la interfaz de usuario? | La interfaz de usuario debe ser intuitiva y fácil de usar. Debe permitir una navegación sencilla entre las diferentes funciones del sistema, como la gestión de docentes, la gestión de sedes, la justificación de faltas, la generación de informes, entre otras. Además, sería beneficioso contar con paneles de control que muestren datos relevantes de manera clara y concisa. |
| 04 | ¿Existen flujos de trabajo o procesos específicos que el sistema debe ser capaz de manejar? | Sí, hay varios flujos de trabajo que el sistema debe manejar. Por ejemplo, el proceso de registro y gestión de docentes y personal administrativo y logístico, el proceso de marcado de asistencia y notificación de ausencias no registradas, el proceso de justificación de faltas y atrasos, y el proceso de generación de informes de asistencia, son algunos de los principales. |
| 05 | ¿Hay algún requisito relacionado con la generación de informes, consultas o análisis de datos? | Sí, necesitamos que el sistema sea capaz de generar informes detallados sobre la asistencia de los docentes, incluyendo faltas, atrasos y otros datos relevantes. Estos informes deben ser personalizables y fáciles de entender. Además, sería útil poder realizar consultas y análisis de datos para identificar tendencias y patrones en la asistencia del personal.  También debe ser en el formato del Ministerio de Educación para poder enviar los reportes relacionados con la asistencia del personal. |
| 06 | ¿Existen requisitos de rendimiento o velocidad que el sistema debe cumplir para ser considerado funcionalmente aceptable? | Que sea rápido al momento de generar los reportes nada más. |
| 07 | ¿Existen requisitos relacionados con la seguridad de los datos o la protección de la información dentro del sistema? | Sería bueno que los datos delicados de los docentes sea encriptados. |
| 08 | ¿Qué niveles de disponibilidad y confiabilidad espera del sistema? | El sistema debe estar siempre disponible tanto para los docentes que registren sus justificaciones como para los administradores del sistema. El módulo de marcaje no presencial se habilitará solo cuando se requiera por lo demás el personal de la institución debe poder enviar sus justificaciones cuando desee. |
| 09 | ¿Hay alguna restricción específica en cuanto al tiempo de respuesta del sistema o la velocidad de carga de las páginas? | No en realidad, pero sería bueno que se cargue lo antes posible. Que no se demore en la pantalla de carga. |
| 10 | ¿Existen requisitos de escalabilidad para manejar un aumento en la carga de usuarios o datos? | Se debe tener en cuenta que el sistema puede crecer conforme crezca la institución. |
| 11 | ¿Qué requisitos tiene en términos de usabilidad y accesibilidad para garantizar que el sistema sea fácil de usar para todos los usuarios? | Debe ser entendible de manejar para docentes de edad avanzada. |
| 12 | ¿Existen restricciones de compatibilidad con hardware, software o sistemas operativos existentes en su entorno? | No en realidad, pero se debe poder acceder al sistema desde cualquier navegador de internet. |
| 13 | ¿Existen restricciones de tiempo para la entrega del proyecto? | El tiempo límite se lo pone usted como tesista en base a lo que le pidan en la Universidad. |
| 14 | ¿Hay restricciones de presupuesto que deban tenerse en cuenta? | Se debe ocupar el menor presupuesto posible debido a la dificultad para conseguir recursos dentro de la institución. |
| 15 | ¿Existen restricciones tecnológicas o de recursos? | No, ninguna. |
| 16 | ¿Hay alguna consideración específica que debamos tener en cuenta en relación con la interfaz de usuario? | Se debe personalizar el sistema con los colores y sello de la institución. |
| 17 | ¿Existen requisitos de seguridad o privacidad que deban ser abordados? | Solo los mencionados con respecto a la encriptación de los datos delicados de docentes. |
| 18 | ¿Hay algún otro aspecto importante que debamos considerar para el éxito del proyecto? | Un aspecto importante en el sistema es que el personal pueda registrar su asistencia desde cualquiera de las 3 sedes de la institución. |

Anexo B. Entrevista de refinamiento de requerimientos

**Requerimientos funcionales**

**RF01: Gestión de Docentes (CRUD)**

**Descripción**: El sistema debe permitir al administrador agregar, ver, editar y eliminar la información de los docentes. Adicionalmente un docente puede estar habilitado o no dentro del sistema por motivos de permisos médicos reportados previamente o permisos especiales.

**Preguntas planteadas**:

1. ¿Qué campos específicos del CV del docente deben incluirse? (Educación, experiencia laboral, habilidades, etc.)

Además de los que ya se mencionó que se encuentran en la matriz Excel que le proporcione, sería bueno que se agregue los cursos realizados por el personal, las categorías de cada docente y datos médicos relevantes sobre cada docente como lo son el tipo de sangre, enfermedades, alergias y si toma medicamentos.

1. ¿Cómo se identificará la identidad del docente al agregar o editar su información?

Al docente se lo puede identificar con su cédula dentro del sistema.

1. ¿Un docente puede pertenecer a más de una sede?

Si, un docente puede dar clases en más de una sede como es el caso de los docentes de materias especiales como cultura física.

1. ¿Un docente puede tener más de un horario durante la semana?

Claro que sí, existen varios docentes que tienen horarios diferentes cada día de la semana.

**RF02: Gestión de Sedes (CRUD)**

**Descripción**: El administrador debe poder gestionar la información de las tres sedes de la Unidad Educativa Rumiñahui.

**Preguntas planteadas**:

1. ¿Qué información exacta se debe gestionar para cada sede? (Ubicación, capacidad, horarios, etc.)

Por cada sede sería bueno que se agregue el nombre por el cual se conoce como por ejemplo la sede 2 es conocida como el ex Irene Caicedo, también sería bueno agregar la ubicación y el número de docentes fijos y docentes especiales que son los que trabajan en varias sedes.

**RF03: Envío de Notificaciones de Marcaje de Asistencia**

**Descripción**: El sistema debe enviar notificaciones al administrador del sistema y a los docentes que no registren su asistencia en los biométricos de la institución.

**Preguntas planteadas**:

1. ¿Qué tipo de notificaciones se enviarán? (Correo electrónico, mensajes en la aplicación, etc.)

Las notificaciones se enviarán al correo electrónico provisto por cada docente.

1. ¿Cuándo se enviarán las notificaciones?

Sería bueno que las notificaciones se envíen pasando 5 minutos luego de la hora de marcaje para que el docente pueda tener la posibilidad de notificar de forma oportuna el porqué del atraso.

**RF04: Gestión de Justificaciones de Docentes**

**Descripción**: El administrador debe poder registrar las justificaciones de atrasos y faltas de los docentes con su respectivo certificado. Además, los docentes y el personal de la institución pueden agregar su justificación en un módulo al cual tengan acceso, esta justificación debe ser validada por el administrador.

**Preguntas planteadas**:

1. ¿Cómo se validará la justificación proporcionada por los docentes y el personal?

Para justificar las inasistencias de debe seguir el formulario que se maneja a nivel de la institución y en caso de permiso médico por el ministerio de educación.

1. ¿Qué información específica debe incluirse en el certificado de justificación?

Toda la información que se encuentra en los documentos que le mencione.

1. ¿Todo el personal de la institución debe poder justificar su inasistencia o atraso?

Por supuesto que sí.

1. ¿El personal de la empresa debe poder acceder desde su hogar a esta función?

Sería excelente. También sería importante que el personal pueda subir un documento que justifique su inasistencia además del formulario mencionado. También es importante entender que no todas las justificaciones serán validadas de inmediato por lo cual sería importante manejar estados en cada justificación.

**RF05: Sincronización de los Registros de Asistencia de los Relojes Biométricos de las Sedes**

**Descripción**: El administrador debe poder sincronizar la información de marcado de asistencia de las tres sedes de la institución. Esto alimenta al sistema con la información de marcaje de cada docente.

**Preguntas planteadas**:

1. ¿Con qué frecuencia se realizará la sincronización?

Sería todos los días en la mañana para poder tener el registro del día anterior.

1. ¿Cómo se manejarán los conflictos si hay datos duplicados o inconsistentes?

Se puede resaltar las inconsistencias dentro del sistema, puede ser usando el color rojo.

**RF06: Generación de Reportes**

**Descripción**: El administrador debe poder generar reportes de la asistencia de cada docente, por sede, por jornadas y casos especiales (docentes que trabajan en más de una jornada), faltas y atrasos.

**Preguntas Planteadas**:

1. ¿Qué formatos de reporte se requieren? (PDF, Excel, etc.)

El reporte es en el formato dado por el Distrito, que se encuentra en formato Excel. Ese no se puede modificar.

1. ¿Cuáles son los reportes específicos que necesita?

Serían los reportes de asistencia donde se visualicen los ausentes, atrasos, inconsistencias, reportes por docentes, que se pueda generar una hoja de vida por cada docente.

1. ¿Qué información específica debe incluirse en cada tipo de reporte?

En el de asistencia se debe incluir todo lo que se encuentra en el formato dado, en los otros sería toda la información relevante del docente.

**RF07: Inicio de Sesión**

**Descripción**: El administrador debe poder iniciar sesión de forma segura en el sistema.

**Preguntas Planteadas**:

1. ¿Qué medidas de seguridad se implementarán para proteger el inicio de sesión?

Sería bueno que se cree un usuario específico para cada docente y que la contraseña sea la cédula de cada docente. Esto para evitar que se olviden la contraseña de uso del sistema.

1. ¿Quiénes pueden iniciar sesión en el sistema?

Se debe diferenciar los módulos del sistema, sería el de administración al cual solo debería tener acceso yo y en caso de que se solicite alguna autoridad de la institución. Los docentes deben tener acceso al módulo de marcaje de asistencia no presencial y de registro de justificaciones.

**RF08: Gestión del Personal Administrativo de la Institución (CRUD)**

**Descripción**: El sistema debe permitir al administrador agregar, ver, editar y eliminar la información del personal administrativo de la institución.

**Preguntas Planteadas**:

1. ¿Qué roles específicos tendrán los miembros del personal administrativo?

Ninguno en específico.

1. ¿Qué información específica del personal administrativo se necesita?

Lo mismo que se necesita de todo el personal de la institución.

**RF09: Gestión del Personal de Logística de la Institución (CRUD)**

**Descripción**: El sistema debe permitir al administrador agregar, ver, editar y eliminar la información del personal logístico de la institución.

**Preguntas Planteadas**:

1. ¿Qué funciones específicas desempeñarán los miembros del personal logístico?

Son los encargados de la limpieza, guardias y todo lo que tiene que ver con el mantenimiento de la institución.

1. ¿Qué información específica del personal de logística se necesita?

De igual manera toda la información como si fuera un docente.

**RF10: Marcaje de Asistencia en Caso de No Presencialidad en la Institución**

**Descripción**: El sistema debe permitir al personal de la institución marcar su asistencia incluso cuando no estén físicamente presentes en la institución. Esto puede ocurrir debido a circunstancias como teletrabajo, licencias médicas, capacitaciones fuera del lugar de trabajo, o cualquier otra razón válida.

Este módulo de marcaje de asistencia se activará únicamente cuando sea necesario y puede ser habilitado por cada sede de la institución de manera independiente.

**Preguntas Planteadas**:

1. ¿Cómo se autenticará la identidad del personal que realiza el marcaje de asistencia remoto? ¿Se utilizarán credenciales de acceso o algún otro método?

Claro como se menciono antes deberán ingresar con su usuario y su contraseña que sería la cédula.

1. ¿Qué información específica se registrará durante el marcaje de asistencia remoto? ¿Solo la hora de entrada/salida o se incluirán otros detalles?

Solo la hora de entrada y salida, mientras más simple sea es mejor.

1. ¿Cómo se manejarán las situaciones excepcionales, como el marcaje de asistencia fuera de horario o en días no laborables?

Se resaltarán como inconsistencias dentro del sistema.

**RF11: Gestión de la asistencia de los empleados de la institución**

**Descripción:** El sistema debe permitir visualizar las incidencias de asistencia que hayan ocurrido durante un lapso dado. Se debe identificar los docentes que no han registrado su asistencia.

**Preguntas Planteadas:**

1. ¿Existe algún otro tipo de información que se necesite en este módulo?

Sería importante resaltar las inconsistencias en los tiempos planteados. En el sistema para poder ver qué docentes las tienen.

1. ¿Cuál es la información principal que se debe visualizar dentro del sistema con respecto a la asistencia del personal de la institución?

La asistencia por cada día

**Preguntas adicionales**

1. **¿Existe algún requerimiento adicional que no se haya mencionado durante esta entrevista?**

Sería importante que se diferencie la sede en la cual se realizó el marcaje o si fue no presencial que también se identifique.

1. **¿Existe algún comentario adicional sobre los requerimientos funcionales del sistema?**

**No ninguno además de los que ya se habló en la entrevista pasada y en esta.**

**Requerimientos no funcionales**

**RNF01: Seguridad de la Información**

**Descripción**: Todos los datos almacenados en el sistema deben estar encriptados para garantizar la seguridad y privacidad de la información.

**Preguntas Planteadas**:

1. **¿Qué algoritmo de encriptación se utilizará? (Por ejemplo, AES, RSA, etc.)**

**Eso queda a criterio de usted, no tengo mucho conocimiento respecto al tema.**

1. **¿Cómo se gestionarán las claves de encriptación y su almacenamiento seguro?**

**De igual manera eso quedaría a criterio suyo.**

**RNF02: Identidad del Sistema**

**Descripción**: La interfaz de la aplicación debe estar acorde a los colores institucionales.

**Preguntas Planteadas**:

1. ¿Cuáles son los colores específicos que deben utilizarse? ¿Existe una paleta de colores institucionales definida?

Sería el color azul que predomine, el resto pueden ser verde, blanco, rojo y negro que creo que son los colores básicos.

**RNF05: Marcaje de Asistencia en Diferentes Sedes**

**Descripción**: El personal de la institución debe poder registrar su asistencia en cualquiera de las tres sedes institucionales.

**Preguntas Planteadas**:

1. ¿Cómo se identificará la sede específica durante el marcaje de asistencia?

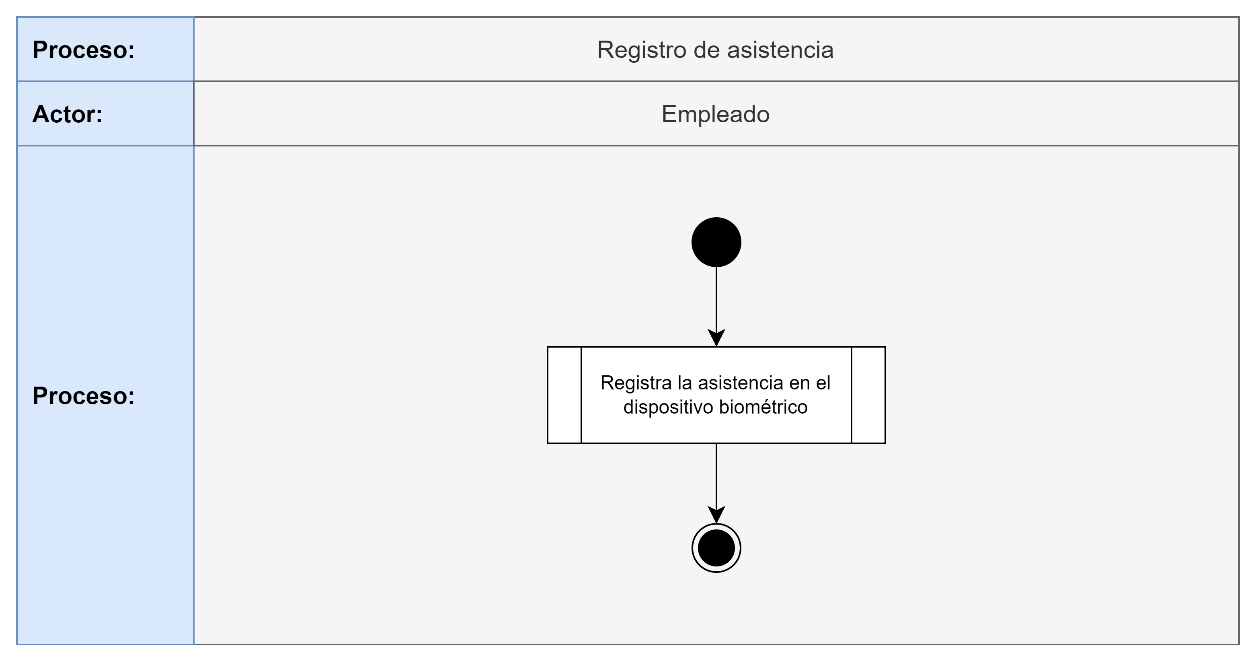
Eso puede ser identificando el biométrico en el cual se registra la asistencia.

1. ¿Se requiere algún tipo de validación para asegurar que el personal solo marque asistencia en la sede correcta?

No ninguna, es más sería importante que los docentes puedan registrar su asistencia en cualquiera de las sedes.

Anexo C. Documento de formalización de requerimientos - IEEE 830

Anexo D. Diagrama de proceso de registro de asistencia



Anexo E. Diagrama de proceso de reporte de asistencia de personal

Diagrama

Descripción generada automáticamente

Anexo F. Diagrama de proceso de registro de asistencia no presencial

Diagrama

Descripción generada automáticamente

Anexo G. Diagrama de proceso de justificación de inasistencia

Diagrama

Descripción generada automáticamente

Anexo H. Diagrama de proceso de generación de reportes de personal

Diagrama

Descripción generada automáticamente