

**INSTITUTO TÉCNICO INCOS PANDO**  
**CARRERA SISTEMAS INFORMÁTICOS**



**PROYECTO DE GRADO**

**SISTEMA INFORMÁTICO DE CONTROL DE ENTRADA  
Y SALIDA DEL PERSONAL DEL AEROPUERTO “CAP.  
ANIBAL ARAB FADUL” DE COBIJA-PANDO**

**PARA OPTAR AL TÍTULO DE TÉCNICO SUPERIOR EN  
SISTEMAS INFORMÁTICOS**

**POSTULANTES:** JUANA LUDIM ADUVIRI SALGADO  
REYNALDO HUANCA MORALES

**TUTOR:** ING. PEDRO VASQUEZ PEREZ

**COBIJA - PANDO - BOLIVIA**  
**2023**

### **DEDICATORIA**

"Dedicamos este proyecto de grado a nuestras familias, quienes nos han brindado un amor incondicional y un apoyo invaluable a lo largo de todo el proceso. También deseamos dedicar este trabajo a nuestros amigos y compañeros de clase, quienes han sido parte fundamental de nuestro aprendizaje y crecimiento profesional. Por último, expresamos nuestro sincero agradecimiento a nuestro tutor y al instituto técnico INCOS-PANDO, por proporcionarnos las herramientas necesarias para llevar a cabo este proyecto con éxito."

## **AGRADECIMIENTOS**

En primer lugar, queremos expresar nuestro profundo agradecimiento a Dios por habernos otorgado una nueva oportunidad en nuestras vidas para crecer como personas y seres humanos.

No podemos pasar por alto a nuestro apreciado tutor, el Ing. Pedro Vásquez Pérez, cuyos valiosos comentarios y observaciones nos han permitido culminar con éxito este proyecto. Su experiencia y orientación fueron fundamentales en nuestra formación.

A la Lic. Natalia Palma Apaza, queremos agradecerle su asesoría, dedicación, tiempo y guía constante en el desarrollo de este proyecto. Su apoyo nos ha brindado confianza y nos ha impulsado a dar lo mejor de nosotros mismos.

Queremos expresar nuestro especial reconocimiento al Gerente Administrativo de NAABOL, quien ha brindado un apoyo incondicional al permitirnos desarrollar este proyecto en su institución. Su confianza en nuestro trabajo ha sido un estímulo importante para nosotros.

A nuestros amados padres, les expresamos nuestro más profundo agradecimiento por su constante apoyo a lo largo de nuestra época de formación. Sus palabras de aliento y su respaldo incondicional nos han dado fuerzas para seguir adelante.

Queremos hacer mención especial a nuestros compañeros, quienes generosamente nos han brindado sus valiosos consejos y apoyo en cada etapa de este proyecto. Su colaboración ha sido fundamental en nuestro crecimiento y aprendizaje.

Por último, extendemos nuestro agradecimiento al Instituto Técnico INCOS-PANDO por abrirnos las puertas de sus aulas y por inculcarnos conocimientos, valores y principios que nos han guiado en nuestro camino hacia el éxito. Su compromiso con la educación ha dejado una huella duradera en nosotros.

## RESUMEN

El objetivo principal del proyecto de grado es desarrollar un sistema informático de control de entrada y salida del personal del aeropuerto "Cap. Anibal Arab Fadul" mediante la implementación de la tecnología de lectura de códigos QR. El propósito de esta iniciativa es mejorar el control de acceso al aeropuerto, y se identificaron una serie de beneficios asociados a este sistema, como el incremento de la eficiencia operativa, la simplificación del proceso de registro, la reducción de errores y la generación de informes en tiempo real.

El registro manual de entrada y salida del personal del aeropuerto provoca demoras y falta de precisión en los datos. Por lo tanto, se implementó un sistema automatizado que mejora tanto el control como la seguridad, sin incurrir en costos adicionales significativos. Las dificultades permanentes al registro manual incluían la falta de información precisa, el tiempo empleado en el proceso, la demora en la elaboración de informes y la falta de registros actualizados de personal.

El aeropuerto "Cap. Anibal Arab Fadul" está ubicado en la ciudad de Cobija, Pando, este aeropuerto que brinda servicios de transporte aéreo a nivel nacional, cuenta con instituciones al interior en las cuales trabajan varios empleados para el buen funcionamiento del aeropuerto, es por eso que NAABOL (Navegación Aérea y Aeropuertos Bolivianos) empezó a registrar el ingreso y salida de cada persona que trabaja al interior del aeropuerto.

En la ejecución de este proyecto, se optó por utilizar la metodología ágil XP (Extreme Programming). Asimismo, se emplearon diversas herramientas, entre ellas Visual CODE, MySQL, PHP, HTML, JavaScript y Simple QR Code, con el objetivo de garantizar un funcionamiento eficiente del sistema.

Este sistema automatizado de control de entrada y salida del personal en el aeropuerto, basado en la tecnología de lectura de códigos QR, contribuye a mejorar de manera significativa la eficiencia, precisión y seguridad en el proceso de registro y control de acceso al aeropuerto.

## ÍNDICE

INTRODUCCIÓN .....	1
1. CAPITULO I.....	3
1.1. TEMA .....	3
1.2. DIAGNÓSTICO Y JUSTIFICACIÓN.....	3
1.2.1. DIAGNÓSTICO .....	3
1.2.2. JUSTIFICACIÓN .....	3
1.3. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA TÉCNICO/TECNOLÓGICO .....	4
1.3.1. PLANTEAMIENTO DEL PROBLEMA .....	4
1.3.2. FORMULACIÓN DEL PROBLEMA TÉCNICO/TÉCNOLOGICO .....	5
1.4. OBJETIVO GENERAL Y OBJETIVOS ESPECÍFICOS .....	5
1.4.1. OBJETIVO GENERAL .....	5
1.4.2. OBJETIVOS ESPECÍFICOS.....	5
1.5. ENFOQUE METODOLÓGICO .....	5
1.5.1. ENFOQUE METODOLÓGICO DE LA INVESTIGACIÓN .....	5
1.5.1.1. TECNICAS DE REQUERIMIENTOS .....	6
1.5.2. ENFOQUE METODOLÓGICO DE DESARROLLO DE SOFTWARE..	6
2. CAPITULO II MARCO TEÓRICO CONCEPTUAL.....	7
2.1. MARCO LEGAL.....	7
2.2. NORMAS DE SEGURIDAD DEL AEROPUERTO .....	7
2.2.1. ZONAS DE SEGURIDAD RESTRINGIDA .....	7
2.2.2. PROCEDIMIENTOS RELACIONADOS A LOS SISTEMAS DE CONTROL DE ACCESO .....	8
2.3. TECNOLOGÍA DE LECTURA DE CÓDIGO QR .....	8
2.3.1. CARACTERÍSTICAS GENERALES .....	9
2.4. SISTEMAS DE INFORMACIÓN.....	10
2.4.1. COMPONENTES BÁSICOS.....	10
2.5. HERRAMIENTAS DE DESARROLLO DE SOFTWARE.....	11
2.6. LENGUAJE DE PROGRAMACIÓN.....	11

2.6.1.	PHP .....	11
2.6.2.	JAVASCRIPT .....	13
2.6.3.	HTML .....	13
2.7.	SISTEMA GESTOR DE BASE DE DATOS .....	14
2.7.1.	GESTORES DE BASE DE DATOS RELACIONALES .....	14
2.7.2.	GESTORES DE BASE DE DATOS NO RELACIONALES .....	15
2.7.3.	MYSQL.....	15
2.7.3.1.	CARACTERÍSTICAS PRINCIPALES .....	15
2.7.3.2.	VENTAJAS.....	16
2.7.3.3.	DESVENTAJAS .....	16
2.8.	METODOLOGÍA DE DESARROLLO DE SOFTWARE .....	17
2.8.1.	METODOLOGÍAS TRADICIONALES.....	17
2.8.1.1.	PRINCIPALES METODOLOGÍAS TRADICIONALES.....	17
2.8.2.	METODOLOGÍAS AGILES.....	19
2.8.2.1.	MANIFIESTO ÁGIL.....	20
2.8.2.2.	PRINCIPALES METODOLOGÍAS ÁGILES.....	21
2.8.3.	DIFERENCIAS ENTRE METODOLOGÍAS TRADICIONALES Y ÁGILES .....	23
2.8.4.	METODOLOGÍA DE DESARROLLO DE SOFTWARE XP (EXTREME PROGRAMING) .....	24
2.8.4.1.	CICLO DE VIDA DE LA METODOLOGÍA XP .....	25
2.8.4.2.	LAS FASES DE LA METODOLOGÍA DE PROGRAMACIÓN EXTREMA .....	26
3.	CAPITULO III PROPUESTA DE INOVACIÓN O SOLUCIÓN DEL PROBLEMA .....	31
3.1.	APLICACIÓN DE LA METODOLOGÍA XP (PROGRAMACIÓN EXTREMA).....	31
3.1.1.	FASE 1: PLANEACIÓN .....	31
3.1.1.1.	ANÁLISIS DE REQUERIMIENTOS.....	32
3.1.1.2.	HISTORIAS DE USUARIO .....	32

3.1.1.3. DESCRIPCION DE HISTORIA DE USUARIOS.....	33
3.1.1.4. PLAN DE ENTREGAS.....	37
3.1.1.5. ITERACIONES.....	38
3.1.2. FASE 2: DISEÑO .....	42
3.1.3. FASE 3: CODIFICACIÓN .....	44
RESULTADOS ESPERADOS .....	46
CONCLUSIONES .....	47
RECOMENDACIONES.....	48
FUENTES DE INFORMACIÓN Y BIBLIOGRAFÍA.....	49

## ÍNDICE DE FIGURAS

Figura 2.1	Tecnología de Código QR. ....	9
Figura 2.2	Características de código QR. ....	9
Figura 3.1	Modelo entidad relación. ....	42
Figura 3.2	Modelo de estructura. ....	43
Figura 3.3	Editor de código fuente. ....	44
Figura 3.4	Interfaz. ....	45
Figura 3.5	Iteraciones de prueba. ....	45



## ÍNDICE DE TABLAS

Tabla 2.1	Diferencia de metodologías. ....	23
Tabla 3.1	Fases y procesos de la metodología XP. ....	31
Tabla 3.2	Lista de requerimientos del usuario. ....	32
Tabla 3.3	Historia de Usuario: Registro de personal. ....	33
Tabla 3.4	Historia de Usuario: Impresión de credencial. ....	34
Tabla 3.5	Historia de Usuario: Control general de registros de entrada y salidas. ....	35
Tabla 3.6	Historia de Usuario: Reportes por rango de fechas. ....	35
Tabla 3.7	Historia de Usuario: Usuario Administrador. ....	36
Tabla 3.8	Historia de Usuario: Personal de Seguridad. ....	37
Tabla 3.9	Plan de entregas por historias de usuario. ....	38
Tabla 3.10	TAREA 1: Diseño de formulario de registro de personal. ....	39
Tabla 3.11	TAREA 02: Diseño de la generación de la credencial para el personal. ....	39
Tabla 3.12	TAREA 03: Diseño de control del personal del aeropuerto. ....	39
Tabla 3.13	TAREA 04: Diseño del formulario de acceso a personal de seguridad. ....	40
Tabla 3.14	TAREA 5: Diseño de usuario y contraseña para el personal de seguridad. ....	40
Tabla 3.15	TAREA 1: Diseño del formulario de datos del personal. ....	41
Tabla 3.16	TAREA 2: Modificar del formulario los datos del personal. ....	41
Tabla 3.17	TARJETA CRC: USUARIO. ....	43
Tabla 3.18	TARJETA CRC: EMPLEADO SEGURIDAD. ....	44

## **ÍNDICE DE ANEXOS**

ANEXO 1	: Observación realizada .....	1
ANEXO 2	: Guía de cuestionario .....	2
ANEXO 3	: Cuestionario llenado por el personal de seguridad de NAABOL ..	3

## INTRODUCCIÓN

Actualmente, la informática se ha convertido en un factor muy importante en la vida de una institución y/o empresa, ya sea como herramienta o una aplicación para procesar datos, existen diferentes razones para contar con un sistema de información el cual permita el procesamiento de la misma de forma automática, el almacenamiento y manejo de grandes cantidades de información.

Hoy en día los sistemas de información están basados en computadoras que son objetos de gran consideración en la toma de decisiones oportunas, confiables y efectivas en cuanto a técnicas de planificación, programación y administración con el fin de garantizar su éxito, limitar el tiempo y reducir costos.

Existen diferentes tipos de sistemas automatizados entre las cuales están los sistemas de control de personal, esencialmente en el tema de la asistencia y cumplimiento de los horarios de trabajo, los sistemas de control tradicionales están basados en un control manual o registros escritos de las instituciones, que en muchas ocasiones es susceptible a la alteración de la información y demoras al momento de registrarlas, la cual se evidenció en la institución de NAABOL.

NAABOL (Navegación Aérea y Aeropuertos Bolivianos), es una institución la cual brinda apoyo al tránsito aéreo a todas las aeronaves a nivel nacional del Estado Plurinacional de Bolivia en la cual se evidencio que en algunos aeropuertos no existe un registro automatizado, ya que en diferentes ciudades las instituciones que trabajan independientemente manejan su registro de manera individual, así también se ve que en la ciudad de Cobija-Pando la institución hace el registro manual de todos los empleados que trabajan al interior del aeropuerto.

Debido a esta razón, nace la idea de automatizar el registro del todo el personal que trabaja en el Aeropuerto “Cap. Anibal Arab Fadul” y tiene como finalidad brindar una mejora en el proceso del registro, control de entrada y salida del personal que trabaja al interior del Aeropuerto, aplicando la tecnología de entrada

de datos por medio de un lector de QR (Quick Response = código de respuesta rápida).

El sistema permite registrar la hora y fecha de entrada y salida del personal mediante un lector de QR (Quick Response), dicho dispositivo se conectará a una PC (Computadora Personal) principal en la cual se guardarán los datos del personal, la hora y fecha de ingreso - salida en una base de datos, mantendrá la información actualizada mediante la generación de reportes diarios y mensuales los cuales coadyuvaran en la toma de decisiones al jefe de seguridad del aeropuerto.

La implementación de este proyecto se encarga de automatizar los procesos que realizaba el personal de NAABOL (Navegación Aérea y Aeropuertos Bolivianos) manualmente, reduciendo el tiempo de registro y evitando malos entendidos con los empleados, brindará reportes que permita tener día a día la información actualizada de entrada y salida de cada empleado del aeropuerto.

# **1. CAPITULO I**

## **1.1. TEMA**

El sistema informático de control de entrada y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, mediante un lector de códigos QR que ayuda a mejorar la seguridad del aeropuerto y garantizar la identificación del personal que ingresa y sale de las instalaciones teniendo un mejor control del personal en el aeropuerto. Con el sistema automatizado se reducirán errores humanos, se reducirá tiempo de registro y mejorará la eficiencia de los reportes del aeropuerto.

## **1.2. DIAGNÓSTICO Y JUSTIFICACIÓN**

### **1.2.1. DIAGNÓSTICO**

El aeropuerto “Cap. Anibal Arab Fadul” que encuentra ubicado en la ciudad de Cobija-Pando del Estado Plurinacional de Bolivia, cuenta con instituciones al interior en las cuales trabajan varios empleados para el buen funcionamiento del aeropuerto, es por eso que NAABOL empezó a registrar el ingreso y salida de cada persona que trabaja al interior del aeropuerto de manera manual en planillas de registro preguntando a cada persona ciertos datos antes de ingresar, esto se hace para tener mayor control y seguridad del aeropuerto.

### **1.2.2. JUSTIFICACIÓN**

El desarrollo de un sistema informático automatizado de control de entrada y salida de personal proporciona beneficios significativos para la institución. Entre estos beneficios se incluyen una mayor eficiencia operativa, simplificación del proceso de registro de entrada - salida, reducción en el tiempo dedicado al registro manual del control y la capacidad de realizar un seguimiento actualizado del personal.

Además, con un sistema automatizado, se puede reducir los errores humanos y se puede mejorar la precisión de los registros de entrada y salida, esto garantiza el cumplimiento de las regulaciones laborales.

Para la implementación del proyecto no se requiere mayores costos, la institución como tal, cuenta con los equipos necesarios para implementar el proyecto.

### **1.3. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA TÉCNICO/TECNOLÓGICO**

#### **1.3.1. PLANTEAMIENTO DEL PROBLEMA**

El proceso manual de registro de entrada y salida del personal que desempeña sus labores en el aeropuerto "Cap. Aníbal Arab Fadul" ha dado origen a una serie de problemas que están teniendo un impacto adverso en la gestión y eficiencia de las operaciones. La demora en el registro diario y la compilación de datos de los empleados está generando errores en los registros, lo que, a su vez, está resultando en una disminución de la calidad en el control de seguridad del aeropuerto.

Mediante diferentes consultas directas a los funcionarios de seguridad de NAABOL y observando las tareas que estos realizan, se han identificado las siguientes dificultades de esta institución:

- Dificultad en el momento de pedir la información precisa para el registro de los empleados del aeropuerto.
- El registro del personal de manera manual diario, se emplea demasiado tiempo.
- Existe una demora en la elaboración de reportes del personal por la forma manual que se realiza.
- Falta de un registro actualizado y preciso del personal que trabaja en el aeropuerto, lo que afecta negativamente la calidad del control y la seguridad en el aeropuerto.

Por todo lo anterior expuesto podemos indicar entonces que el problema principal es: la deficiencia en el proceso de registro manual diario de entrada y salida del personal en el aeropuerto "Cap. Anibal Arab Fadul" de la ciudad de Cobija-Pando.

### **1.3.2. FORMULACIÓN DEL PROBLEMA TÉCNICO/TÉCNOLOGICO**

¿Cómo podemos mejorar el proceso de registro manual de entrada y salida del personal del aeropuerto “Cap. Aníbal Arab Fadul” para reducir el tiempo de registro, aumentar la precisión de los registros, realizar reportes inmediatos y elevar la calidad del control de seguridad en el aeropuerto?

### **1.4. OBJETIVO GENERAL Y OBJETIVOS ESPECÍFICOS**

#### **1.4.1. OBJETIVO GENERAL**

Desarrollar un sistema informático de procesos de registro de entrada y salida del personal del aeropuerto “Cap. Anibal Arab Fadul” mediante la tecnología QR para mejorar la eficiencia, precisión y reducir el tiempo de registro.

#### **1.4.2. OBJETIVOS ESPECÍFICOS**

- Análisis de requerimientos del proceso actual de registro de entrada y salida del personal en el aeropuerto “Cap. Anibal Arab Fadul” para identificar sus deficiencias y demoras.
- Diseño de los módulos del sistema para el registro automatizado que utilice tecnología QR para agilizar la entrada y salida del personal, enfocándose en la eficiencia y precisión de la recopilación de datos.
- Desarrollo de módulo de integración del código QR a los módulos del sistema y generar reportes, pruebas del sistema de acuerdo a estándares de calidad.

### **1.5. ENFOQUE METODOLÓGICO**

El enfoque metodológico que se usa en el sistema informático de control de ingreso y salida del personal del aeropuerto son: en el enfoque metodológico de la investigación (enfoque metodológico cualitativo) y en el enfoque metodológico del desarrollo del software (metodología XP).

#### **1.5.1. ENFOQUE METODOLÓGICO DE LA INVESTIGACIÓN**

Enfoque metodológico de la investigación cualitativa es el conjunto de estrategias y técnicas utilizadas para llevar a cabo una investigación que busca comprender

y describir una realidad subjetiva a través de la interpretación de datos obtenidos de palabras, textos, discursos, imágenes, etc. (investigalia)

#### **1.5.1.1. TECNICAS DE REQUERIMIENTOS**

Para el análisis de requerimientos del proyecto, se emplearon las siguientes técnicas de recolección de información:

- **OBSERVACIÓN.-** Se aplicó la técnica de observación no participativa, siguiendo la guía establecida. Esta técnica permitió obtener información sin establecer ninguna relación con las personas observadas ni formar parte de su actividad al registrar la entrada y salida de personas del aeropuerto. (Ver Anexo 1).
- **ENTREVISTA.-** Se realizó entrevista al Jefe de aeropuerto “Cap. Anibal Arab Fadul” aplicando una guía de formulario de forma directa y personal. (Ver Anexo 2).
- **CUESTIONARIO.-** Se llevó a cabo una entrevista en formato de cuestionario con preguntas abiertas al jefe de seguridad del aeropuerto, con el objetivo de recopilar información más precisa sobre los problemas que actualmente genera el control de registro de entrada y salida del personal del aeropuerto. (Ver Anexo 3).

#### **1.5.2. ENFOQUE METODOLÓGICO DE DESARROLLO DE SOFTWARE**

El enfoque metodológico de desarrollo de software metodología XP, o Programación Extrema es una metodología ágil de desarrollo de software que se centra en la calidad del software y la satisfacción del cliente. Utiliza prácticas como la planificación ágil, el diseño simple, la codificación en parejas, las pruebas continuas y la integración continua para garantizar que el software se entregue de forma rápida y con calidad. (SYDLE)



## **2. CAPITULO II MARCO TEÓRICO CONCEPTUAL**

### **2.1. MARCO LEGAL**

### **2.2. NORMAS DE SEGURIDAD DEL AEROPUERTO**

#### **2.2.1. ZONAS DE SEGURIDAD RESTRINGIDA**

a) El aeropuerto se divide en dos partes: Parte Aeronáutica y Parte Publica. La parte aeronáutica, se subdivide a su vez en Zona de seguridad restringida y zona Estéril.

b) El número de puntos de acceso a la zona de seguridad restringida, debe limitarse al mínimo necesario, debiendo estar controlado o cerrado.

c) El acceso de personal y vehiculos a la zona de seguridad restringida, debe limitarse a los que tienen una necesidad operativa.

d) Las zonas de seguridad restringidas deben ser patrulladas por el personal de seguridad del aeropuerto, en coordinación con los organismos de seguridad del estado, asignados al aeropuerto.

e) El administrador del aeropuerto en cumplimiento a los procedimientos descritos en su programa de seguridad, deben cumplir las siguientes funciones:

1) Control de todo acceso a toda área de operaciones aéreas del aeropuerto, que incluya:

i) Métodos y procedimientos para prevenir el ingreso de personas y vehículos no autorizados.

ii) Un sistema de identificación y registro de aquellas personas y vehículos autorizados para ingresar.

iii) Un método para diferenciar entre las personas autorizadas para tener acceso parcial y las personas autorizadas para acceder a la totalidad del área de seguridad.

2) Control de movimiento de personas, vehículos y equipos de soporte de tierra dentro de área de operaciones, observando el cumplimiento de la utilización

visible de identificación de acceso aeroportuario de personas y vehículos, en coordinación con los organismos de seguridad del estado.

3) Detección y toma de acción inmediata para controlar toda intrusión o intento de acceso a un área de operación aérea por una persona o personas, cuyo ingreso no está autorizado de acuerdo con el programa de seguridad, en coordinación con los organismos de seguridad del estado.

### **2.2.2. PROCEDIMIENTOS RELACIONADOS A LOS SISTEMAS DE CONTROL DE ACCESO**

a) El administrador del aeropuerto, debe incluir en su programa de seguridad, un sistema, método o procedimiento (mecánico, electrónico y/o informático), para identificar y controlar el acceso de pasajeros, personas y vehículos a las zonas de seguridad restringida del aeropuerto.

b) Tal sistema, método o procedimiento, debe asegurar que solo aquellas personas autorizadas, tengan acceso a las zonas de seguridad restringidas del aeropuerto y deben proveer medios específicos que aseguren que dicho acceso sea negado inmediatamente en el (los) punto(s) de control a aquellas personas cuya autorización de acceso hubiera caducado o cambiado. El sistema, método o procedimiento, deberá suministrar los medios para diferenciar entre personas autorizadas para acceder a la totalidad del aérea de seguridad. El sistema, método o procedimiento, deberá ser capaz de limitar o negar el ingreso a cualquier individuo no autorizado, con hora y fecha. (DGAC, 2017)

### **2.3. TECNOLOGÍA DE LECTURA DE CÓDIGO QR**

Un código QR (Quick Response), es la evolución del código de barras. Es un módulo para almacenar la información en una matriz de datos o en un código de barras bidimensional. Fue creado en 1994 por la compañía japonesa denso wave, subsidiaria de Toyota. Presenta tres cuadrados en las esquinas que permiten detectar la posición del código al lector. El objetivo de los creadores, un equipo de dos personas dirigido por Masahiro Hara, fue que el código permitiera que su contenido se leyera a alta velocidad.

**Figura 2.1** *Tecnología de Código QR.*



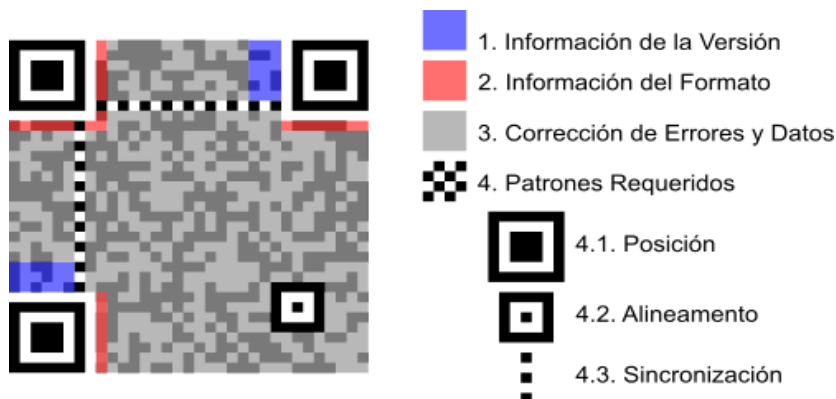
**Fuente:** Wikipedia.

Los códigos QR son una herramienta importante en la tecnología actual proporcionando una forma rápida y conveniente de compartir información, almacenando información en un matriz de datos, los códigos QR se pueden leer con un dispositivo móvil, cámara web, dispositivos de lector de QR, proporcionando información sobre una amplia variedad de temas.

### **2.3.1. CARACTERÍSTICAS GENERALES**

La estructura general de un código QR es una matriz bidimensional de módulos de dos colores contrastados, en principio blancos y negros, hay varias versiones de códigos QR según la cantidad de módulos que forman la matriz: van desde la versión 1 (con una matriz de 21 x 21 módulos) hasta la versión 10 (con 177 x 177 módulos). Las versiones de más módulos admiten mayor cantidad de información en el código.

**Figura 2.2** *Características de código QR.*



**Fuente:** Wikipedia

En la actualidad encontramos códigos QR en los que se pueden incorporar imágenes personalizadas, contribuyendo a dar un aspecto más artísticos y personal, siendo muy útil en los códigos que dan un acceso a la información de una persona.

Los códigos QR también pueden leerse desde computadoras personales, teléfonos inteligentes o tabletas, mediante dispositivos de captura de imagen como escáneres o cámaras de fotos, programas que lean los datos QR y una conexión a internet para las direcciones web. (Wikipedia, Wikipedia, enciclopedia libre, s.f.)

## **2.4. SISTEMAS DE INFORMACIÓN**

Un sistema de información es el conjunto de agente, código y procesos que interactúan coordinadamente entre si con un fin común o propósito.

En informática, los sistemas de información ayudan a administrar, recolectar, recuperar, procesar, almacenar y distribuir información relevante para los procesos fundamentales y las particularidades de cada organización. La importancia de un sistema informático radica en la eficiencia en la correlación de una gran cantidad de datos ingresados a través de procesos diseñados para cada área con el objetivo de producir información válida para la posterior toma de decisiones.

Un sistema de información se destaca por su diseño, facilidad de uso, flexibilidad, mantenimiento automático de los registros, apoyo en la toma de decisiones críticas y mantener el anonimato en informaciones irrelevantes.

### **2.4.1. COMPONENTES BÁSICOS**

Un sistema de información debe cumplir con los siguientes componentes básicos interactuando entre sí:

- El hardware, equipo físico utilizado para procesar y almacenar datos.
- El software y los procedimientos utilizados para transformar y extraer información.

- Los datos que representan las actividades de la empresa.
- La red que permite compartir recursos entre computadoras y dispositivos.
- Las personas que desarrollan, mantienen y utilizan el sistema. (Wikipedia, Wikipedia enciclopedia libre, s.f.)

## **2.5. HERRAMIENTAS DE DESARROLLO DE SOFTWARE**

### **2.6. LENGUAJE DE PROGRAMACIÓN**

Un programa se escribe en un lenguaje de programación y las operaciones que conducen a expresar un algoritmo en forma de un programa se llama programación. Así pues, los lenguajes utilizados para escribir programas de computadoras son los lenguajes de programación y programadores son los escritores y diseñadores de programas. El proceso de traducir un algoritmo en pseudocódigo a un lenguaje de programación se denomina codificación, y el algoritmo escrito en un lenguaje de programación se denomina código fuente.

Hoy en día, la mayoría de los programadores emplean lenguajes de programación como C++, C, C#, Java, Visual Basic, XML, HTML, Perl, PHP, JavaScript, etc..., aunque todavía se utilizan, sobre todo profesionalmente, los clásicos COBOL, FORTRAN, Pascal o el mítico BASIC. Estos lenguajes se denominan lenguajes de alto nivel y permiten a los profesionales resolver problemas convirtiendo sus algoritmos en programas escritos en alguno de estos lenguajes de programación. (Aguilar, 2008).

La implementación de lenguajes de programación permite el trabajo conjunto y coordinado, a través de un conjunto afín y finito de instrucciones posibles, de diversos programadores o arquitectos de software, para lo cual estos lenguajes imitan, al menos formalmente, la lógica de los lenguajes humanos o naturales.

#### **2.6.1. PHP**

PHP (Hypertext Preprocessor), es un lenguaje del lado del servidor que se caracteriza por su potencia, versatilidad, robustez y modularidad. Los programas escritos en PHP son embebidos directamente en el código HTML y ejecutados

por el servidor web a través de un intérprete antes de transferir al cliente que lo ha solicitado un resultado en forma de código HTML puro. Al ser un lenguaje que sigue la corriente open source, tanto el interprete como su código fuente son totalmente accesibles de forma gratuita en la red.

Por su flexibilidad, PHP resulta un lenguaje muy sencillo de aprender; especialmente para programadores familiarizados con lenguajes como C, Perl o Java, debido a las similitudes de sintaxis entre ellos.

Por su puesto, es un lenguaje multiplataforma; los programas funcionan igual sobre diferentes plataformas, trabajando sobre la mayoría de servidores web y estando preparado para interactuar con más de 20 tipos de bases de datos.

En comparación con otro tipo de tecnologías similares, PHP resulta más rápido, independientemente de la plataforma y más sencillo de aprender y utilizar.

Todas estas características han hecho de este lenguaje uno de los que mayor crecimiento ha experimentado en los últimos años, desde su aparición en 1994. Es de destacar especialmente la facilidad para la conectividad con sistemas gestores de bases de datos a través de un gran número de funciones especializadas. (Angel Cobo, 2005)

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es decir es independiente del navegador, pero sin embargo para que las paginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, muchas variantes Unix (incluido HP-UX, Solaris y OpenBSD), Microsoft Windows, Mac OS X, RISC OS y probablemente alguno más. PHP soporta la mayoría de Web Server y muchos otros. PHP tiene módulos disponibles para la mayoría de los servidores.

### **2.6.2. JAVASCRIPT**

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas.

Una página web dinámica es aquella que incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario.

Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios.

A pesar de su nombre, JavaScript no guarda relación con el lenguaje de programación Java. Legalmente, JavaScript es una marca de la empresa Sun Microsystems. (Perez, 2009)

### **2.6.3. HTML**

Las definiciones existentes de HTML son múltiples y más o menos complejas. Una definición sencilla y de marcado carácter práctico es la que se propone a continuación: HTML es un lenguaje de descripción de hipertexto compuesto por una serie de comandos, marcas, o etiquetas, también denominadas “Tags” que permiten definir la estructura lógica de un documento web y establecer los atributos del mismo (color de texto, contenidos multimedia, hipervínculos, etc).

En resumen, es un lenguaje que permite crear páginas web y para ello utiliza unos comandos o etiquetas que indican o marcan qué se debe mostrar y de qué forma.

Los comandos siempre van incluidos entre los signos < > e insertados en el propio texto que compone el contenido de la página. Especifican su estructura (las distintas partes de la página) y formato. Además, permiten la inserción de contenidos especiales como imágenes, videos, sonidos, etc. (Angel Cobo, 2005)

## **2.7. SISTEMA GESTOR DE BASE DE DATOS**

Un sistema de gestor de base de datos (SGBD) o database management system (DBMS) reúne el conjunto de programas empleados para administrar y gestionar la información presente en una base de datos. Estos sirven de interfaz entre la base de datos, las aplicaciones y el usuario, permitiendo recuperar la información de forma sencilla, rápida y eficiente.

Algunas de sus funciones son permitir la definición y almacenamiento de la información, acceder a los datos y modificarlos, garantizando la seguridad e integridad de los mismos. Además, se usan para efectuar análisis y generar informes, de la forma más eficiente posible.

Para funcionar correctamente, los gestores de base de datos están compuestos por un lenguaje de definición de datos, uno de manipulación y otro de consulta.

### **2.7.1. GESTORES DE BASE DE DATOS RELACIONALES**

Se destinan al manejo de base de datos relacionales, las cuales permiten que cualquier tabla establezca vínculos o relaciones con otra. En este modelo de base de datos la información almacenada se representa en forma de columnas (atributos) y filas (registros). Los atributos especifican el tipo de datos que se manejan y cada registro contiene el valor de este tipo de datos.

El medio de acceso más común a este tipo de gestores de base de datos es SQL (Structured Query Language) o lenguaje de consulta estructurada.

Las funcionalidades básicas de gestores de base de datos relacionales son: Creación, lectura, actualización, eliminación.

Estas permiten crear un sistema bien organizado que impulsa el tratamiento adecuado de los datos. Algunos de los gestores de base de datos relacionales más habituales son: Oracle MySQL, Microsoft SQL Server, MariaDB, MySQL.



### **2.7.2. GESTORES DE BASE DE DATOS NO RELACIONALES**

Esta clase de sistemas no requiere emplear estructuras de datos fijos, como las tablas. Por este motivo, son muy útiles para manejar grandes cantidades de información manteniendo una buena velocidad de respuesta.

Se caracterizan porque no son rígidas, permiten gestionar la información con una alta escalabilidad horizontal y emplean muchos más nodos que los gestores de bases de datos relacionales. Por todo ello, se utilizan en aplicaciones que necesitan leer y/o escribir grandes volúmenes de datos.

Entre los principales gestores de base de datos no relacionales destacan: MongoDB, Redis, Cassandra, DynamoDB. (Unir, 2023)

### **2.7.3. MYSQL**

MySQL es un sistema gestor de base de datos relacional muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, es una opción atractiva tanto para aplicaciones comerciales, como de entretenimiento precisamente por su facilidad de uso y tiempo reducido de puesta en marcha. Esto y su libre distribución en internet bajo licencia GPL le otorgan como beneficios adicionales (no menos importantes) contar con un alto grado de estabilidad y un rápido desarrollo. (Luis Alberto Casillas Santillan)

#### **2.7.3.1. CARACTERÍSTICAS PRINCIPALES**

Inicialmente, MySQL carecía de algunos elementos esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de esto, atrajo a los desarrolladores de páginas web con contenido dinámico, debido a su simplicidad, de tal manera que los elementos faltantes fueron complementados por la vía de las aplicaciones que la utilizan. Poco a poco estos elementos faltantes, están siendo incorporados tanto por desarrolladores internos, como por desarrolladores de software libre.

En las últimas versiones se pueden destacar las siguientes características principales:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Aprovecha la potencia de sistemas de multiprocesos, gracias a su implementación multihilo.
- Flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas.

#### **2.7.3.2. VENTAJAS**

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de sistemas operativos.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Conectividad y seguridad.

#### **2.7.3.3. DESVENTAJAS**

- Un gran porcentaje de las utilidades de MySQL no están documentadas.
- No es intuitivo, como otros programas (Access).

## **2.8. METODOLOGÍA DE DESARROLLO DE SOFTWARE**

Las metodologías de desarrollo de software son un conjunto de técnicas y métodos organizativos que se usan para diseñar soluciones de software. El objetivo de estas metodologías es organizar los equipos de trabajo para que desarrollen las funciones de un programa de la mejor manera posible. Existen diferentes metodologías de desarrollo de software entre las que podemos mencionar están las metodologías tradicionales y las metodologías ágiles

### **2.8.1. METODOLOGÍAS TRADICIONALES**

Las metodologías tradicionales son denominadas, a veces, de forma despectiva, como metodologías pesadas. Centran su atención en llevar una documentación exhaustiva de todo el proyecto, la planificación y el control del mismo, en especificaciones precisas de requisitos y modelado y en cumplir con un plan de trabajo, definido todo esto, en la fase inicial del desarrollo del proyecto. (Maida, 2015)

#### **2.8.1.1. PRINCIPALES METODOLOGÍAS TRADICIONALES**

➤ **MODELO EN CASCADA.-** Sugiere un enfoque sistemático y secuencial, disciplinado y basado en análisis, diseño, pruebas y mantenimiento. Al final de cada etapa se reúnen y revisan los documentos para garantizar que se cumplen los requerimientos antes de avanzar a la fase siguiente, sus etapas son: requisitos, análisis, diseño, codificación y prueba.

➤ **MODELO EN CASCADA EN V.-** Se basa en el modelo en cascada con la innovación de procurar actividades de pruebas más efectivas y productivas mediante la introducción de validaciones en la medida en que se avanza en el proyecto; dado que en el modelo tradicional las pruebas se introducían al final los efectos aparecían en forma tardía. Las pruebas necesitan empezarse lo más pronto posible en el ciclo de vida y estas actividades deberían ser llevadas a cabo en paralelo con las actividades de desarrollo.

➤ **MODELO DE DESARROLLO INCREMENTAL.-** Se basa en el desarrollo a partir del incremento de la funcionalidad del programa, se puede considerar un

precursor de las modernas metodologías iterativas. El primer incremento es a menudo un desarrollo esencial, apenas con los requisitos básicos, cada incremento representa una entrega escalable. Los primeros incrementos son versiones incompletas del producto final, pero proporcionan al usuario la funcionalidad.

➤ **MODELO DE DESARROLLO EVOLUTIVO (ESPIRAL).**- Conjugaba una naturaleza iterativa en la construcción de prototipos con aspectos controlados y sistemáticos del modelo en cascada. Cuando se aplica este modelo en espiral, el software se desarrolla en una serie de entregas evolutivas. Cada una de las actividades del marco de trabajo representan un segmento de la ruta del espiral. En cada ciclo repetitivo va ganando madurez el producto final.

➤ **MODELO EVOLUTIVO POR PROTOTIPOS.**- Modelo iterativo a través del cual es posible identificar los requerimientos del cliente mediante la construcción de un prototipo de funcionalidad simulada; si no se ajusta a la expectativa del cliente, se construye otro prototipo con una definición mejorada, el diseño va evolucionando ajustándose cada vez más al requerimiento, aunque sea su funcionalidad será simulada hasta tanto se aclaran la totalidad de los requerimientos con la validación del último prototipo.

➤ **MODELO DE DESARROLLO BASADO EN COMPONENTES.**- Promueve el uso de desarrollos “pre-fabricados” que se pueden reutilizar, con ello se emplean arquitecturas, diseños y software de una aplicación para construir otra. Representa un paso importante hacia la agilización.

➤ **RAPID APPLICATION DEVELOPMENT (RAD).**- O desarrollo rápido de aplicaciones. La metodología se centra en una lista de tareas y una estructura de desglose del trabajo orientada a la rapidez, aunque no está alineada al “manifiesto ágil” si buscó responder a la necesidad de agilizar las entregas de aplicaciones. Comprende el desarrollo bajo un modelo iterativo, la construcción de prototipos y el uso de utilidades CASE (Computer Aided Software

Engineering) esto es, aplicaciones informáticas dirigidas a aumentar la productividad en todos los aspectos del ciclo de desarrollo.

➤ **RATIONAL UNIFIED PROCESS (RUP).**- Metodología basada en los modelos en cascada y por componentes. Presenta las siguientes características: es dirigido por “casos de uso” esto es, la descripción del servicio que el usuario requiere del sistema y la secuencia de iteraciones usuario – sistema; se centra en la arquitectura, dicta pautas específicas para la construcción del equipo y las escalas de tiempo, es iterativa e incremental. Es una de las metodologías clásicas vigentes y más usadas para el análisis, desarrollo y documentación de sistemas orientados a objetos, muy aplicada en proyectos de gran complejidad y magnitud con apoyo de equipos expertos.

➤ **MICROSOFT SOLUTION FRAMEWORK (MSF).**- Fue introducida por primera vez en 1994 como un conjunto de las mejores prácticas (principios, modelos, disciplinas, conceptos y directrices) en los desarrollos de Software de Microsoft y Microsoft Consulting Service. Es flexible, permite aplicar de manera individual e independiente cada uno de sus componentes, es escalable según la magnitud del proyecto; fundamentada en los modelos espiral y cascada. Profesa la aplicación de 8 principios fundamentales para una mejor organización del trabajo; modelos o esquemas para la organización de los equipos y disciplinas de gestión. (INNOVA, 2018)

### **2.8.2. METODOLOGÍAS AGILES**

Las metodologías de desarrollo ágil buscan elaborar software totalmente funcional en el tiempo o plazo establecido para el desarrollo del proyecto. Utilizan un proceso ágil, es decir que si los requerimientos del software cambian en cualquier etapa en la que se encuentre el proyecto, el equipo debe adaptar el producto a cambios ya que la agilidad como tal es la respuesta efectiva al cambio. (Cevallos, 2015)

### **2.8.2.1. MANIFIESTO ÁGIL**

En una reunión celebrada en febrero de 2001 en Utah-EEUU, nace el término “Ágil” aplicado al desarrollo de software. En esta reunión participaron un grupo de 17 expertos de la industria del software, su objetivo fue proyectar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto.

Tras esta reunión se creó The agile Alliance, una organización, sin ánimo de lucro dedicado a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el manifiesto ágil, un documento que resume la filosofía ágil. Este manifiesto ágil comienza enumerando los principales valores del desarrollo ágil, como son:

- Al individuo y a las iteraciones del equipo de desarrollo sobre el proceso y las herramientas.
- Desarrollar software que funciona más que conseguir una buena documentación.
- La colaboración con el cliente más que la negociación de un contrato.
- Responder a los cambios más que seguir estrictamente un plan.

Los valores anteriores inspiran a los doce principios del manifiesto ágil, son características que diferencian un proceso ágil de uno tradicional. Los dos primeros son generales y resumen gran parte del espíritu ágil, estos son:

- I. La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- II. Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- III. Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible de entregas.

- IV. La gente del negocio y los desarrolladores deben de trabajar juntos a lo largo del proyecto.
- V. Construir el proyecto en tomo a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- VI. El dialogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- VII. El software que funcione es la medida principal de progreso.
- VIII. Los procesos agiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- IX. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- X. La simplicidad es esencial.
- XI. Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
- XII. En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivos y según esto ajusta a su comportamiento.

#### **2.8.2.2. PRINCIPALES METODOLOGÍAS ÁGILES**

Las metodologías agiles resuelven los problemas surgidos, posteriormente, a la masificación del uso del computador personal, dado que las expectativas y necesidades por parte de los usuarios se hicieron más urgentes y frecuentes.

Fue así como al comienzo de los 90 surgieron propuestas metodológicas para lograr resultados más rápidos en el desarrollo del software sin disminuir su calidad.

Entre las principales metodologías agiles se encuentran las siguientes:

➤ **PROGRAMACIÓN EXTREMA (XP).**- La programación extrema o Extreme Programing, es un enfoque de la ingeniería de software formulado por kent beck, se considera el más destacado de los procesos agiles de desarrollo de software.

Metodología de programación extrema XP. Sus fases son: Planificación, Diseño, Codificación, Pruebas, Lanzamiento.

➤ **SCRUM.-** Esta especialmente indicada para proyectos con un rápido cambio de requisitos. Sus características principales se pueden resumir en dos: El desarrollo del software se realiza mediante iteraciones denominadas sprint, con una duración de 30 días, el resultado de cada sprint es un incremento ejecutable que se muestra la cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. Sus fases son: Sprint Planning/Planeacion del sprint, Scrum team meeting/Reunión de equipo de scrum, Backlog refinement/Refinamiento del Backlog, Sprint review/Revisión del sprint, Retrospective/Retrospectiva del Sprint.

➤ **KANBAN.-** Es un método muy visual muy utilizado en la gestión ágil de proyectos. Muestra una imagen del proceso de trabajo, que permite ver posibles cuellos de botella en el desarrollo, que permite entregar un producto con calidad y a tiempo. Se basa en la visualización del flujo de trabajo y la limitación de la cantidad de trabajo en curso. Se utiliza para optimizar la eficiencia y la entrega continua de valor al cliente. Sus fases son: Establecer el flujo de trabajo, definir las fases de producción, añadir tareas, realizar un seguimiento adecuado, evaluación de procesos y áreas de mejora.

➤ **CRYSTAL METODOLOGÍAS.-** Se trata de un conjunto de metodologías para el desarrollo de software caracterizadas por encontrarse centradas en las personas que componen equipo y la reducción al máximo del número de artefactos producidos, familia de metodologías ágiles que se centran en la adaptabilidad y la flexibilidad. Cada metodología de cristal se adapta a un conjunto específico de circunstancias y requisitos, y se basa en un conjunto de principios y prácticas específicas. Sus fases son: Puesta en escena, revisiones, monitoreo, paralelismo y flujo, estrategia de diversidad holística, técnica de puesta a punto de la metodología, puntos de vista de usuario.



➤ **ADAPTIVE SOFTWARE DEVELOPMENT (ASD).**-Presupone que las necesidades del cliente son cambiantes. La iniciación de un proyecto involucra definir una misión para el, determinar las características, las fechas y descomponer el proyecto en una serie de pasos individuales, cada uno de los cuales puede abarcar entre cuatro u ocho semanas. Sus fases son: Especular, aprender, colaborar.

➤ **PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE.**-Es un conjunto de actividades necesarias para transformar los requisitos del usuario en un sistema software. Este proceso se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Sus fases son: Inicio, elaboración, construcción y transición.

➤ **DYMANIC SYSTEMS DEVELOPMENT METHOD.**- Es la única de las metodologías planteadas surgida de un consorcio formado originalmente por 17 miembros fundadores en enero de 1994. El objetivo de consorcio era producir una metodología de dominio público que fuera independiente de las herramientas y que pudiera ser utilizada en proyectos de tipo RAD (Rapid Application Development). Sus fases son: Estudio de factibilidad, estudio de negocio, iteración de modelo funcional, iteración del diseño y construcción, implantación. (CASTRO, 2016)

### 2.8.3. DIFERENCIAS ENTRE METODOLOGÍAS TRADICIONALES Y ÁGILES

**Tabla 2.1** Diferencia de metodologías.

METODOLOGÍAS ÁGILES	METODOLOGÍAS TRADICIONALES
Flexibilidad y capacidad de respuesta rápida ante modificaciones en el proyecto.	Rigidez y capacidad de respuesta lenta ante modificaciones en el proyecto.
Clientes más comprometidos e implicados en el desarrollo del proyecto.	Mínima interacción entre clientes y desarrolladores.

Pequeños equipos de trabajo (un promedio de 10 integrantes).	Grandes grupos de trabajo y con distintas distribuciones.
Tiempos de entregas más cortos.	Tiempos de entregas más extendidos.
Gran diversidad de roles en el equipo de desarrollo.	Mínima diversidad de roles en el equipo de desarrollo.
Pocas normas y políticas, procesos menos controlados.	Gran cantidad de normas y políticas, procesos muy controlados.
Garantía de que el producto cumpla las expectativas del cliente (evaluación continua).	No hay seguridad de que el producto cumpla todos los requisitos del cliente.
Posibilidad de respuesta ante cambios.	Obligado seguimiento estricto de lo planeado al comienzo del desarrollo.
Poca dependencia de la arquitectura de software.	Dependencia mediante modelos de la arquitectura software.

**Fuente:** Elaboración propia

#### **2.8.4. METODOLOGÍA DE DESARROLLO DE SOFTWARE XP (EXTREME PROGRAMING)**

Esta metodología posee un enfoque ágil, ya que enfatiza la comunicación cara a cara con el cliente o usuario en vez de la documentación. Además, el trabajo es realizado mediante la colaboración de equipos auto-organizados y multidisciplinarios, inmersos en un proceso compartido de toma de decisiones a corto plazo. En cada interacción del ciclo de vida incluye: planificación, análisis de requisitos, diseño, codificación, pruebas y documentación. Teniendo gran importancia el concepto de finalizado, ya que el objetivo de cada iteración no es agregar toda la funcionalidad para justificar el lanzamiento del producto al mercado, sino incrementar el valor por medio de software que funciona sin errores.

#### **2.8.4.1. CICLO DE VIDA DE LA METODOLOGÍA XP**

El ciclo de vida ideal de XP consiste en 6 fases: exploración, planificación de entrega, iteraciones, producción, mantenimiento y muerte del proyecto:

**1. Exploración:** El cliente plantea a grandes rasgos las historias de usuario son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto, ya que se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema, construyendo un prototipo. Esta fase toma pocas semanas o pocos meses, dependiendo del tamaño y familiaridad que tengas los programadores con la tecnología.

**2. Planificación de entrega:** El cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente, una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días.

**3. Iteraciones:** Esta fase incluye varias iteraciones sobre el sistema antes de ser entregado. El plan de entrega está compuesto por iteraciones de no más de tres semanas. En la primera iteración se puede intentar establecer una arquitectura del sistema que pueda ser utilizada durante el resto del proyecto. Esto se logra escogiendo las historias de usuario que fueren la creación de esta arquitectura, sin embargo, esto no siempre es posible ya que es el cliente quien decide que historias de usuario se implementaran en cada iteración. Al final de la última iteración el sistema estará listo para entrar en producción. Los elementos que deben tomarse en cuenta durante la elaboración del plan de la iteración son historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Será la tarea del cliente retroalimentar al equipo de desarrolladores

después de cada iteración con los problemas con los que se ha encontrado, mostrando sus prioridades, expresando sus sensaciones.

**4. Producción:** Requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase.

**5. Mantenimiento:** Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción. La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura.

**6. Muerte del proyecto:** Es cuando el cliente no tiene más historias para ser incluidas en el sistema. Esto indica que se satisfacen todas las necesidades del cliente en aspectos como rendimiento y fiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo. (Cesar Munder Calderin, 2017)

#### **2.8.4.2. LAS FASES DE LA METODOLOGÍA DE PROGRAMACIÓN EXTREMA**

Las fases de la metodología de Programación Extrema son:

##### **a) Planeación:**

➤ **Historias de usuario:** El paso de cualquier proyecto siga la metodología XP es definir las historias de usuario con el cliente. Las historias de usuario tienen la misma finalidad que los casos de uso, pero con algunas diferencias: Constan de 3 ó 4 líneas escritas por el cliente en un lenguaje no técnico sin hacer mucho hincapié en los detalles; no se debe hablar ni de posibles algoritmos para su

implementación ni de sus diseños de base de datos adecuados, etc. Son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. También se utilizan en la fase de pruebas, para verificar si el programa cumple con lo que especifica la historia del usuario. Cuando llega la hora de implementar una historia de usuario, el cliente y los desarrolladores se reúnen para concretar y detallar lo que tiene que hacer dicha historia. El tiempo de desarrollo ideal para una historia de usuario es entre 1 y 3 semanas.

➤ **Release Planning (Plan de entregas):** Después de tener ya definidas las historias de usuario es necesario crear un plan de publicaciones, en inglés “Release plan”, donde se indiquen las historias de usuario que se crearán para cada versión del programa y las fechas en las que se publicaran estas versiones. Un Release plan es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa. Después de reléase plan tiene que estar claros estos cuatro factores: los objetivos que se deben cumplir (que son principalmente las historias que se deben desarrollar en cada versión); el tiempo que tardarán en desarrollarse y publicarse las versiones del programa, el número de personas que trabajarán en el desarrollo y cómo se evaluará la calidad del trabajo realizado.

➤ **Iteraciones:** Todo proyecto que siga la metodología XP, se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el reléase planning que serán implementadas. También se seleccionan las historias de usuario que no pasaron al test de aceptación que se realizó al terminar la iteración anterior. Estas historias de usuario son divididas en tareas de entre 1 y 3 días de duración que se asignarán a los programadores.

➤ **La velocidad del proyecto:** Es una medida que representa la rapidez con la que se desarrolla el proyecto; estimarla es muy sencillo, basta con contar el

número de historias de usuario que se pueden implementar en una iteración; de esta forma, se sabrá el cupo de historias que se pueden desarrollar en las distintas iteraciones. Usando la velocidad del proyecto controlaremos que todas las tareas se puedan desarrollar en el tiempo del que dispone la iteración. Es conveniente reevaluar esta medida cada 3 ó 4 iteraciones y si se aprecia que no es adecuada hay que negociar con el cliente un nuevo plan de entregas.

➤ **Programación en parejas:** La metodología XP, aconseja la programación en parejas pues incrementa la productividad y la calidad del software desarrollado. El trabajo en parejas involucra a dos programadores trabajando en el mismo equipo; mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. De esta forma se consigue un código y diseño con gran calidad.

➤ **Reuniones diarias:** Es necesario que los desarrolladores se reúnan diariamente y expongan diariamente, soluciones e ideas de forma conjunta. Las reuniones tienen que ser fluidas y todo el mundo tiene que tener voz y voto.

#### **b) Diseño:**

➤ **Diseños simples:** La metodología XP sugiere que hay que conseguir diseños simples y sencillos. Hay que procurar hacerlo todo por lo menos complicado posible para conseguir un diseño fácilmente entendible e implementable que a la larga costará menos tiempo y esfuerzo desarrollar.

➤ **Glosario de términos:** Usar glosario de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

➤ **Riesgos:** Si surgen problemas potenciales durante el diseño, XP sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema.

➤ **Funcionabilidad extra:** Nunca se debe añadir funcionalidad extra al programa, aunque se piense que en un futuro será utilizada. Sólo el 10% de la misma es utilizada, lo que implica que el desarrollo de funcionalidad extra es un desperdicio de tiempo y recursos.

➤ **Refactorizar:** Refactorizar es mejorar y modificar la estructura y codificación de códigos ya creados sin alterar su funcionalidad. Refactorizar supone revisar de nuevo estos códigos para procurar optimizar su funcionamiento. Es muy común rehusar códigos ya creados que contienen funcionalidades que no serán usadas y diseños obsoletos.

**c) Codificación:** el cliente es una parte más del equipo de desarrollo; su presencia es indispensable en las distintas fases de XP. A la hora de codificar una historia de usuario su presencia es aún más necesaria. No olvidemos que los clientes son los que crean las historias de usuario y negocian los tiempos en los que serán implementadas. Antes del desarrollo de cada historia de usuario el cliente debe especificar detalladamente lo que ésta hará y también tendrá que estar presente cuando se realicen los test que verifiquen que la historia implementada cumple la funcionalidad especificada. La codificación debe hacerse atendiendo a estándares de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad.

**d) Pruebas:** Uno de los pilares de la metodología XP es el uso de test para comprobar el funcionamiento de los códigos que vayamos a implementando. El uso de los test en XP es el siguiente:

➤ Se deben crear las aplicaciones que realizarán los test con un entorno de desarrollo específico para test.

➤ Hay que someter a test las distintas clases del sistema omitiendo los métodos más triviales.

- Se deben crear los test que pasarán los Códigos antes de implementarlos; en el apartado anterior se explicó la importancia de crear antes los test que el código.
- Un punto importante es crear test que no tenga ninguna dependencia del código que en un futuro evaluará.
- Como se comentó anteriormente los distintos test se deben subir al repositorio de código acompañados del código que verifican.
- Test de aceptación, los test mencionados anteriormente sirven para evaluar las distintas tareas en las que ha sido dividida una historia de usuario.
- Al ser las distintas funcionalidades de nuestra aplicación no demasiado extensas, no se harán test que analicen partes de las mismas, sino que las pruebas se realizarán para las funcionalidades generales que deben cumplir el programa especificado en la descripción de requisitos. (Bustamante, 2014)



### 3. CAPITULO III PROPUESTA DE INOVACIÓN O SOLUCIÓN DEL PROBLEMA

#### 3.1. APLICACIÓN DE LA METODOLOGÍA XP (PROGRAMACIÓN EXTREMA)

En este acápite del proyecto se describe la aplicación de la metodología XP, cada fase describe el proceso el cual nos permitirá lograr nuestro objetivo, en la siguiente tabla se observa las fases y el proceso de la metodología XP.

**Tabla 3.1** Fases y procesos de la metodología XP.

FASES DE LA METODOLOGÍA XP	PROCESO DE LA METODOLOGÍA XP	PROCESO WebML
Planeación	➤ Historia de usuarios.	
	➤ Plan de entregas.	
	➤ Iteración.	
Diseño		➤ Modelo entidad relación.
		➤ Modelo de componentes.
Codificación	➤ Interfaz gráfico.	
Prueba	➤ Pruebas de aceptación.	

**Fuente:** Elaboración propia

##### 3.1.1. FASE 1: PLANEACIÓN

En la fase de la planeación se determinará los requerimientos funcionales del usuario los cuales serán primordiales para definir en funcionamiento del sistema.

### 3.1.1.1. ANALISIS DE REQUERIMIENTOS

Las descripciones de los requerimientos del usuario están descritas en la siguiente tabla:

**Tabla 3.2** Lista de requerimientos del usuario.

REFERENCIAS	REQUERIMIENTOS	PRIORIDAD
R1	Registro de personal	Alta
R2	Generar credencial	Alta
R3	Control general de registros de entradas y salidas	Alta
R4	Reportes por rango de fechas	Medio
R5	Usuario Administrador	Medio
R6	Personal de Seguridad	Medio

**Fuente:** Elaboración propia

### 3.1.1.2. HISTORIAS DE USUARIO

De acuerdo a los requerimientos del usuario procedemos a describir las historias de usuario, donde se describen las prioridades, riesgo en desarrollo, iteraciones y puntos estimados.

➤ **PRIORIDAD:** De acuerdo a la entrevista realizada al jefe de seguridad de NAABOL, se pudo identificar 2 grados de prioridad alta y media, para el desarrollo del sistema de control de entrada y salida del personal del aeropuerto, los cuales son tomados en cuenta en las historias de usuario.

➤ **RIESGO EN DESARROLLO:** De acuerdo al análisis realizado en la primera etapa realizada del proyecto puede existir riesgo al desarrollar de forma

inadecuada la solución de las historias de usuario, por lo que se consideran 2 grados de riesgo alto y medio.

➤ **ITERACION:** Este punto describe el número de iteraciones en el cual se espera implementar la historia del usuario, el tiempo promedio de entrega de iteración será aproximadamente de 2 semanas y al menos se tendrá 2 iteraciones para el desarrollo e implementación de las historias de usuario.

➤ **PUNTOS ESTIMADOS:** Los puntos estimados describe el tiempo promedio en semanas del desarrollo del sistema, para el cual se considera una escala de 1 a 6 semanas aproximadamente en los cuales se desarrollará el sistema.

### 3.1.1.3. DESCRIPCION DE HISTORIA DE USUARIOS

**REGISTRO DE PERSONAL:** Sistema de control de ingreso y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, permite registrar al personal que ingresa al aeropuerto, este procedimiento es realizado por el personal de seguridad del aeropuerto.

**Tabla 3.3 Historia de Usuario:** Registro de personal.

HISTORIA DE USUARIO	
<b>Numero: 1</b>	<b>Usuario: Administrador</b>
<b>Nombre de Historia:</b> Registro de Personal	
<b>Prioridad en Negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 1
Programador responsable: Juana Ludim Aduviri Salgado Reynaldo Huanca Morales	
<b>Descripción:</b> el registro del personal es indispensable para el funcionamiento del sistema este procedimiento lo realizara el administrador del sistema, para su registro se considera los siguientes datos: nombre, apellido paterno,	

---

apellido materno, fecha de nacimiento, número de documento, teléfono móvil, institución, placa de vehículo y estado.

---

**Observaciones:** los datos deben ser proporcionados del personal que trabaja en aeropuerto, los requisitos marcados con (\*) son obligatorios además esta operación solo lo puede realizar el administrador del sistema.

---

**Fuente:** Elaboración Propia

---

**IMPRESIÓN DE CREDENCIAL:** Sistema de control de ingreso y salida del personal del aeropuerto “Cp. Anibal Arab Fadul”, permite generar credenciales digitales con un código QR y datos necesarios para el personal del aeropuerto.

**Tabla 3.4 Historia de Usuario:** Impresión de credencial.

---

**HISTORIA DE USUARIO**

---

<b>Numero: 1</b>	<b>Usuario: Administrador</b>
------------------	-------------------------------

---

**Nombre de Historia:** Impresión de credencial

---

<b>Prioridad en Negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
-----------------------------------	-----------------------------------

---

<b>Puntos estimados:</b> 8	<b>Iteración asignada:</b> 1
----------------------------	------------------------------

---

Programador responsable: Juana Ludim Aduviri Salgado

Reynaldo Huanca Morales

---

**Descripción:** la impresión de credencial para personal es indispensable para el funcionamiento del sistema este procedimiento lo realizara el administrador del sistema, para su funcionamiento se considera código de cada persona que trabaja en el aeropuerto.

---

**Observaciones:** El código del personal es su número de documento de cada trabajador (Cedula de identidad).

---

**Fuente:** Elaboración Propia

---

**CONTROL GENERAL:** Sistema de control de ingreso y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, tiene una lista de control general del personal donde muestra que personal ingreso o salió con fecha y hora actualizada.

**Tabla 3.5 Historia de Usuario:** Control general de registros de entrada y salidas.

HISTORIA DE USUARIO	
<b>Numero: 1</b>	<b>Usuario: Administrador</b>
<b>Nombre de Historia:</b> Control general de registros de entrada y salida	
<b>Prioridad en Negocio:</b> Alta	<b>Riesgo en desarrollo:</b> Alto
<b>Puntos estimados:</b> 8	<b>Iteración asignada:</b> 1
Programador responsable: Juana Ludim Aduviri Salgado Reynaldo Huanca Morales	
<b>Descripción:</b> El sistema muestra un lista del personal que ingrese o salga del aeropuerto con fecha y la hora.	
<b>Observaciones:</b> para registrar al personal en esta lista se debe escanear su credencial digital.	
<b>Fuente:</b> Elaboración Propia	

**REPORTES:** Sistema de control de ingreso y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, tiene una búsqueda por rango de fecha y genera una lista del todo el personal de entrada y salida en la fecha seleccionada.

**Tabla 3.6 Historia de Usuario:** Reportes por rango de fechas.

HISTORIA DE USUARIO	
<b>Numero: 1</b>	<b>Usuario: Administrador</b>
<b>Nombre de Historia:</b> Reportes por rango de fechas	

<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 5	<b>Iteración asignada:</b> 1
Programador responsable: Juana Ludim Aduviri Salgado Reynaldo Huanca Morales	
<b>Descripción:</b> El sistema permite buscar por rango de fechas, donde muestra y genera una lista del todo el que salió y entro en fecha seleccionada.	
<b>Observaciones:</b> En el reporte de rango de fechas debe ingresar de que fecha hasta que fecha quiere ver o generar un reporte.	
<b>Fuente:</b> Elaboración Propia	
<b>USUARIO ADMINISTRADOR:</b> Sistema de control de entrada y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, permite asignar roles al personal, en este sistema está considerado dos roles, administrador y personal de seguridad.	
<b>Tabla 3.7 Historia de Usuario:</b> Usuario Administrador.	
<b>HISTORIA DE USUARIO</b>	
<b>Numero:</b> 1	<b>Usuario:</b> Administrador
<b>Nombre de Historia:</b> Usuario Administrador	
<b>Prioridad en Negocio:</b> Medio	<b>Riesgo en desarrollo:</b> Medio
<b>Puntos estimados:</b> 3	<b>Iteración asignada:</b> 1
Programador responsable: Juana Ludim Aduviri Salgado Reynaldo Huanca Morales	
<b>Descripción:</b> Permite asignar usuario, contraseña y rol al personal , en este sistema está considerado dos roles, administrador y personal de seguridad.	

---

**Observaciones:** Para asignar roles al personal tanto como administrador o seguridad debe realizar nuevo registro.

---

**Fuente:** Elaboración Propia

**PERSONAL DE SEGURIDAD:** Sistema de control de ingreso y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, tiene el rol importante que es personal de seguridad que realizara el control minucioso del sistema.

**Tabla 3.8 Historia de Usuario:** Personal de Seguridad.

HISTORIA DE USUARIO	
Numero: 1	Usuario: Administrador
Nombre de Historia: Personal de Seguridad	
Prioridad en Negocio: Medio	Riesgo en desarrollo: Medio
Puntos estimados: 3	Iteración asignada: 1
Programador responsable: Juana Ludim Aduviri Salgado Reynaldo Huanca Morales	
Descripción: El sistema tiene para registrar personal nuevo de la seguridad y asignarle el rol de seguridad.	
Observaciones: Al registra nuevo personal de seguridad permite asignar el rol.	

---

**Fuente:** Elaboración Propia

#### 3.1.1.4. PLAN DE ENTREGAS

En esta etapa de la fase de la planificación se describe el plan de entregas donde se detalla las historias de usuario y el tiempo estimado para el desarrollo de cada historia, además se describe las iteraciones correspondientes de cada historia de usuario, esto nos permitirá contar con un tiempo aproximado de desarrollo de todo el sistema.

**Tabla 3.9** Plan de entregas por historias de usuario.

N°	Historia de usuario	Tiempo estimado			Iteración								
		S.	D.	H.	1	2	3	4	5	6	7	8	9
1	Registro de personal	4	20	60	x								
2	Generar credencial	6	30	30		x							
3	Control general de registros de entradas y salidas	2	10	30			x						
4	Reportes por rango de fechas	2	10	30			x						
5	Usuario Administrador	3	15	45	x								
6	Personal de Seguridad	4	20	60				x					

**Fuente:** Elaboración Propia

### 3.1.1.5. ITERACIONES

➤ **Primera iteración:** Para el desarrollo del proyecto consideramos las historias de usuario más importantes, para primera iteración consideramos las siguientes historias:

**H1:** USUARIO ADMINISTRADOR

**H2:** REGISTRO DE PERSONAL

#### TAREAS DE LA HISTORIA 1: USUARIO ADMINISTRADOR

Para concretar este procedimiento desarrollamos un conjunto de tareas, para lo cual se desarrolla el diseño de formulario de la historia 1, el cual permitirá datos del usuario administrador los cuales son indispensables para el ingreso al sistema.



**Tabla 3.10** TAREA 1: Diseño de formulario de registro de personal.

Tarea N°1	
Número de tarea: 1	Número de historia: 1
Nombre de la tarea: Registro del personal.	
Tipo de tarea: Desarrollo	Puntos estimados: 4
Fecha de inicio: 01-03-2023	Fecha fin: 20-03-2023
Descripción: Se desarrollará un formulario, el cual permita introducir a las personas que trabajan en el aeropuerto “Cap. Anibal Arab Fadul”.	
<b>Fuente:</b> Elaboración Propia	

**Tabla 3.11** TAREA 02: Diseño de la generación de la credencial para el personal.

Tarea N° 2	
Número de tarea: 2	Número de historia: 1
Nombre de la tarea: Generar credencial del personal.	
Tipo de tarea: Desarrollo	Puntos estimados: 6
Fecha de inicio: 01-05-2023	Fecha fin: 30-05-2023
Descripción: Se diseñará un credencial con lectura de tecnología código de QR para el control del registro de ingreso y salida del aeropuerto “Cap. Anibal Arab Fadul”.	
<b>Fuente:</b> Elaboración Propia	

**Tabla 3.12** TAREA 03: Diseño de control del personal del aeropuerto.

Tarea N° 3	
Número de tarea: 3	Número de historia: 1
Nombre de la tarea: Control general del personal.	

Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha de inicio: 01-06-2023	Fecha fin: 10-06-2023
Descripción: Se diseñará un formulario que permita controlar el registro de ingreso y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”.	
<b>Fuente:</b> Elaboración propia	

**Tabla 3.13** TAREA 04: Diseño del formulario de acceso a personal de seguridad.

Tarea N°4	
Número de tarea: 4	Número de historia: 1
Nombre de la tarea: Acceso a personal de seguridad.	
Tipo de tarea: Desarrollo	Puntos estimados: 4
Fecha de inicio: 01-03-2023	Fecha fin: 20-03-2023
Descripción: Se desarrollará un formulario, el cual permita introducir a la persona que será el operador para el control del registro del aeropuerto “Cap. Anibal Arab Fadul”.	
<b>Fuente:</b> Elaboración Propia	

**Tabla 3.14** TAREA 5: Diseño de usuario y contraseña para el personal de seguridad.

Tarea N° 5	
Número de tarea: 5	Número de historia: 1
Nombre de la tarea: Usuario y contraseña para el personal de seguridad.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha de inicio: 01-03-2023	Fecha fin: 20-03-2023

---

Descripción: Se creará un usuario y contraseña al personal de seguridad que operará el sistema de control de registro de ingreso y salida del personal de aeropuerto “Cap. Anibal Arab Fadul”.

---

**Fuente:** Elaboración Propia

## **TAREAS DE LA HISTORIA 2: REGISTRO DE PERSONAL**

Para concretar este procedimiento desarrollamos un conjunto de tareas, para lo cual se desarrolla el diseño de formulario de la historia 2, el cual permitirá datos de los usuarios que trabajan en el aeropuerto es parte indispensable para el desarrollo del sistema.

**Tabla 3.15** TAREA 1: Diseño del formulario de datos del personal.

Tarea N° 1	
Número de tarea: 1	Número de historia: 2
Nombre de la tarea: Formulario de datos del personal.	
Tipo de tarea: Desarrollo	Puntos estimados: 2
Fecha de inicio: 01-08-2023	Fecha fin:20-08-2023
Descripción: Se desarrollará un formulario de registro de datos del personal que trabaja en el aeropuerto, el cual permita introducir los datos requeridos del personal y verificar si la operación se realizó con éxito.	

**Fuente:** Elaboración Propia

**Tabla 3.16** TAREA 2: Modificar del formulario los datos del personal.

Tarea N° 2	
Número de tarea: 2	Número de historia: 2
Nombre de la tarea: Modificar	
Tipo de tarea: Desarrollo	Puntos estimados: 2

Fecha de inicio: 01-08-2023

Fecha fin:20-08-2023

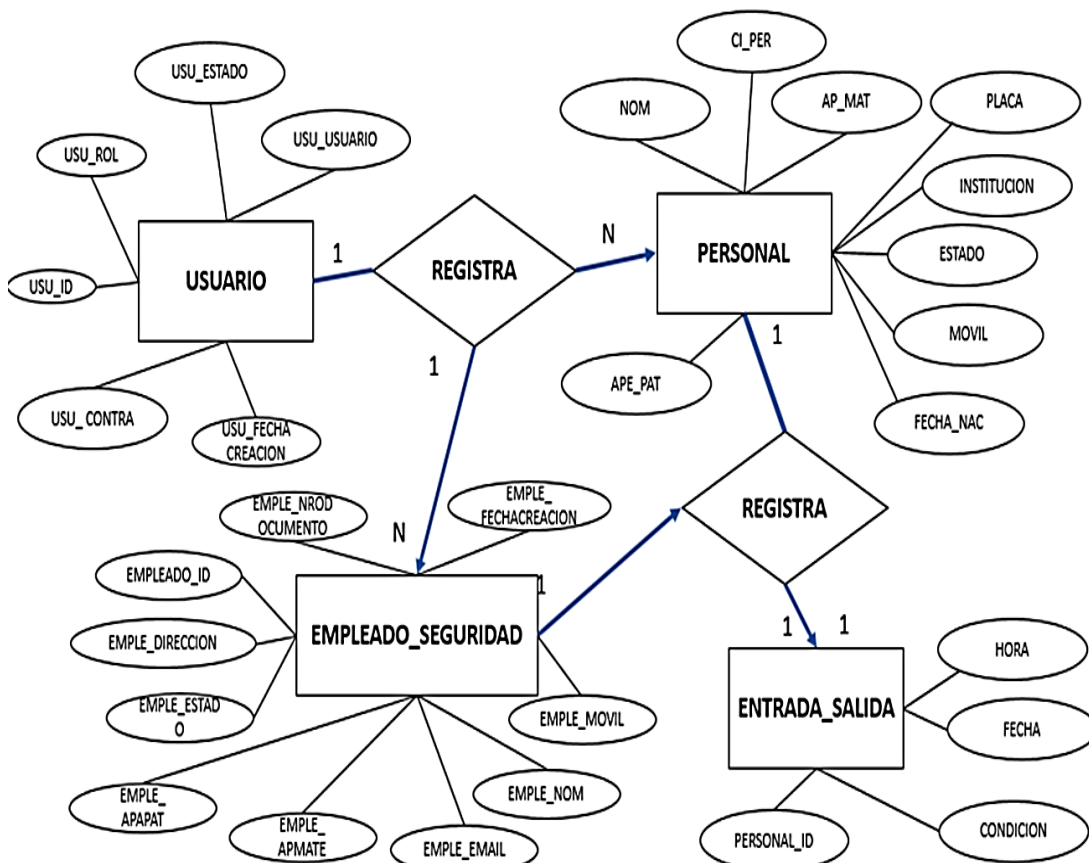
Descripción: Se desarrollará un formulario, el cual permita realizar cambios o modificaciones en los datos según requerimiento del usuario para personal que trabaja en el aeropuerto, y verificar si la operación se realizó con éxito.

**Fuente:** Elaboración Propia

### 3.1.2. FASE 2: DISEÑO

#### ➤ MODELO ENTIDAD RELACION:

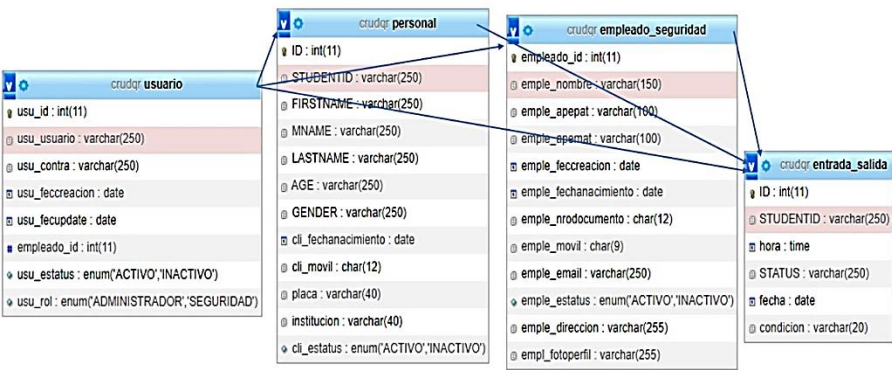
**Figura 3.1** Modelo entidad relación.



**Fuente:** Elaboración Propia

➤ **MODELO DE ESTRUCTURA:** El modelo de estructura representa las tablas de la base de datos relacionados.

**Figura 3.2** *Modelo de estructura.*



**Fuente:** Elaboración Propia

➤ **TARJETA CRC (CLASE-RESPONSABILIDAD-COLABORACION):** La función de estas tarjetas es ayudar a dejar el pensamiento procedimental para incorporarse el enfoque orientado a objetos. Estas tarjetas se dividen de dividen en tres secciones que contiene la información del nombre de la clase, sus responsabilidades y sus colaboradores.

**Tabla 3.17** TARJETA CRC: USUARIO.

TARJETA CRC	
NOMBRE DE LA CLASE: Usuario	
DESCRIPCION: Esta clase contiene los datos con relación al usuario	
RESPONSABILIDADES	COLABORADORES
➤Crear nuevo empleado de seguridad.	➤Personas
➤Crear nuevo registro.	
➤Actualizar datos.	

**Fuente:** Elaboración Propia

**Tabla 3.18** TARJETA CRC: EMPLEADO SEGURIDAD.

TARJETA CRC	
NOMBRE DE LA CLASE: Empleado Seguridad	
DESCRIPCION: Esta clase contiene los datos con relación al empleado de seguridad.	
RESPONSABILIDADES	COLABORADORES
<ul style="list-style-type: none"><li>➤Registrar la entrada de las personas.</li><li>➤Registrar la salida de las personas.</li><li>➤Actualizar datos.</li></ul>	<ul style="list-style-type: none"><li>➤Personas</li></ul>

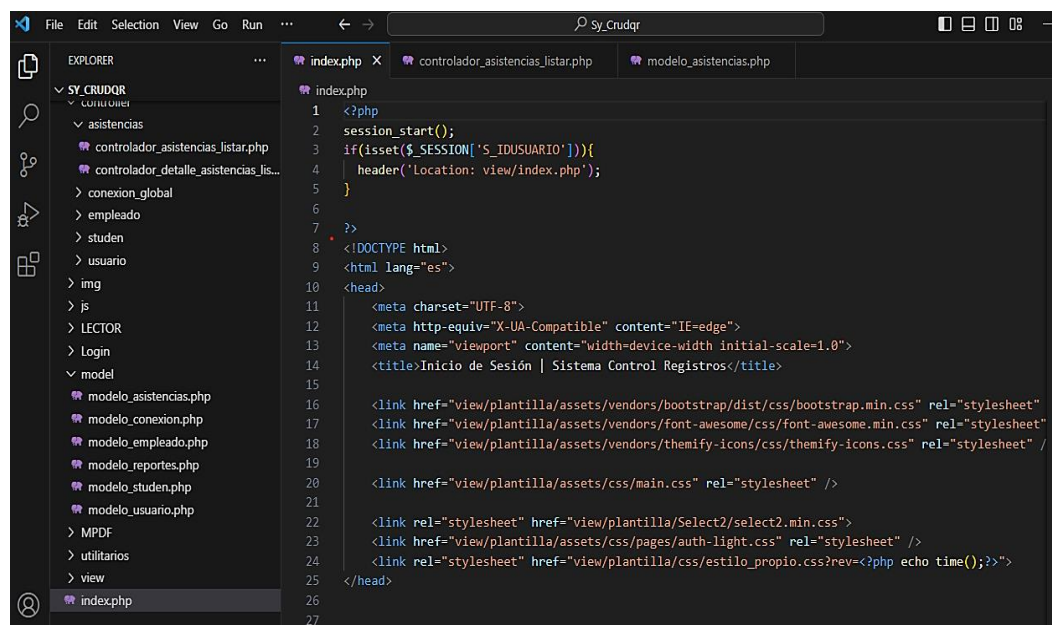
**Fuente:** Elaboración Propia

**3.1.3. FASE 3: CODIFICACIÓN**

Para el desarrollo del sistema utilizamos el lenguaje de programación php, javscriptp, y herramientas como html, además gestor de base de datos; MySQL.

➤ **Código fuente:**

**Figura 3.3** *Editor de código fuente.*



**Fuente:** Elaboración Propia

➤ **Interfaz gráfico:**

**Figura 3.4** *Interfaz.*



**Fuente:** Elaboración Propia

➤ **Fase de prueba:**

**Figura 3.5** *Iteraciones de prueba.*

#	N°Documento	Nombre Completo	N°Celular	Placa	institución	Estado	acción
1	9129638	REYNALDO HUANCA MORALES	71712058	BAT-2079	FELCN	ACTIVO	<a href="#">[icon]</a>
2	22147475	JOSELITO HUANCA MORALES	45457474	FFFF-44	NAABOL	ACTIVO	<a href="#">[icon]</a>
3	15478694	GUALBERTO ROQUE ROJAS	75930838	SIN VEHICULO	GRUPO AÉREO	ACTIVO	<a href="#">[icon]</a>
4	66666666	GUSTAVO ESOINOZA LITORAL	71712058	SIN VEHICULO	BOA	ACTIVO	<a href="#">[icon]</a>
5	72736222	JUAN MILLAS GONZALES	736125521	SIN VEHICULO	FELCN	ACTIVO	<a href="#">[icon]</a>

Registros del (1 al 5) total de 5 registros 0 fila seleccionada

Anterior 1 Siguiente

**Fuente:** Elaboración Propia

## RESULTADOS ESPERADOS

- Se realizó un análisis exhaustivo de los requisitos actuales del proceso de registro de entrada y salida del personal en el aeropuerto “Cap. Anibal Arab Fadul”. Esta evaluación detallada permitió identificar claramente las deficiencias y demoras existentes, la recopilación de esta información sirvió de base sólida para la implementación de soluciones eficaces.
- Se obtuvo un diseño completo y funcional de los módulos del sistema para el registro automatizado, destacando la implementación efectiva de la tecnología QR con el propósito de agilizar significativamente el proceso de control de entrada y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, mejorando la eficiencia y precisión de la recopilación de datos, optimizando la gestión de información en el proceso de registro.
- La implementación y funcionalidad del módulo de integración del código QR en los diferentes módulos del sistema del registro automatizado, se logró una integración fluida y eficiente que permite la lectura y utilización efectiva de los códigos QR para agilizar la entrada y salida del personal del aeropuerto “Cap. Anibal Arab Fadul”, facilitando la sincronización y actualización precisa de datos entre los distintos módulos.
- Reporte de informes, con el sistema automatizado del proceso de control de ingreso y salida basado en códigos QR, se pueden generar reportes y análisis detallados sobre el movimiento del personal, este sistema proporciona una visión completa y precisa de las actividades de entrada y salida, permitiendo un análisis exhaustivo de los datos recopilados.



## CONCLUSIONES

Con el desarrollo de este sistema se pudo llegar a conclusiones importantes:

- La automatización del proceso de registro de entrada y salida del personal en el aeropuerto “Cap. Anibal Arab Fadul” ha demostrado ser un logro fundamental, contribuyendo de manera sustancial a mejorar la eficiencia operativa y fortalecer la seguridad en esta importante institución, a través del desarrollo de un sistema de registro automatizado basado en tecnología de códigos QR, que ha sido central en este avance, proporcionando una solución eficaz y moderna.
- La reducción sustancial en los tiempos de registro, la mejora en la precisión de los registros y la facilitación del control de seguridad son resultados tangibles que demuestran el impacto positivo de este proyecto.
- Este proyecto no solo ha logrado mejoras operativas concretas, sino que también ha sentado un precedente importante para la adopción de tecnologías modernas en el aeropuerto “Cap. Anibal Arab Fadul”.

## RECOMENDACIONES

El presente sistema de control de entrada y salida del personal que trabaja en el aeropuerto “Cap. Anibal Arab Fadul”, tiene las siguientes recomendaciones:

- Mantenimiento preventivo continuo, se recomienda establecer un programa de mantenimiento preventivo regular para el sistema de registro automatizado. Esto garantizará su rendimiento óptimo a lo largo del tiempo y reducirá el riesgo de fallas inesperadas.
- Actualizaciones tecnológicas, establecer un plan para mantener el sistema actualizado con las últimas tecnologías y estándares de seguridad. Esto ayudara a garantizar que el sistema siga siendo efectivo y seguro a medida que evolucionen las tecnologías.
- Capacitar al personal del aeropuerto “Cap. Anibal Arab Fadul”, sobre el funcionamiento y características del sistema de control de entrada y salida del personal, para dar una buena utilidad del sistema.
- Monitoreo de usuario y contraseñas, establecer personal capacitado para el uso del sistema con claves de seguridad personales.

## FUENTES DE INFORMACIÓN Y BIBLIOGRAFÍA

Aguilar, L. J. (2008). *Fundamentos de Programacion*. España: McGRAW-HILL/INTERAMERICANA DE ESPAÑA. S.A.U.

Angel Cobo, P. G. (2005). *PHP y MySQL Tecnologías para el desarrollo de aplicaciones web*. España: Diaz de Santos.

Bustamante, D. (2014). *Metodología actual metodología XP*. Barinas.

CASTRO, M. A. (2016). *Metodologia agil de desarrollo de software programacion extrema*. Nicaragua.

Cesar Munder Calderin, Y. B. (2017). *Metodologias de desarrollo de software (XP)*. La Habana.

Cevallos, k. (2015). *Metodologia de desarrollo agil: XP y Scrum*.

DGAC, R. 1. (2017). *dgac.gob.bo*. Obtenido de *dgac.gob.bo*.

INNOVA. (2018). *Evolucion de las metodologias y modelos utilizados en el desarrollo de software*. Guayaquil.

investigalia. (s.f.). <https://investigaliacr.com/investigacion/el-enfoque-cualitativo-de-investigacion/>. Obtenido de <https://investigaliacr.com/investigacion/el-enfoque-cualitativo-de-investigacion/>.

Luis Alberto Casillas Santillan, M. G. (s.f.). *Bases de datos en MySQL*.

Maida, E. G. (2015). *METODOLOGIAS DE DESARROLLO DE SOFTWARE*. ARGENTINA: UCA.

Perez, J. E. (2009). *Introduccion a JavaScript*. española.

SYDLE. (s.f.). <https://www.sydle.com/es/blog/extreme-programming-602ee205da4d096809438c9c>. Obtenido de <https://www.sydle.com/es/blog/extreme-programming-602ee205da4d096809438c9c>.

Unir. (2023). *Unir la universidad en internet*. Obtenido de Unir la universidad en internet: <https://mexico.unir.net/ingenieria/noticias/gestores-de-base-de-datos/>

Wikipedia. (s.f.). *Wikipedia enciclopedia libre*. Obtenido de Wikipedia enciclopedia libre:

[https://es.wikipedia.org/wiki/Sistema\\_de\\_informaci%C3%B3n#:~:text=En%20inform%C3%A1tica%2C%20un%20sistema%20de,satisfacer%20una%20necesidad%20de%20informaci%C3%B3n](https://es.wikipedia.org/wiki/Sistema_de_informaci%C3%B3n#:~:text=En%20inform%C3%A1tica%2C%20un%20sistema%20de,satisfacer%20una%20necesidad%20de%20informaci%C3%B3n).

Wikipedia. (s.f.). *Wikipedia, enciclopedia libre*. Obtenido de Wikipedia, enciclopedia libre: [https://es.wikipedia.org/wiki/C%C3%B3digo\\_QR](https://es.wikipedia.org/wiki/C%C3%B3digo_QR)

**ANEXOS**

## ANEXO 1 : Observación realizada



Fuente: Elaboracion Propia

Imagen : planilla de registro

NOMBRE Y APELLIDO		CEDULA DE IDENTIDAD	VEHICULO TIPO	PLACA	HORA	ING.	SAL.	Nº	PASE	AREA DE INGRESO	INSTITUCION O PROCEDENCIA	FECHA	FUNCIONARIO DE REGISTRO
Yadira Miranda		7571461	Auto		18:20					ADUANAS	ADUANAS	13/05/23	Guatemala
Edoardo Vargas		4571353	Auto		18:40					ADUANAS	ADUANAS	13/05/23	Guatemala
David Rojas		7804178	Auto		18:50					ADUANAS	ADUANAS	13/05/23	Guatemala
Sergio Nolasco Rojas		7203750	Auto		19:10					SOFA	SOFA	13/05/23	Guatemala
Yusni Kravynsky		702100	Auto		19:10					SOFA	SOFA	13/05/23	Guatemala
Guillermo		1781056	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Roberto Guadalupe		1020276	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Admiral Diez Cruz		4427124	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Raul Vaca		4427124	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Manuel Chacon		4321772	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Wladimir Torres		4715005	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Ruddy Samuel Gomez		4554402	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Ramiro Sanga Herry		1020276	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Hector Soto		2152203	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Roberto Rojas		2340042	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Jose Maria Jimenez		6846280	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Sandra Blanca		7253238	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Virginia Garcia		6197727	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Yolanda Reyes		4427124	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Roberto Ramirez		5067211	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Roberto Berlich		1575553	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
Victor Velaz		4973283	Auto		19:20					SOFA	SOFA	13/05/23	Guatemala
		1951146								SOFA	SOFA	13/05/23	Guatemala
		A216362								SOFA	SOFA	13/05/23	Guatemala

Fuente: NAABOL

## ANEXO 2 : Guía de cuestionario

### CUESTIONARIO

Al Señor.....

Cargo.....

Fecha.....

1.- ¿Cuál es la estructura organizacional de la institución de NAABOL?

R.-

2.- ¿Cuál es la función de NAABOL?

3.- ¿Quién es el encargado o responsable de la seguridad del perímetro del aeropuerto?

4.- ¿Cuál es el procedimiento actual para controlar la entrada y salida del personal del aeropuerto?

5.- ¿Qué medidas de seguridad se implementan para verificar la identidad del personal antes de permitirles acceder a áreas restringidas?

6.- ¿Con que fin se desarrolla este procedimiento?

7.- ¿Qué tipo de identificación se requiere para el control de entrada y salida del personal?

8.- ¿Cuál es el proceso de registro de entrada y salida del personal a su lugar de trabajo dentro del aeropuerto?

9.- ¿Se utiliza alguna tecnología de verificación biométrica, como huellas dactilares o reconocimiento facial, para el control de entrada y salida del personal?

10.- ¿Cuál es la frecuencia de los controles de seguridad realizados al personal durante el día?

11.- ¿Qué datos se registran del personal?

12.- ¿Cuánto tiempo tarda en registrar al personal?

13.- ¿Cuántas instituciones hay al interior del aeropuerto que se tenga conocimiento?

14.- ¿Al momento de registrar cuantos errores son los que se cometen?

15.- ¿Qué dificultados se tiene al momento de registrar?

16.- ¿Con que fin se realiza el control del personal del aeropuerto?

17.- ¿Cómo se manejan las situaciones en las que se detecta una entrada o salida no autorizada de personal en el aeropuerto?

### CUESTIONARIO

Al Señor *Humberto Alvarado*.....

Cargo *Coordinador ASIS en NAABOL*.....

Fecha *27-07-2025*.....

1.- ¿Cuál es la estructura organizacional de la institución de NAABOL?

R.- Director ejecutivo, Directores regionales, JEFES de Aeropuerto, subvotos ASIS, JEFES de Infraestructura, Bomberos SEI, y Oficinas de seguridad.

2.- ¿Cuál es la función de NAABOL?

R.- El Control de la navegación aérea en Bolivia y del tránsito de aeronaves en el espacio aéreo Boliviano.

3.- ¿Quién es el encargado o responsable de la seguridad del perímetro del aeropuerto?

R.- El encargado y responsable es NAABOL conjuntamente con organismo de Seguridad del Estado FABS CDIRESA.

4.- ¿Cuál es el procedimiento actual para controlar la entrada y salida del personal del aeropuerto?

R.- La inspección y registro manual de pasajeros y personas que de alguna manera cumple una función operativa en el aeropuerto.

5.- ¿Qué medidas de seguridad se implementan para verificar la identidad del personal antes de permitirles acceder a áreas restringidas?

R.- Si es personal que trabaja en el aeropuerto se pide la credencial otorgada por el Administrador del aeropuerto y control de identidad otorgado por el personal de la institución.

6.- ¿Con que fin se desarrolla este procedimiento?

R.- Con el fin de identificar que sea la persona correcta que presenta la credencial o identificación personal.

7.- ¿Qué tipo de identificación se requiere para el control de entrada y salida del personal?

R.- Credencial o carnet de identidad.

8.- ¿Cuál es el proceso de registro de entrada y salida del personal a su lugar de trabajo dentro del aeropuerto?

R.- El registro es manual de entrada y salida.

9.- ¿Se utiliza alguna tecnología de verificación biométrica, como huellas dactilares o reconocimiento facial, para el control de entrada y salida del personal?

R.- No se utiliza ninguna tecnología digital o biométrica.

10.- ¿Cuál es la frecuencia de los controles de seguridad realizados al personal durante el día?

R.- El Control y Prevención es todos los días de 08:00 a 16:00 pm.

11.- ¿Qué datos se registran del personal?

R.- Nombre completo, C.I., institución a la que pertenece.

12.- ¿Cuánto tiempo tarda en registrar al personal?

R.- Como 2 minutos máximo.

13.- ¿Cuántas instituciones hay al interior del aeropuerto que se tenga conocimiento?

NAABOL, DGC, Adana, Migración, BOA, YPFB, ZOFRA, Defensoría del Niño, FELIC, FABS, DREDA, Policía, Concisionarios y personal del aeropuerto.

14.- ¿Al momento de registrar cuantos errores son los que se cometen?

R.- Al Anotar el nombre y carnet de identificación.

15.- ¿Qué dificultados se tiene al momento de registrar?

R.- Hay veces que se cometen con un lapicero o por el tiempo.

16.- ¿Con que fin se realiza el control del personal del aeropuerto?

R.- Con el objetivo de identificar a personas que quieren cometer un acto de interferencia ilícita.

17.- ¿Cómo se manejan las situaciones en las que se detecta una entrada o salida no autorizada de personal en el aeropuerto?

R.- Ante la verificación de registro en los formularios de producción en el área que corresponda pero es muy dificultoso o le hace reducir la verificación del mismo.

Fuente: Elaboración propia

**ANEXO 3:** Cuestionario llenado por el personal de seguridad de NAABOL



**Fuente:** Elaboración propia