



UNIVERSIDAD DE PAMPLONA



**Control de préstamo de equipos del laboratorio de
Electrónica Digital ubicado en el edificio Enrique
Rochereaux de la Universidad de Pamplona
mediante identificación biométrica**

Autor:

Rolando Ramiro Villamizar Pabón

Director:

José Daniel Ramírez Corzo

**Ingeniería Electrónica
Departamento de Ingenierías Eléctrica, Electrónica, Sistemas y
Telecomunicaciones
Facultad de Ingenierías y Arquitectura
Universidad de Pamplona
Pamplona**

Índice

Prefacio.....	V
Agradecimientos.....	VII
Resumen	VIII
1. Introducción.....	2
1.1. Problema	2
1.2. Justificación.....	3
1.3. Objetivos	3
1.4. Distribución Bibliográfica:	4
2. Marco Teórico.....	6
2.1. Base de datos	6
2.1.1. Los Datos	7
2.2. Modelado de datos y modelo de datos	8
2.2.1. Concepto de modelo	8
2.2.2. Elementos básicos de un Modelo de Datos.....	9
2.2.3. Modelos lógicos basados en objetos	10
2.3. Relaciones y variables de relación.....	10
2.3.1. Diseño de bases de datos con el Modelo E/R	11
2.3.2. Presentación de MySQL	11
2.4. El Lenguaje de Programación Java	12
2.4.1. Objetivos del diseño de Java	12
2.4.3. Securizado	13
2.5. Huella dactilar.....	14
2.5.1. Precursores en la Biometría.....	14
2.5.2. Adquisición de Huellas dactilares	14
2.5.3. La Huella dactilar en su estructura como método de identificación	15
2.5.4. Reconocimiento por medios de los puntos característicos de Hullas dactilares	16
2.5.4.1. Extracción de Vectores de Características o patrones.....	16
3. Metodología, estructura y programación.....	21

3.2.	Estructura Base de Datos	24
3.2.1.	Identificación Entidades y relaciones	26
3.2.2.	Modelo Entidad Relación (MER).....	28
3.2.3.	Creación base de datos en Xampp	30
3.3.	Lector de Huellas Digital Perna U.are.u 4500	31
3.3.1.	Configuración y programación de Digital Personal U.are.U 4500 USB Biometric Fingerprint Scanner Fingerprint Reader	32
3.3.2.	Creación de un Objeto	33
3.3.3.	Aplicación del lector de Huellas	33
4.	Aplicativo en Java.....	36
4.1.	Características del sistema	36
4.2.	Rol –administrador-.....	36
4.2.1.	Archivo	38
4.2.2.	Funcionalidades	39
4.2.2.1.	Estudiantes.....	39
4.2.3.	Prestamos	42
4.2.3.1.	Inicio Sesión	42
4.2.3.2.	Proceso de solicitud	42
4.2.4.	Consultas	44
4.2.4.1.	Estudiantes.....	44
4.2.4.3.	Solicitudes	45
4.3.	Rol –Estudiante-.....	46
4.4.	Pruebas del aplicativo	47
4.5.	Errores detectados y solución	49
5.	Conclusiones	54
6.	Recomendaciones.....	55
7.	Bibliografía.....	56

Lista de figuras

Figura 1. Imagen simplificada de un sistema de base de datos.....	7
Figura 2. Huella dactilar, ilustrando los patrones de una Huella.....	15
Figura 3. Tipos Patrones de identificación.....	17
Figura 4. Tratamiento de imagen para extracción de patrones.....	18
Figura 5. Diagrama de flujo Solicitud de préstamo.....	23
Figura 6. formato FLA-03 v.04.....	25
Figura 7. Diagrama relación de dos entidades.....	28
Figura 8. Modelo de Identidad relación.....	29
Figura 9. Pantalla de Inicio XAMPP.....	30
Figura 10. Pantalla de bienvenido phpmyadmin.....	30
Figura 11. Creación de una tabla en phpmyadmin.....	31
Figura 12. Captura Pantalla, Ubicación de librerías para Digital Persona.....	32
Figura 13. Captura de pantalla, conexión base de datos con XAMPP.....	34
Figura 14. Ventana de inicio sección rol-administrativo.....	37
Figura 15. Funciones Archivo.....	38
Figura 16. Registro de Estudiante.....	39
Figura 17. Registro de Huella.....	40
Figura 18. Registro de un nuevo equipo.....	41
Figura 19. Registro tipo de Equipo.....	41
Figura 20. Solicitud de préstamo Rol –administrador-.....	42
Figura 21. Realizar una solicitud rol –administrador-.....	43
Figura 22. Adición de un equipo solicitud rol –administrador-.....	43
Figura 23. Estudiantes Inscritos en el sistema de préstamo.....	44
Figura 24. Consulta de equipos.....	45
Figura 25. Consulta y control de solicitudes rol-administrador-.....	45
Figura 26. Inicio de sesión para préstamos rol-estudiante-.....	46
Figura 27. Ventana de solicitud de equipos rol-estudiante-.....	47
Figura 28. a. Registro de un nuevo tipo equipo b. Registro de equipo c. Registro de usuario d. Solicitud de préstamo rol –estudiante- e. Solicitud de préstamo rol –administrador- f. Consulta de solicitudes realizadas g. Consulta de equipos luego de aprobar solicitudes h. Consulta de equipos luego de recibir las solicitudes.....	48

Lista de tablas

Tabla 1. Relación de propiedades de identidad con variables creadas en Mysql.....	27
Tabla 2. Descripción de relaciones del MER	29
Tabla 3. Errores destacados y solución	50

Prefacio

Prefacio	V
Agradecimientos.....	VII
Resumen.....	VIII

“Nuestras virtudes y nuestros defectos son inseparables, como la fuerza y la materia. Cuando se separan, el hombre deja de existir”.

Nikola Tesla

Agradecimientos

Un agradecimiento eterno a Dios, quien mi vida ha puesto las personas correctas y quienes me han apoyado.

A mis padres quienes, con su labor diaria, me inculcaron valores, para ser una persona de bien y dedicada a mis sueños, y me han dado un apoyo incondicional en mi crecimiento continuo.

También mi agradecimiento para mi hermana que siempre ha estado ayudándome y apoyándome en mis decisiones, y que sin importar la situación ha estado a mi lado para darme la mano.

A mi novia quien en el tiempo que hemos compartido, siempre me ha apoyado en mis días de éxito, y las ocasiones que he caído ha estado a mi lado hasta que me he levantado.

A mis docentes de Colegio y Universidad ellos siempre me inculcaron el valor de la responsabilidad y amor por lo que se hace.

A mis amigos de colegio, a quienes la vida ha llevado por caminos distintos al perseguir sus sueños, pero con quienes a pesar todo puedo contar, y felizmente seguimos unidos y apoyándonos mutuamente cuando es necesario.

Resumen

Se presenta el desarrollo de un aplicativo, programado en Java con el fin de llevar el control de préstamo de los equipos del laboratorio de Electrónica Digital ubicado en el edificio Enrique Rochereaux de la Universidad de Pamplona, este software o aplicativo lleva el registro de cada préstamo guardándolo en una base de datos, con el beneficio de que los equipos sean entregados al estudiante por medio de identificación biométrica. La programación del aplicativo se llevó a cabo con Java orientado a objetos con el fin de no extender la programación y el aplicativo fuera funcional e intuitivo para su manejo; este aplicativo está conectado con una base de datos local, está se trabaja con Mysql bajo el software Xamp que permite la fácil comunicación entre la base de datos y el aplicativo desarrollado.

El aplicativo cuenta con dos interfaces conectadas por una conexión LAN, con el fin de que el estudiante pueda realizar solicitudes sin alterar el trabajo del administrador del aplicativo. Desde la interfaz de administrador se pueden realizar ingresos de nuevos datos de equipos y usuarios, eliminación de datos, y administrar la información del número de equipos disponibles para préstamos.

La implementación del aplicativo permite dejar atrás las solicitudes de equipo por medio físico, pasando a formato digital, permitiendo reducir la cantidad de papel usado, de este modo se apoya la tendencia de cero papeles promovida por la Universidad de Pamplona.

1.Introducción

1. Introducción	2
1.1. Problema	2
1.2. Justificación.....	3
1.3. Objetivos	3
1.4. Distribución Bibliográfica:	4

1.Introducción

1.1. Problema

El uso de equipos del laboratorio de Electrónica Digital del edificio Enrique Rochereaux para el desarrollo de prácticas, montajes o investigación es continuo en transcurso de cada semestre; este laboratorio dispone para las prácticas, diversos equipos de electrónica con el fin de que él o la estudiante y/o docente cuente con los implementos necesarios para ello.

Para el uso de estos equipos, los estudiantes solicitan el préstamo por medio de formato impreso, y para el préstamo, el estudiante debe identificarse con carnet y presentar el formato diligenciado con el fin de validar su identidad; estos formatos son archivados con el fin de llevar un control de préstamo por cada equipo.

Cada semestre los estudiantes desarrollan un gran número de prácticas y la cantidad de formatos archivados al final de semestre es abundante, allí se evidencia una gran cantidad de papel usado; registrar el uso y préstamo del equipo se torna una tarea larga, y en ocasiones este control no arroja un resultado claro, pues al realizar la estadística final de su uso, hay formatos en los que la ortografía o caligrafía no permiten que se identifique de forma clara que equipo fue usado y por quien. Por ende, se propone realizar este proceso de forma digital. La digitalización de dicho proceso permitirá un registro más claro y veraz de estos préstamos.

Este trabajo de grado se hace el desarrollo de Software de una base de datos parametrizado a partir de identificación biométrica el cual requiere conocimiento de programación para el diseño de un software, que registre los prestamos realizados, y cada registro almacene la información necesaria para llevar un control con el objetivo de ser alternativa de cambio al uso de nuevas tecnologías esto en aras cumplir con la directiva de cero papel, promovida por la Universidad de Pamplona.

1.2. Justificación

La aplicación de las nuevas tecnologías para agilizar ciertos procesos permite el mejoramiento de los mismos. Este proyecto tiene la finalidad de proponer ante el Sistema De Gestión(SIG) de la Universidad de Pamplona la implementación de un software para el control de préstamos en el laboratorio de Electrónica Digital ubicado en el edificio Enrique Rochereaux de la Universidad de Pamplona; con esta aplicación se busca que los préstamos estén soportados por el reconocimiento biométrico de cada estudiante que solicite el uso de equipos este laboratorio, esto con el fin de llevar la información de préstamos en una base de datos de la cual se pueda tener información, estado y observaciones de cada uno de los equipos dispuestos para el préstamo a estudiantes. Esta aplicación ayudará al manejo digital de diversos formatos reduciendo el uso de papel al suprimir las solicitudes físicas, participando así en la iniciativa de Cero Papel que la Universidad de Pamplona busca impulsar continuamente.

1.3. Objetivos

Para la realización de un proyecto de desarrollo hay que examinar detenidamente el área sobre la que va intervenir con el proyecto, para determinar las funcionalidades pensando en llevar a un óptimo desempeño los resultados del software. Con este argumento se describe el objetivo principal del proyecto.

- *Desarrollar un software para el control de préstamo de equipos del laboratorio de Electrónica Digital ubicado en el edificio Enrique Rochereaux de la Universidad de Pamplona mediante identificación biométrica.*

También se cuentan con unos objetivos complementarios para el desarrollo de la investigación.

- *Implementar una nueva propuesta de protocolo de préstamo de equipos de laboratorio basado en identificación Biométrica y almacenamiento de*

información en base de datos en tiempo real.

- *Establecer una base de datos en base al formato de toma física de inventario (FAI-09 v0.1) del Sistema Integrado de Gestión de la Universidad de Pamplona para monitoreo y control del préstamo de equipos.*
- *Diseñar una interfaz gráfica de apoyo para el control de los procedimientos establecidos por parte del Sistema Integrado de Gestión para la Gestión de Laboratorios.*
- *Validar el funcionamiento del software desarrollado*

1.4. Distribución Bibliográfica:

El libro se organiza mediante cinco capítulos concernientes a la Introducción, Marco Teórico, Estructura de base de datos, Programación, Resultados y Conclusiones.

Cada capítulo tiene los ítems correspondientes al tema de interés.

2.Marco Teórico

2. Marco Teórico	6
2.1. Base de datos	6
2.1.1. Los Datos	7
2.2. Modelado de datos y modelo de datos.....	8
2.2.1. Concepto de modelo.....	8
2.2.2. Elementos básicos de un Modelo de Datos.....	9
2.2.3. Modelos lógicos basados en objetos	10
2.3. Relaciones y variables de relación.....	10
2.3.1. Diseño de bases de datos con el Modelo E/R	11
2.3.2. Presentación de MySQL.....	11
2.4. El Lenguaje de Programación Java.....	12
2.4.1. Objetivos del diseño de Java	12
2.4.3. Securizado.....	13
2.5. Huella dactilar	14
2.5.1. Precursores en la Biometría	14
2.5.2. Adquisición de Huellas dactilares	14
2.5.3. La Huella dactilar en su estructura como método de identificación ...	15
2.5.4. Reconocimiento por medios de los puntos característicos de Hullas dactilares	16
2.5.4.1. Extracción de Vectores de Características o patrones.....	16



2.Marco Teórico

2.1. Base de datos

“Una base de datos se puede definir como: Una relación de datos integrados como redundancia controlada y con una estructura que se refleja las interrelaciones y restricciones existentes en el mundo real [1].”

La información que se maneja diariamente se interpreta como datos, que se pueden ser compartidos por diferentes usuarios y aplicaciones; Los datos pueden ser de diferentes tipos, cada tipo de dato tiene su descripción única.

Algunos conceptos de Base de datos:

- Un sistema de base de datos es un sistema computarizado para guardar registros, cuya finalidad general es almacenar información y permitir a los usuarios recuperar y actualizar esa información con base en peticiones. La información en cuestión puede ser cualquier cosa que sea de importancia para el individuo u organización; en otras palabras, todo lo que sea necesario para auxiliarle en el proceso general de su administración. La Figura 1 es una imagen simplificada de un sistema de base de datos. pretende mostrar que un sistema de base de datos comprende cuatro componentes principales: Datos, Hardware, Software y Usuarios. [2]
- A diferencia de la forma como se agrupan los datos en los sistemas tradicionales, en estos sistemas no se encuentran datos redundantes, Mas bien, en una base de datos son redundantes o están repetidos únicamente aquellos datos que es imprescindible que lo estén para que en la base de datos se puedan reflejar las relaciones entre los datos. Esto quiere decir que por ejemplo en una base de datos no aparecerá repetida la dirección de un cliente, pero si aparecerá la identificación repetida en varias tablas con el fin de relacionar dichas tablas. [1]

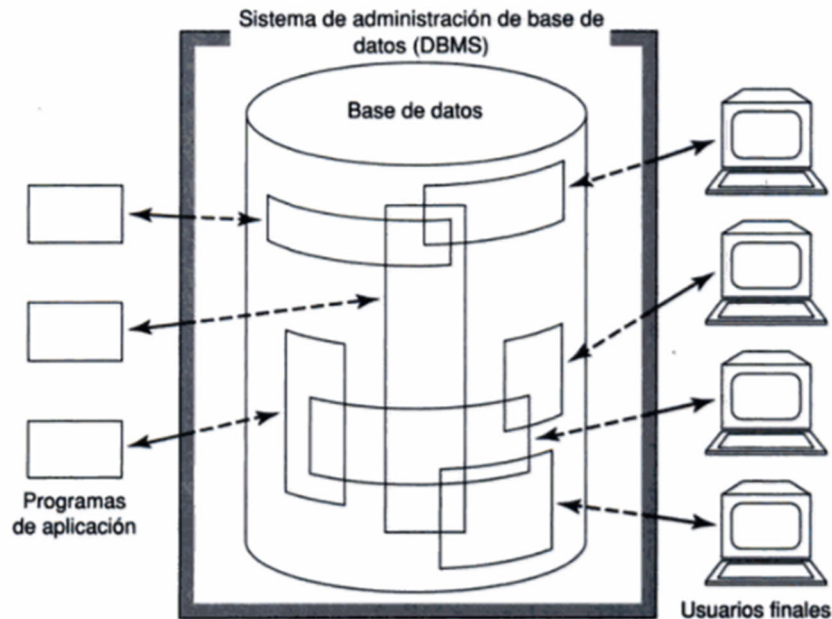


Figura 1. Imagen simplificada de un sistema de base de datos

2.1.1. Los Datos

Los datos de una base son independientes a las aplicaciones, lo que se interpreta cada dato almacenado una base de datos se clasifica según su tipo “la relación de modificaciones sobre el programa no debe implicar modificar la base de datos u viceversa” [3], por lo que modificar un dato dentro de una base de datos no modificara su estructura, esto con el fin de que la manipulación de diferentes usuarios no modifique la estructura y genere errores en los programas que trabajan con la base de datos.

Los datos en general se manejan integrados como compartidos, Esto representa una ventaja importante de los sistemas de bases de datos, ahora veremos una explicación de los términos integrados y compartidos.

Integrados: “Queremos decir que podemos imaginar a la base de datos como una unificación de varios archivos que de otro modo seria distintos, con una redundancia entre ellos eliminada al menos parcialmente [2]”. Donde en una base de datos tendremos datos redundantes como “por ejemplo una base de datos dada



podría contener un archivo EMPLEADO que proporciona los nombres de cada empleado, domicilios, departamentos, sueldos, etc. y un archivo de inscripciones que representa la inscripción de los empleados a los cursos de capacitación [2]”. Como vemos en el ejemplo anterior tenemos dos tablas de una base de datos, una que almacena la información de los empleados, y otra donde se almacena diversos cursos, donde el empleado requiere una inscripción, pero ya se tiene la información del empleado, por tal motivo la inscripción no requiere incluir la información redundante del empleado.

Compartida: “Queremos decir que las piezas individuales de datos en la base pueden ser compartidas entre diferentes usuarios y que cada uno de ellos puede tener acceso a la misma pieza de datos, probablemente con fines diferentes. [2]” tomado el ejemplo que se vio anteriormente de los empleados, podemos decir que los datos se encuentran compartidos y dos dependencias diferentes actúan sobre dichos datos, teniendo que un departamento de Talento humano maneja la información de cada empleado, y un departamento de capacitación maneja las inscripciones,

2.2. Modelado de datos y modelo de datos

2.2.1. Concepto de modelo

Es una representación relativa sencilla, por lo general gráfica, de estructuras de datos reales más complejas. En términos generales un modelo es una abstracción de un objeto o hecho real más complejo, en términos generales, un modelo es una abstracción de un objeto o hecho real más complejo. La función principal de un modelo es ayudar que el lector entienda las complejidades del ambiente real. Dentro del ambiente de una base de datos [4]. El modelo desde un punto de vista científico, es una especie de analogía o imagen mental de los fenómenos en términos de algo que les es familiar. El movimiento ondulatorio de la luz es un ejemplo, no podemos ver las ondas luminosas como podemos ver las ondas en el agua; pero es inútil pensar que la luz está formada por ondas por que los



experimentos de óptica indican que en muchos aspectos la luz se comporta igual que las ondas de agua. [2]

El objetivo de un modelo es ofrecernos un cuadro que se pueda visualizar y ofrecernos una idea que podamos captar cuando no podemos ver que está sucediendo en realidad; “Un modelo nunca es perfecto, y los investigadores tratan constantemente de aginar sus modelos o de establecer nuevos cuando los anteriores ya no son adecuados [2]”.

2.2.2. Elementos básicos de un Modelo de Datos

Los elementos básicos de todos los modelos son: entidades, atributos, relaciones y restricciones. Una entidad es cualquier cosa (una persona, lugar, cosa o hecho) acerca de la cual se han de coleccionar y guardar datos. Una entidad representa un tipo particular de objeto en el mundo real. Como una entidad representa un tipo particular de objeto, las entidades son “distinguibles”, es decir cada vez que se representa una de ellas es única y distinta. Por ejemplo, una entidad CLIENTE tendría muchas ocurrencias distinguibles, por ejemplo, John Smit, Pedro Dinamita, Tom Sticland, etc. Las entidades pueden ser objetos físicos, como clientes o productos, pero las entidades también pueden ser abstracciones, como rutas de vuelo [4].

Un **Atributo** es una característica de una entidad, por ejemplo, CLIENTE sería descrita por atributos tales como apellido paterno, nombre, teléfono, dirección y límite de crédito del cliente. Los atributos son el equivalente de los campos en los sistemas de archivos [4].

Una **Relación** describe una asociación entre entidades. Por ejemplo, existe una relación entre clientes y agentes que se puede describir como sigue: un agente puede atender a numerosos clientes y cada cliente puede ser atendido por un agente. Los modelos de datos usan tres tipos de relaciones: uno a muchos, muchos a muchos, y uno a uno. Los diseñadores de base de datos por lo general usan notaciones breves 1:M o 1..* M:N o *.* y 1:1 o 1..1, respectivamente. (Aun cuando



la notación M: es una leyenda estándar para la relación de muchos a muchos, la notación M:M también puede usarse.) [4].

Una **Restricción** se aplica a los datos, las restricciones son importantes porque ayudan a asegurar la integridad de los datos. Las restricciones se expresan normalmente en forma de reglas. Por Ejemplo:

- El Salario de un empleado puede tener valores que están entre 6 000 y 350 000.
- El PDC (promedio de calificaciones) de un estudiante está entre 0.00 y 4.00.
- Cada Clase debe tener un y solo un maestro. [4].

2.2.3. Modelos lógicos basados en objetos

Permiten representar una estructura bastante flexible y hacen posible la especificación de los límites de los datos. El más representativo es el modelo Entidad-Relación (MER), este modelo consiste en un conjunto de objetos básicos denominados Entidades y de las Relaciones que se dan entre estos. Una Entidad es la representación de un objeto que se puede distinguir de otros, dado a que cada Entidad se le atribuyen ciertos **atributos** que le identifican. Una Relación es una asociación entre varias entidades. [5]

2.3. Relaciones y variables de relación

Al referirnos a bases de datos, el término de bases de datos relacionales es encontrado en diversos libros como “una base de datos relacional es en esencia solo una base de datos en la que los datos son percibidos como tablas [2]”.

Ahora, en que contextos informales es común tratar los términos “relación” y “tabla” como si fueran sinónimos; de hecho, el término “tabla” se usa informalmente con mucha más frecuencia que el término “relación”. En resumen, la explicación es la siguiente:



- Los sistemas relacionales se basan en el modelo relacional. A su vez, este modelo es una teoría abstracta de datos que se está basada en ciertos aspectos de las matemáticas (principalmente en la teoría de conjuntos y la lógica de predicados).
- Los principios del modelo relacional fueron establecidos originalmente en 1969-70 por E.D. Codd (en ese entonces, investigador de IBM). Fue a fines de 1968 que Codd, Matemático de formación, descubrió por primera vez que la disciplina de la matemática podría ser usada en ciertos principios sólidos y cierto rigor a un campo –la administración de bases de datos- que hasta ese momento era muy deficiente en cualquiera de estas cualidades.

2.3.1. Diseño de bases de datos con el Modelo E/R

En cierto sentido, un diagrama E/R construido de acuerdo con las reglas esbozadas en la

2.3.2. Presentación de MySQL

MySQL es un sistema de administración de bases de datos relacionados rápido, robusto y fácil de usar. Se adapta bien a la administración de datos en un entorno de red, especialmente en arquitecturas cliente/servidor. se proporciona con muchas herramientas y es compatible con muchos lenguajes de programación. Es el más célebre SGBDR del mundo open source, en particular gracias a su compatibilidad con el servidor de página Web Apache y el lenguaje de páginas Web dinámicas PHP.

Este servidor de bases de datos es interrogable por SQL(Structure Query Language), el lenguaje estándar más popular para interrogar bases de datos SQL permite manipular los datos muy fácilmente. [5]



2.4. El Lenguaje de Programación Java

Java se caracteriza como un lenguaje sencillo, orientado a objetos, permitiendo que sea un lenguaje robusto, securizado. Se caracteriza a Java como un lenguaje sencillo, orientado a objetos, distribuido interpretado, robusto, securizado, independiente de las arquitecturas, portable, eficaz, multihilo y dinámico. [1]

2.4.1. Objetivos del diseño de Java

En base a las necesidades expresadas, se necesitaba un lenguaje y una plataforma más sencillas y eficientes destinados al desarrollo y al despliegue de aplicaciones securizadas, en sistemas heterogéneos en un entorno distribuido, con un consumo de recursos mínimos y que funcionaran en cualquier plataforma física y de software.

El diseño de Java aportó una respuesta eficaz a esas necesidades:

- Lenguaje de sintaxis sencilla orientado a objetos e interpretación, que permite optimizar el tiempo y el ciclo de desarrollo
- Las aplicaciones son portables sin modificación alguna en numerosas plataformas físicas y sistemas operativos.
- Las aplicaciones son resistentes, porque el motor de ejecución de Java se encarga de la gestión de la memoria (Java runtime Environment), y es más fácil escribir programas sin fallos en comparación a C++, debido a un mecanismo de gestión de errores más evolucionado y escrito.
- Las aplicaciones y en particular las aplicaciones gráficas son eficientes debido a la puesta en marcha y a la asunción del funcionamiento de varios procesos ligeros (thread y multithreading)
- El funcionamiento de las aplicaciones está securizado, en particular en el caso de las applets de Java en los cuales el motor de ejecución de Java se encarga de que el applet no realice ninguna manipulación u operación peligrosa [3].



2.4.2. Robusto

Java es un lenguaje fuertemente tipado y estricto. Por ejemplo, la declaración de las variables debe ser obligatoriamente explícita en Java. Se verifica el código (sintaxis, tipos) en el momento de la compilación y también de las ejecuciones, lo que permite reducir los errores y los problemas de incompatibilidad de versiones. Además, Java se encarga totalmente de la gestión de los punteros y el programador tiene manera de acceder a ellos, lo que evita la sobre escritura accidental de datos en memoria y la manipulación de datos corruptos. [1]

2.4.3. Securizado

Dados los campos de aplicación de java, es muy importante que haya un mecanismo que vigile la seguridad de las aplicaciones y los sistemas. El motor de ejecución java(JRE) es el encargado de esta tarea. El JER se apoya en particular en el archivo de texto java.policy, que contiene la información relativa a la configuración de la seguridad. En java, el JRE es encargado de gestionar el consumo de memoria de los objetos, y no el compilador como es el caso en C++. Puesto que en java no hay punteros sino referencias a objetos, el código compilado contiene identificadores sobre los objetos que luego el JRE traduce en direcciones de memoria; esta parte es totalmente opaca para los desarrolladores.

En el momento de la ejecución de un programa Java, JRE utiliza un proceso llamado el callsLoeader que realiza la carga del bytecode (Lenguaje binario intermedio) contenido en las clases Java. A continuación, se analiza el bytecode con el fin de controlar que no se genere ni manipulación punteros en memoria y que tampoco hubo violación de acceso.

Como Java es un lenguaje distribuido se implementan los principales protocolos de acceso a la red. Se puede configurar el JRE con el fin de controlar el acceso a la red de sus aplicaciones:

- Prohibir todos los accesos.



- Autorizar el acceso solamente a la máquina anfitriona de donde procede el código de aplicación. Es la configuración por defecto para los applets Java.
- Autorizar el acceso a máquinas en la red externa (más allá del firewall), en el caso de que el código de la aplicación también proceda de una máquina anfitriona de la red externa.
- Autorizar todos los accesos, Es la configuración por defecto para las aplicaciones de tipo cliente pesado. [1]

2.5. Huella dactilar

2.5.1. Precursores en la Biometría

Veremos algunos estudios relevantes que han permitido llevar a la huella como una forma de identificación única;

“Marcelo Malpighi, identificó que las características de los dedos no cambiaban y eran diferentes unas de otras y dio la primera noticia sobre estudios científicos de los dibujos papilares [6]”; vemos que se empleó a ver la huella como una forma única por cada ser humano, y concluyó que está no cambia a pesar de los años. “Jan Evangelist Purkinje, estableció que las huellas eran únicas y las figuras que se formaban en ellas no cambiaban con el paso del tiempo; Henry Faulds, publicó varias revistas en las que trató temas de gran importancia como la identificación por medio de la huella dactilar y hace sugerencias en relación a la huella dejada en el lugar de los hechos para identificar a los delincuentes. [7]” Evangelis Pukinje en sus estudios logra presentar varios aspectos de la huella y presenta “las diferentes formas de huellas digitales creando además, un sistema para clasificarlas, dividiéndolas en 9 grupos que son: curva transversal, franja central longitudinal, franja oblicua, presilla oblicua, almendra, espiral, elipse, círculo, doble verticilo; pero su trabajo no tuvo grandes frutos y fue olvidado [7]”;

2.5.2. Adquisición de Huellas dactilares

La forma en la que se realiza el proceso de adquisición de huellas dactilares es muy diferente dependiendo del tipo de aplicación biométrica en la que se van



procesar las imágenes obtenidas, el método tradicional de adquisición de huellas dactilares tomadas han tenido siempre y sigue teniendo lugar, en el ámbito judicial y forense, actualmente los sistemas automáticos de reconocimiento de huellas dactilares tiene también aplicación en otros contextos, como por ejemplo los sistemas de acceso a entornos de seguridad, donde los requeridos de funcionamiento y exigentes de la aplicación, obligan al uso de técnicas on-line. En estos casos las huellas son capturadas a través de un dispositivo de adquisición electrónica. El esquema general de estos dispositivos comprende la siguiente parte: Sensor de lectura encargado de capturar la imagen de la huella; conversor analógico/digital, encargado de convertir la imagen analógica entregada por el sensor de una señal digital; Interfaz de comunicaciones con otros dispositivos externos. dependiendo el principio físico de funcionamiento del sensor, se definen los diferentes tipos: ópticos, de estado sólido y ultrasónico [8]

2.5.3. La Huella dactilar en su estructura como método de identificación

Está formada por crestas papilares y surcos interpapilares están localizados en la piel los cuales forman los dibujos en la falange distal de los dedos de las manos, lo que hace que se formen puntos característicos, un núcleo y uno o varios deltas, con el fin de llevar a cabo la verificación de identidad [9], para entender de forma gráfica los diferentes aspectos que conforman una huella veremos en la Figura 2. la cual se señala de forma superficial cada uno de las características encontradas.

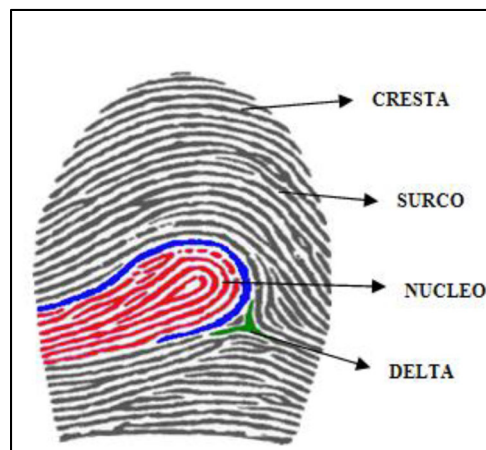


Figura 2. Huella dactilar, ilustrando los patrones de una Huella



La anterior figura puede la vista de una huella escaneada, de la cual se pueden ampliar y extraer la serie de patrones, estos patrones son detalles que se forman en las crestas papilares las cuales son encenízales para la identificación de una persona a otra.

2.5.4. Reconocimiento por medios de los puntos característicos de Huellas dactilares

Una huella dactilar cualquiera, es la representación de la morfología superficial de la epidermis de un dedo, ésta posee un conjunto de líneas que, en forma global, aparecen dispuestas en forma paralela. Sin embargo, estas líneas se intersectan y a veces terminan en forma abrita. Los puntos donde están terminan o se bifurcan se conocen técnicamente como puntos característicos. Para concluir si dos huellas dactilares corresponden o no a la misma persona se lleva a cabo un procedimiento o comparación de los puntos característicos de ambas huellas; La correspondencia de huellas dactilares consiste en encontrar el grado de similitud entre dos vectores de características cutas componentes representan a los puntos característicos de cada huella. [10]

2.5.4.1. Extracción de Vectores de Características o patrones

Por razones de seguridad, así como de capacidad de almacenamiento, no es prudente almacenar las imágenes completas de las huellas dactilares en el sistema de reconocimiento de huella dactilar. Lo habitual es hacer un backup de la imagen completa durante el registro en un lugar seguro, pero para el funcionamiento habitual del sistema lo que se hace es, extraer una plantilla única de la imagen en función del método de reconocimiento de patrones o el principio de minucias. Las huellas digitales no solo se componen de segmentos de curva continuos en forma de arcos, curvas y espirales, sino que también cambian de dirección se cortan o bifurcan, los puntos donde estos cambios ocurren se denominan minucias, a continuación, se detallan los tipos de minucias yen la figura 3. [11]

- **Laguna:** Cresta que se divide y vuelve a unirse formando una superficie cerrada.



- **Isla:** Cresta diminuta, prácticamente un punto.
- **Bifurcación:** Punto donde la cresta se divide en dos ramas.
- **Espolón:** Cresta que se bifurca y una de las ramas tiene una longitud muy pequeña.
- **Terminación:** Donde la cresta termina abruptamente.
- **Cruce:** Cresta que une otras dos crestas paralelas. [11]

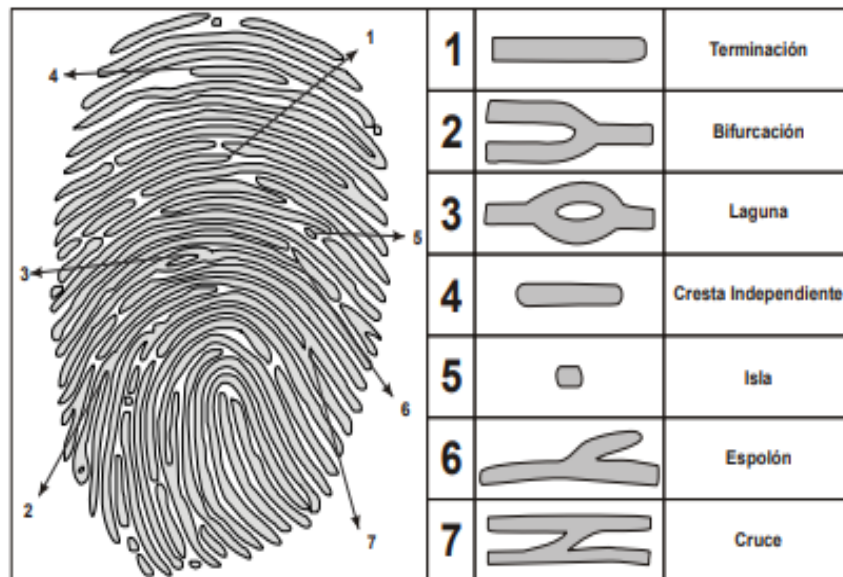


Figura 3. Tipos Patrones de identificación

Los patrones de identificación se pueden dividir en dos categorías

- **Básicas:** Son las de terminación y bifurcación
- **Compuestas:** El resto de patrones pertenecen a esta categoría, pues se originan a partir de las básicas.

Y los principales atributos usados para su almacenamiento son:

- **Dirección:** la determinan la dirección de la cresta a la que pertenecen.
- **Posición:** Posición geométrica de patrón respecto a un eje de referencia.
- **Frecuencia Espacial:** Inversa de la distancia entre dos crestas consecutivas.
- **Curvatura:** índice de variación en las direcciones de las crestas.



Durante el registro, de un nuevo usuario, el sistema realiza un tratamiento de imagen para obtener un patrón de minucias como se observa en la Figura 4, este proceso proporciona la platilla de registro, la cual para un nuevo registro se hacen entre 3 a 4 muestras y durante la verificación se realiza el mismo proceso, la diferencia es que solo se toma una muestra y el patrón de minucias se compara con los registros almacenados del registro.



Figura 4. Tratamiento de imagen para extracción de patrones

Como se observa en la anterior figura, la huella original es una imagen digital la cual se obtiene con un sensor de imagen con la capacidad de digitalizar la señal producida por este. Una vez obtenida la imagen, esta se debe mejorar por medio de filtros (como ejemplo por FFT), de este proceso se obtiene una imagen como se observa de la Figura 4, campo de orientación, paso a seguir es la binarización, la cual consiste llevar la imagen de grises, preparando la imagen para el siguiente proceso, el adelgazamiento, que consiste en aplicar algoritmos consecutivos de adelgazamiento de imágenes con el fin de reducir el grosor de las crestas en la imagen binaria, de esta imagen ya adelgazada se extraen las minucias o características de Galton que consiste el patrón biométrico de la huella. De estos patrones se superponen en la imagen adelgazada para corroborar que cada pixel permanece en una cresta, en caso positivo, si pertenece a una bifurcación o un principio o final de cresta [12].



Algunas de las razones por la que se usa este procedimiento:

- En cada huella dactilar existen entre 30 y 40 muestras, típicamente 36, si cada una se puede almacenar en 4 bytes, Toda la información necesaria para la verificación de un individuo a partir de su huella dactilar se puede almacenar en tan solo 144 bytes, en lugar de 140 Kbytes de una imagen. [11]
- La huella dactilar no se puede recuperar desde una plantilla. De esta forma se reducen las posibilidades de fraude en caso de poder acceder en la base de datos. [11]
- La plantilla puede ser comprimida en cualquier algoritmo de compresión de datos, e incluso someterse a un proceso de codificación. [10]

3. Metodología, estructura y programación

3.	Metodología, estructura y programación.....	21
3.2.	Estructura Base de Datos	24
3.2.1.	Identificación Entidades y relaciones	26
3.2.2.	Modelo Entidad Relación (MER).....	28
3.2.3.	Creación base de datos en Xampp	30
3.3.	Lector de Huellas Digital Perna U.are.u 4500.....	31
3.3.1.	Configuración y programación de Digital Personal U.are.U 4500 USB Biometric Fingerprint Scanner Fingerprint Reader.....	32
3.3.2.	Creación de un Objeto.....	33
3.3.3.	Aplicación del lector de Huellas	33



3. Metodología, estructura y programación.

Un software que lleve el registro de préstamos realizados, requiere que los usuarios, en este caso los estudiantes se registren en el sistema; este registro se usa para tener una base de datos de los estudiantes que realizan algún préstamo dentro del laboratorio, la información que se recopila se usa para llevar un control en los préstamos de equipos de laboratorio de una formas más ágil, precisa y robusta, del esto de los equipos y de quien hace su uso en cada práctica.

El proceso actual de préstamo que los estudiantes realizan, consiste en llenar el formato que se denomina “*Solicitud y Entrega de Materiales, Insumos y Equipos FLA-03 v.04*” del Sistema de Gestión de Calidad de la Universidad de Pamplona, dicho formato se tendrá en cuenta la información que se solicita con el fin de que el proceso de préstamo no se vea alterado y no gane complejidad. El sistema contará con una base de datos, donde se irá actualizando toda la información de cada préstamo.

Tomando en cuenta las necesidades planteadas, el principal objetivo es llevar el proceso de manera digital, debido a esto se observa la necesidad y las funciones que cumple el formato de solicitud de préstamos, buscando tener la misma eficacia que realizar el préstamo con el formato físico. Antes de estructurar una solución se solicitó asesoría de un profesional del tema. En este caso el Ingeniero de sistemas Jorge Andrés Sánchez Carrillo quien orientó el desarrollo de la aplicación.

Para cumplir este objetivo se tienen en cuenta los requisitos necesarios para el diseño de un aplicativo que permita hacer solicitudes, y que se puedan modificar, aceptar, o negar según sea el caso; llevar el registro de esta información manera ordenada requiere estructurar una base de datos, para ello, se identificaron las entidades que se utilizaron en la base de datos que almacenará la información, para ello se realizó un análisis del proceso que se lleva de manera física con el fin de identificar la forma en que los formatos están organizados y los requerimientos de la base de datos como tablas, llaves principales y demás información pertinente.

Durante el desarrollo del aplicativo se encontró la necesidad de tener dos interfaces que faciliten el uso del mismo: el administrador y los estudiantes. La primera proyectando a que será usada y alimentada por un auxiliar y la segunda para las personas que soliciten prestamos, estas sobre la misma red LAN, el cual



permita a los estudiantes hacer consulta de disponibilidad y solicitudes de manera remota.



3.1. Diagrama de flujo solicitud de préstamos

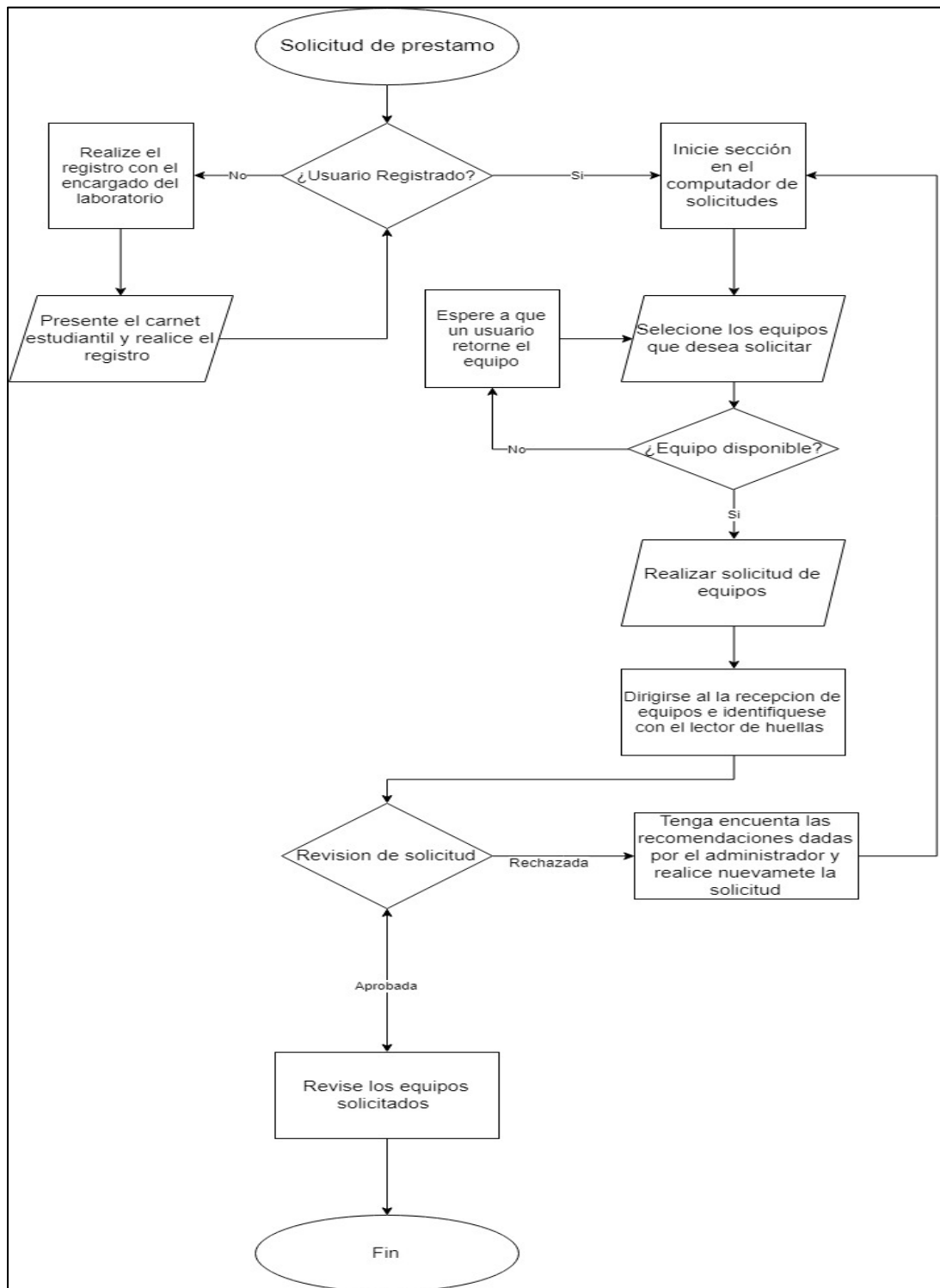


Figura 5. Diagrama de flujo Solicitud de préstamo



Del diagrama de flujo representado gráficamente en la Figura 5, se realiza el proceso de una nueva solicitud de préstamo, en la cual se establece una ruta, donde se plantea en varias partes de proceso una serie de preguntas con el fin de aclarar el proceso.

En la primera pregunta en el sistema de préstamos se plantea para un usuario registrado o nuevo, y que hacer en cada uno de los dos casos; Donde para un usuario nuevo debe realizar un registro previo en el sistema, realizado el registro retorna al punto de inicio y puede realizar una solicitud, para el caso de un usuario registrado el sistema pide iniciar sesión con el número de documento registrado, en un computador cliente dispuesto en el laboratorio para esta tarea, en este ya puede seleccionar los equipos que requiera, donde existe dos casos para la selección de un equipo, el cual está sujeto a la disponibilidad del equipo, y en caso de que no este disponible se debe acercarse al auxiliar del laboratorio para que dé información del equipo, o sugiera una alternativa, en el caso de que los equipos requeridos estén activos, el usuario puede realizar la solicitud y reclamar los equipos en el área dispuesta, la cual se podrá identificar con el lector de huella para mayor seguridad y rapidez en la solicitud, en este paso el auxiliar encargado podrá, aprobar o rechazar la solicitud, la opción de rechazar se usa en casos de que el estudiante se retrate del préstamo de algún equipo, o el auxiliar vea que esa solicitud expire por la fecha de la solicitud, en este caso la solicitud se vuelve al inicio. En caso de ser aprobada se procede a inspeccionar los equipos solicitados para su uso.

3.2. Estructura Base de Datos

Para estructurar una base de datos es necesario conocer el proceso del cual se quiere llevar un registro; de este proceso se identifican los datos que son críticos, que son aquellos de importancia para realizar el préstamo de un equipo, estos datos permiten que el sistema pueda realizar comprobaciones o relaciones entre cada



tabla de la base de datos. Estos datos se priorizaron al revisar el formato actual.

Solicitud y Entrega de Materiales, Insumos y Equipos						Código	FLA-03 v.04			
						Página	1 de 1			
Fecha de Solicitud:	D.	M.	A.	Fecha de Entrega:	D.	M.	A.			
Laboratorio:				Práctica:						
Profesor:				Hora:		Solicitud N.		Número de grupos		
Cantidad Solicitada	Cantidad Entregada	Materiales, Insumos y Equipos				Nº DE GRUPOS ENTREGADOS				
		G1	G2	G3	G4	G5	G6	G7	G8	
Observaciones										
Docente a Cargo					Documento de Identificación					
Nombres y Apellidos					Código y/o Documento de Identificación					
G1										
G2										
G3										
G4										
G5										
G6										
G7										
G8										
Auxiliar Responsable										
Nombres y Apellidos (Firma)					Cédula		R	P	E	Rc
R: Recepciona.		P: Prepara.			E: Entrega.		Rc: Recibe.			<input type="checkbox"/>

Figura 6. formato FLA-03 v.04

En la figura 6, se observa el formato de “Solicitud y entrega de Materiales, Insumos y equipos” con él se lleva el proceso en este momento. En este formato se solicita el listado de equipos necesarios para el desarrollo de las actividades prácticas de las asignaturas que requieran el uso de los equipos del laboratorio por parte de los. Los proyectos de clase se realizan en grupos y la solicitud de los materiales sólo figura a nombre de un responsable por cada grupo y no por persona. Analizando las casillas que tiene el formato, se puede observar que para cada préstamo el estudiante requiere llenar los datos personales como nombre y documento, fecha de solicitud, fecha de entrega, docente a cargo, nombre de la



práctica entre otros, finalmente el formato solicita los datos de las personas autorizadas por la Universidad de Pamplona para la recepción, Preparación, Entrega y quien recibe los equipos.

3.2.1. Identificación Entidades y relaciones

Analizado el formato físico del actual método de préstamo, se procede a la identificación de las entidades. Cada entidad identificada se convertirá en una tabla de la base de datos. Estas entidades tienen ciertos atributos, que son las propiedades que describen a nuestra entidad, uno de estos nos permitirá relacionar una entidad con otra, este atributo se conoce como llave primaria de la entidad.

Al realizar el análisis de los datos que se manejan en el formato FLA-03 v.04, se logran identificar de las siguientes entidades:

- **Estudiante**
- **Usuario**
- **Equipo**
- **Préstamo**

Además de las entidades anteriormente enumeradas se necesita una que permita incluir en la base de datos, datos biométricos que permitirán la identificación de los estudiantes que realicen préstamos. De éste análisis se deduce la necesidad de una entidad para los datos generados por el lector de huellas, otra entidad que los relacione y guarde con más detalle los datos de cada préstamo y una tercera entidad que relacione y agrupe los equipos con el propósito de llevar un control de disponibilidad por cada tipo de equipo, de lo anterior se obtiene como resultado de tres entidades adicionales:

- **Huella**
- **Tipo_Equipo**
- **Detalle_Préstamo**

A estas entidades se le asignan atributos, cada una de estas propiedades permiten almacenar los datos relacionados con la entidad, estas las nombraremos con el comienzo del nombre de la identidad separadas por raya al piso seguido con una palabra que describa la propiedad, con el objetivo de al momento de trabajarlas



en el código evitemos confusiones, en la siguiente Tabla 1, desglosaremos cada entidad con sus respectivos atributos y su relación con la variable en Mysql para tener una idea los datos que vamos a manejar en el software.

Identidad	Atributos	Representación en Mysql: Tipo variable (Longitud)
Estudiante	Código estudiante Nombre estudiante Apellido estudiante Sexo Número de identificación Teléfono estudiante Email estudiante Dirección estudiante	Cod_Est: Varchar (10) Nom_Est: Varchar (30) Ape_Est: Varchar (30) Sex_Est: Varchar (1) Nid_Est: Varchar (10) Tel_Est: Varchar (12) Email_Est: Varchar (50) Dir_Est: Varchar(30)
Usuario	Identificación usuario Cedula usuario Nombre usuario Apellido del usuario Nick de usuario Contraseña de usuario	Usu_id: Varchar (11) Usu_ced: Varchar (12) Usu_nombre: Varchar (30) Usu_ape: Varchar (30) Usu_login: Varchar (20) Usu_pass: Varchar (20)
Huella	Código de huella Nombre de huella Registro de huella Código estudiante	Hue_codi: int (11) Hue_nombre: Varchar (100) Hue_huella: blob Cod_Est: Varchar(10)
Equipo	Numero de chapa Descripción equipo Marca del equipo Modelo de equipo Observaciones Identificación de tipo equipo	Cod_Equi: Varchar (6) Descripcion: Varchar (50) Mar_Equi: Varchar (50) Mod_Equi: Varchar (50) Observaciones: Varchar (50) Tp_id: int(11)
Tipo Equipo	Identificación de tipo equipo Nombre tipo equipo Cantidad de equipos por tipo Cantidad de equipos prestados	Tp_id: int (11) Te_nombre: Varchar (50) Te_cantidad: Int (11) Te_prestados: int(11)
Préstamo	Consecutivo de préstamo Código de estúdiate Fecha de préstamo	Num_Pre: int (11) Cod_Est: Varchar (12) Fec_Pre:Varchar(12)
Detalle Préstamo	Consecutivo de préstamo Numero de chapa Descripción equipo Estado de solicitud Identificación Usuario	Num_Pre: int (11) Cod_Equi: int (11) Descripcion: Varchar (50) Estado: Int (4) Usu_id: Varchar(11)

Tabla 1. Relación de propiedades de identidad con variables creadas en Mysql



3.2.2. Modelo Entidad Relación (MER)

Definidas nuestras identidades, se hacen relaciones entre cada entidad, con el objetivo de verificar el tipo de relación (uno a muchos, o muchos a uno). Cada entidad se relaciona una con otra, como resultado tendremos varias relaciones y más de una será del tipo muchas a muchas, estas relaciones se deben romper o eliminar, tendremos relaciones redundantes que también se deben eliminar, estas relaciones solo se ven si son redundantes al momento de hacer el MER.

El modelo de entidad relación (MER) que presentamos a continuación, están reflejadas la relación no redundante. Las entidades están representadas como un cuadro, cada cuadro tiene sus atributos, en el diagrama representamos las relaciones con una línea trazada, esta línea tiene un extremo abierto y un extremo una raya, indicando una relación muchos a uno.

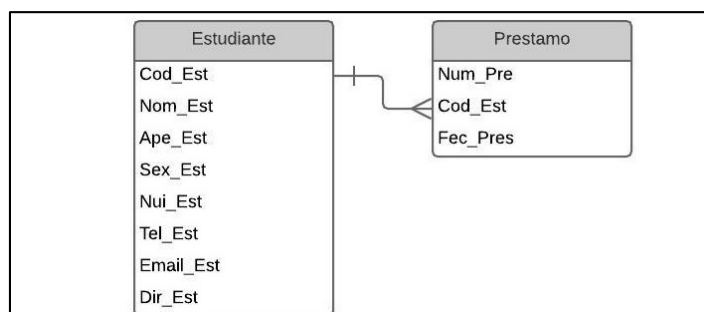


Figura 7. Diagrama relación de dos entidades

La relación anterior representa que un estudiante puede hacer muchos prestamos, y que cada préstamo debe ser solo un estudiante. Completando las relaciones de todas las identidades y dejando las relaciones que mas se acomoden al sistema, se presenta el MER completo en la Figura 8, siendo la estructura de la base de datos que se va a trabajar, como se puede ver las relaciones no hay redundancia de datos ni relaciones Muchas a Muchas, en este caso una relación muchos a muchos fue la de un estudiante puede prestar muchos equipos y un equipo puede ser prestado por muchos estudiantes, en este caso se fue necesario crear una tabla pivote, con el fin de romper la relación muchos a muchos, ya que en esta tabla pivote que en su defecto fue detalle préstamo, colocamos las dos propiedades de mucho a mucho.

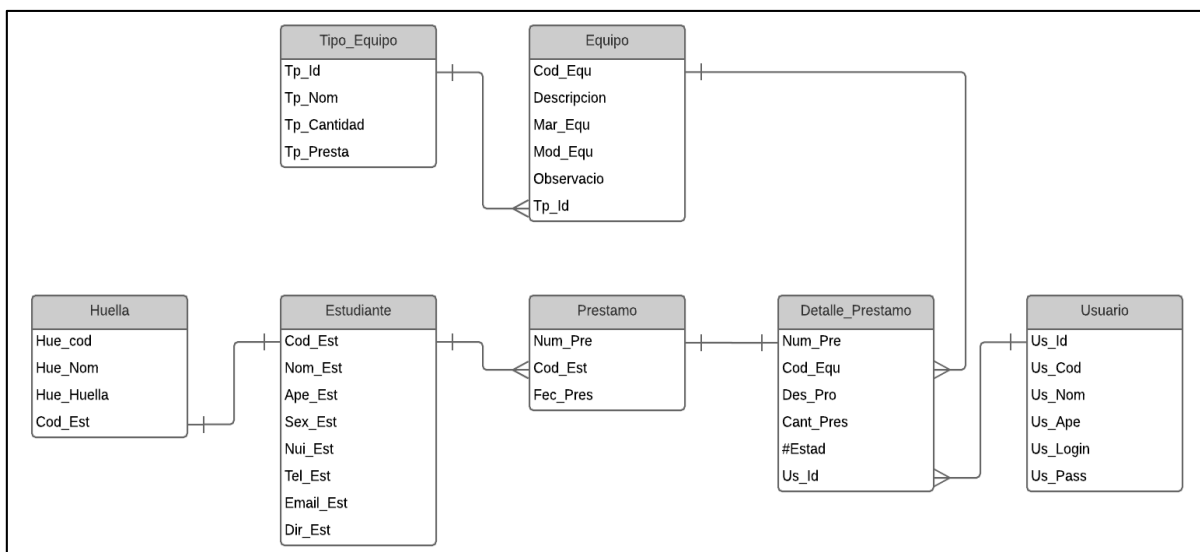


Figura 8. Modelo de Identidad relación

Como se ven en el MER las relaciones que encontramos cuentan con su variable en cada identidad, con el fin de que solo esta variable sea redúndate en la base de datos, también encontramos las relaciones acordes al sistema de préstamos, las cuales describiremos cada una de las relaciones en la Tabla 2, la cual se describe la relación, el tipo de relación y las entidades que se relaciona por cada relación.

Descripcion de Relación	Tipo de relación	Identidad inicio	Identidad fin
Un estudiante solo puede tener una huella, y solo puede haber una huella por estudiante	Uno a uno	Estudiante	Huella
Un solo estudiante puede realizar un préstamo, Hay muchos préstamos por estudiante	Uno a muchos	Préstamo	Estudiante
Un detalle por cada préstamo y un préstamo por cada detalle	Uno a uno	Detalle préstamo	Préstamo
Un préstamo por muchos usuario, y un solo usuario puede realizar un préstamo	Uno a muchos	Detalle Préstamo	Usuario
Un préstamo por cada equipo, muchos equipos por préstamo	Uno a muchos	Detalle Préstamo	Equipo
Un equipo por tipo de equipo, y muchos equipos por tipo de equipo	Uno a muchos	Equipo	Tipo Equipo

Tabla 2. Descripción de relaciones del MER



3.2.3. Creación base de datos en Xampp

Determinadas claramente las entidades, sus respectivas propiedades y establecidas las relaciones entre entidades, se procedió a crear la base de datos en el programa que nos permita realizar una fácil conexión, el programa seleccionado es Xampp, esta selección se hizo por la difusión del mismo y la sencillez de uso para el desarrollo de este tipo de proyectos. La instalación del programa es sencilla se puede descargar desde la página oficial del software sin ningún costo. Una vez instalado Xampp se ejecuta desde la capeta que se instaló y la aplicación de “XAMPP_Control”, esta nos abre la ventana de opciones ver Figura 9 y se selecciona en la parte de Mysql opción Admin.

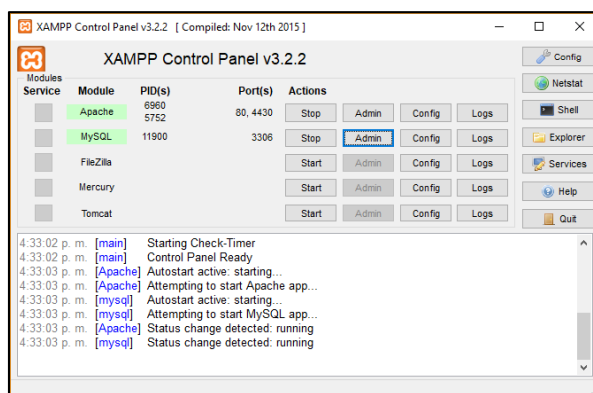


Figura 9. Pantalla de Inicio XAMPP

Esta opción nos re direccionará a una página en el navegador predeterminado del sistema, donde se solicitan los datos de acceso a sistema, por defecto el usuario de inicio es “root” sin clave, esta pestaña del navegador se ve en la Figura 10.

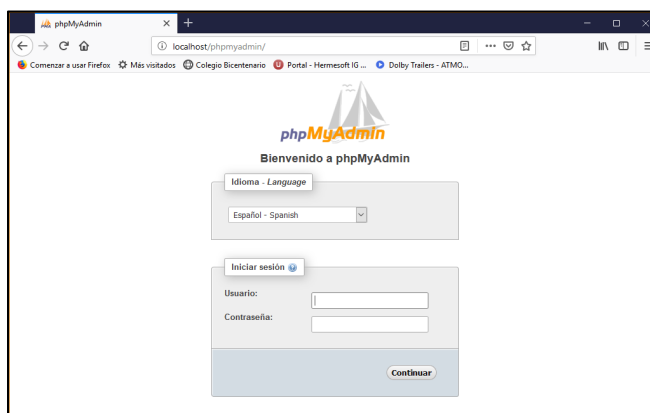


Figura 10. Pantalla de bienvenida phpmyadmin



Una vez iniciado el programa se empieza con la creación de la base de datos desde cero, en la Figura 11 vemos que por el momento no hay ninguna base de datos, de forma que solo se tiene que digitar el nombre de la base de datos y seguido crear las entidades y las propiedades de cada una, el programa permite crear las relaciones de las variables entre las entidades.



Figura 11. Creación de una tabla en phpmyadmin

Las entidades serán nuestras tablas, y en el número de columnas de cada una de ellas corresponderá a la cantidad de propiedades que designamos por cada entidad. Como recomendación para la creación de tablas, se debe empezar por aquellas que tienen una relación de uno a muchos. En el caso de la aplicación desarrollada se tienen las entidades de Huella, Tipo_Equipo y Usuario y demás las cuales se mencionaron anterior mente, al momento de crear una propiedad de una entidad, el sistema nos permite configurarla como obligatoria u opcional, lo cual en el sistema se refleja como que campos son obligatorios al momento de realizar un nuevo registro.

3.3. Lector de Huellas Digital Perna U.are.u 4500

Este lector de huella en la creación del software de préstamo, nos ayudara a facilitar la identificación del estudiante, lo cual, en nuestro sistema será como la firma personal de cada usuario que solicite el préstamo dentro del laboratorio. El uso del lector Digital Persona nos ahorra trabajo y horas de programación, ya que el sistema cuenta con sus propias librerías.



3.3.1. Configuración y programación de Digital Personal U.are.U 4500 USB Biometric Fingerprint Scanner Fingerprint Reader

Para trabajar con este dispositivo es necesario contar con los archivos de instalación, los cuales se encuentran en la página oficial de la marca, que ofrece su descarga de forma gratuita. Este instalador cuenta con los controladores para las versiones de Windows, y las librerías para su trabajo en Java, las que permiten activar o desactivar el dispositivo. El dispositivo tiene como finalidad obtener la imagen escaneada de la huella, y cuenta con una pequeña interfaz que permite su enlace por medio de códigos, para que sea usada solamente en el momento que el usuario indique la necesidad de lectura de la huella. Se realizó la instalación de los controladores que vienen en la descarga, esta se hizo siguiendo los pasos indicados y los drivers se configuran de forma automática. Para la integración con el software diseñado en Java, se deben copiar los archivos de las librerías, éstas la encontramos en la carpeta raíz, que para este caso sería: *“C:\Program Files\DigitalPersona\Bin\Java”* la cual veremos el contenido de esta carpeta en la Figura 12.

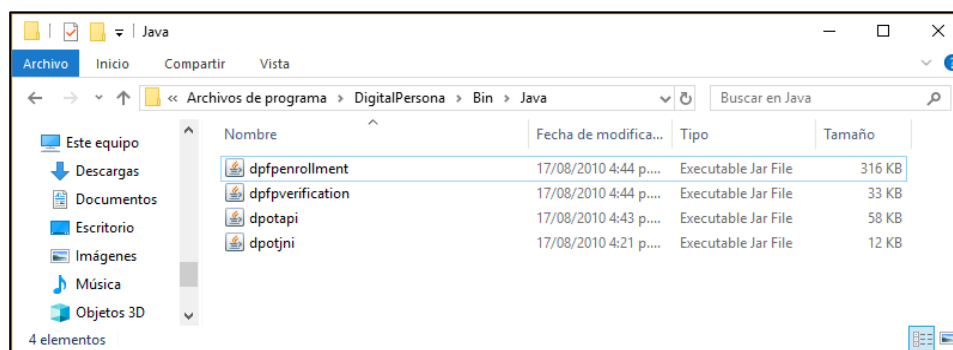


Figura 12. Captura Pantalla, Ubicación de librerías para Digital Persona

Las librerías necesarias son copiadas en la ubicación dentro del directorio Bin la carpeta Java, además se copian los 4 archivos .jar que se encuentran en ella hacia las carpetas *“C:\Archivos de programa\Java\jdk\jre\lib\ext”* y en la carpeta *“C:\Archivos de programa\Java\jre*\lib\ext”*.



3.3.2. Creación de un Objeto

Los objetos hacen referencia a los grupos de métodos usados para la ejecución de tareas específicas de la aplicación. En este caso un objeto será el encargado de la vinculación del lector de huellas, y la creación de los métodos que se crean para el correcto funcionamiento de la captura de las huellas digitales por medio del lector.

Los métodos creados, usaran las librerías que vienen incluidas en el controlador del lector de huellas; estas librerías permiten el control del lector su activación y las ordenes de verificación, inscripción y apagado del mismo. Los métodos creados se desarrollaron bajo la figura de un método, con el propósito de llamar las rutinas en los diferentes objetos que se creen dentro del programa, esto ahorra tiempo en la programación y reduce las líneas de código necesarias.

3.3.3. Aplicación del lector de Huellas

El lector de huellas con los métodos creados dentro de un objeto nos permitirá la vinculación en diferentes objetos de la programación del software. Como se describe en el anterior capítulo, la interfaz del lector de huellas Digital Persona 4500 nos permite realizar registros de huellas nuevas, las cuales se almacenan en la base de datos, la cual para el momento de identificación de una huella el sistema compara los datos guardados, ésta comparación dará como resultado el número de registro de la huella que, por medio de consulta y relación en la base de datos, se relacionará con usuario que está solicitando el préstamo, permitiendo de forma rápida y segura la identificación.

3.4. Conexión Base de datos con el aplicativo.

Una vez configurado y probado el lector de huellas, se procede a la conexión con la base de datos; esta conexión se realiza con el uso de comandos SQL, y las librerías dispuestas en JAVA, que permiten la misma. Para realizar la conexión es necesario el ingreso con el usuario y contraseña de inicio de xampp. Como se ve en la captura de pantalla ilustrada en la Figura 13, se importan las librerías en java lo que permitirá la conexión, y se crea la clase conectar con el fin de que se pueda



llamar cuando se deseen realizar consultas a la base de datos.

```
package claseConectar;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.IOException;
import java.sql.*;
import javax.swing.*;

public class conectar {
    Connection conect = null;
    public Connection conexion()
    {
        try {
            //Cargamos el Driver MySQL
            Class.forName("com.mysql.jdbc.Driver");
            String direccion=leer("direccion.txt");
            conect = DriverManager.getConnection("jdbc:mysql://"+direccion+"/prestamoequipos","root","reactivosunipam16");
        } catch (Exception e) {
            JOptionPane.showMessageDialog(null,"Error "+e);
        }
        return conect;
    }
    public static String leer(String nombre)
}
```

Figura 13. Captura de pantalla, conexión base de datos con XAMPP

3.5. Diseño Interfaz principal

El diseño de la interfaz se realizó usando como base los programas que los usuarios manejan con cotidianidad. La interfaz se compone de una ventana con una barra de opciones, que despliegan una serie de funciones, acordes a las necesidades establecidas partiendo del proceso actual, la interfaz busca que las solicitudes sean fluidas,

Teniendo en cuenta lo anterior, se establece que el estudiante podrá realizar búsquedas de los equipos que se encuentran disponibles, y seleccionar los que requiere. Y que la persona a cargo debe autorizar su préstamo. De esta manera se estableció que la interfaz debe tener dos roles, uno de ellos el estudiante, que será el usuario final y solo tiene acceso al aplicativo para realizar solicitudes de préstamos, y un segundo rol, un usuario administrador que contará con opciones para el registro de información, registro de estudiantes, aceptar o denegar las solicitudes realizadas.

La aplicación requiere que el estudiante al finalizar una solicitud de préstamo se identifique con un lector de huellas; el desarrollo de la interfaz se realizó por medio del entorno de programación NetBeans, entorno de código libre que permite el desarrollo y articulación con diversos lenguajes de programación.

4.Aplicativo en Java

4.	Aplicativo en Java.....	36
4.1.	Características del sistema.....	36
4.2.	Rol –administrador-.....	36
4.2.1.	Archivo.....	38
4.2.2.	Funcionalidades.....	39
4.2.2.1.	Estudiantes.....	39
4.2.3.	Prestamos.....	42
4.2.3.1.	Inicio Sesión.....	42
4.2.3.2.	Proceso de solicitud.....	42
4.2.4.	Consultas.....	44
4.2.4.1.	Estudiantes.....	44
4.2.4.3.	Solicitudes.....	45
4.3.	Rol –Estudiante-.....	46
4.4.	Pruebas del aplicativo.....	47
4.5.	Errores detectados y solución.....	49



4. Aplicativo en Java

4.1. Características del sistema

El sistema de préstamos, cuenta con diversas funciones que serán descritas de forma detallada más adelante. Una de las funciones más relevantes es la conexión con la base de datos local, instalada por el software Xampp; este software permite una conexión sencilla entre el aplicativo y la base de datos. Otra característica para resaltar es la función de reconocimiento biométrico para la identificación de usuarios registrados en el sistema, permitiendo al administrador comprobar la identidad del usuario sin necesidad de solicitar el carnet, además cabe mencionar la existencia de un segundo rol diseñado para estudiantes, que permite al usuario previamente registrado en el sistema, realizar una solicitudes desde un computador remoto conectado a la red local, con ello el estudiante tendrá información actualizada de los equipos disponibles para préstamo.

4.2. Rol –administrador-.

El administrador del aplicativo será quien tenga control sobre los datos incluidos en la base de datos; La persona a cargo de este rol tendrá los privilegios y la autorización para Crear, Modificar o Eliminar datos, teniendo en cuenta que manejará información sobre Estudiantes, Equipos disponibles y tipo de equipos

Este rol será manejado por la persona a cargo de la aprobación de solicitudes, entrega y recepción de los equipos; para el ingreso de este rol, es necesario un usuario y contraseña, por defecto una que no está en la base de datos, con el fin de que en caso de un usuario administrador se le olvide la contraseña, desde este usuario se pueda actualizar una nueva. Como veremos en la siguiente Figura 14 es la ventana de inicio al programa al momento de ejecutarlo.

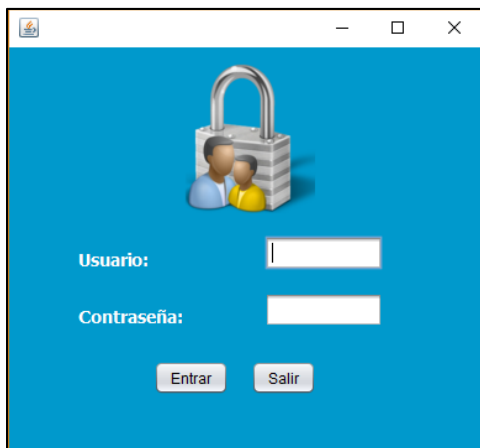


Figura 14. Ventana de inicio sección rol-administrativo

La interfaz se diseñó con una serie de pestañas para un manejo intuitivo del sistema; al desplegar cada pestaña, se pueden acceder diversas opciones que cumplen una función específica del sistema.

Las ventanas por cada pestaña son:

- Archivo
 - Cerrar Sección
 - Salir
- Funcionalidades
 - Estudiantes
 - Equipos
 - Tipo Equipos
- Préstamo
 - Realizar Solicitud
- Consultas
 - Estudiantes
 - Productos
 - Consulta Solicitudes
- Administración
 - Usuarios



Algunas funciones son compartidas por los roles de usuario y administrador, -préstamo- es una función compartida, si el usuario no ha hecho la solicitud previamente, el administrador puede recibir la solicitud de forma verbal, registrarla y hacer la autenticación.

Otra de las funciones es la de consulta, ésta como su nombre lo indica permite al administrador hacer consultas de disponibilidad de equipos, siendo una de las funciones más importantes del sistema; en ella el administrador puede ver las solicitudes en estado de espera, préstamo o recibido, y quién entregó los materiales solicitados.

Cada una de las opciones ejecuta funciones relacionadas al nombre que las identifica; al ser un programa con código en Java orientado a objetos, el sistema es fluido a continuación, se describen de forma detallada las funciones contenidas en el aplicativo:

4.2.1. Archivo

En esta pestaña se encuentran las funciones relacionadas al cierre de sesión o el salir del programa como se puede apreciar en Figura 15. El nombre se toma de los softwares de uso cotidiano lo que permite que los usuarios relacionen las funciones aquí contenidas con el manejo de otros aplicativos, en esta pestaña se ubican las funciones fundamentales de todo software. Permite que el usuario inicie o termine su sesión.

Las sesiones facilitan la identificación de la persona que esté a cargo de la aprobación de solicitudes, lo que proporcionará un registro de la persona que realizó la entrega y permitirá llevar un seguimiento minucioso en los préstamos.

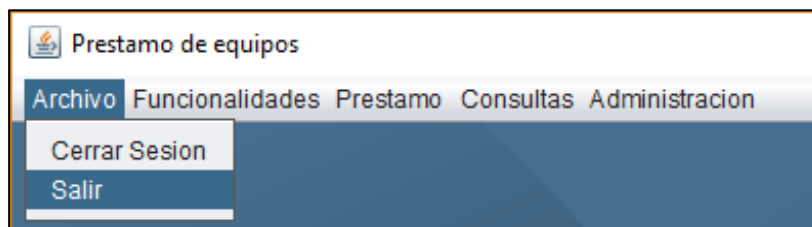


Figura 15. Funciones Archivo



4.2.2. Funcionalidades

Permite el registro de información en la base de datos, en este apartado encontramos las opciones de registro de estudiantes, equipos y tipos de equipos. Estos datos son necesarios en el sistema ya que sobre ellos se realizan las consultas y se conoce de la disponibilidad de equipos para prestamos

4.2.2.1. Estudiantes

En esta función el administrador puede registrar, modificar o eliminar información de los estudiantes (usuarios). Para el registro de un usuario (estudiante) es necesario que él esté presente, ya que en el proceso de registro se solicita el registro dactilar, también se piden otros datos como: nombre, número de identificación y otros datos básicos del estudiante. El registro de los estudiantes solo puede ser realizado desde el rol de administrador.

En la siguiente Figura 16 la se puede observar la ventana de registro de estudiantes; en ella aparecen las diversas opciones con la que el administrador cuenta, como: Nuevo, Guardar, Actualizar, Cancelar y Salir. Estas opciones como sus nombres lo indican, permiten la administración de la información de los estudiantes en la base. Al dar clic derecho sobre uno de los estudiantes registrados permite modificar o eliminar su registro.



Figura 16. Registro de Estudiante.



El registro de un nuevo estudiante requiere el registro dactilar, para ello se debe tener conectado el lector de huellas U 4500 y se necesita la presencia del usuario que se esté registrando en el momento, al dar clic en registrar huella se abre la ventana que se aprecia en la *Figura 17* con las instrucciones para el registro de una nueva huella.

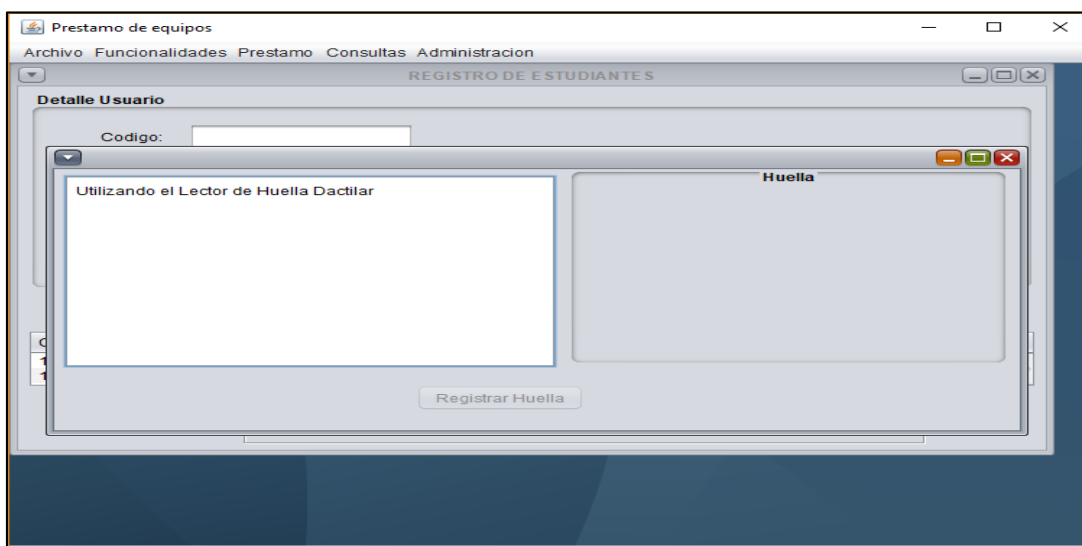


Figura 17. Registro de Huella

Para el registro de una nueva huella es necesario la captura de 4 muestras, el sistema es automático y va tomando la captura con solo colocar el dedo en el lector; en la ventana se va mostrando el estado y la cantidad de muestras tomadas con éxito, cuando las cuatro muestras han sido tomadas el sistema habilito el botón de “Registrar Huella”.

4.2.2.2. Registro de equipos

En esta ventana es similar a la anterior y encontraremos opciones equivalentes; los datos de registro de equipos fueron extraídos de la toma física de inventarios. Dado que el laboratorio cuenta la toma física de inventario esto facilitara alimentar la información en la base de datos; como un extra, a cada equipo se le puede asignar un tipo de equipo, lo que facilita llevar un control sobre la cantidad de equipos disponibles para préstamo. Para la selección de un tipo de equipo se es necesario su previa creación.

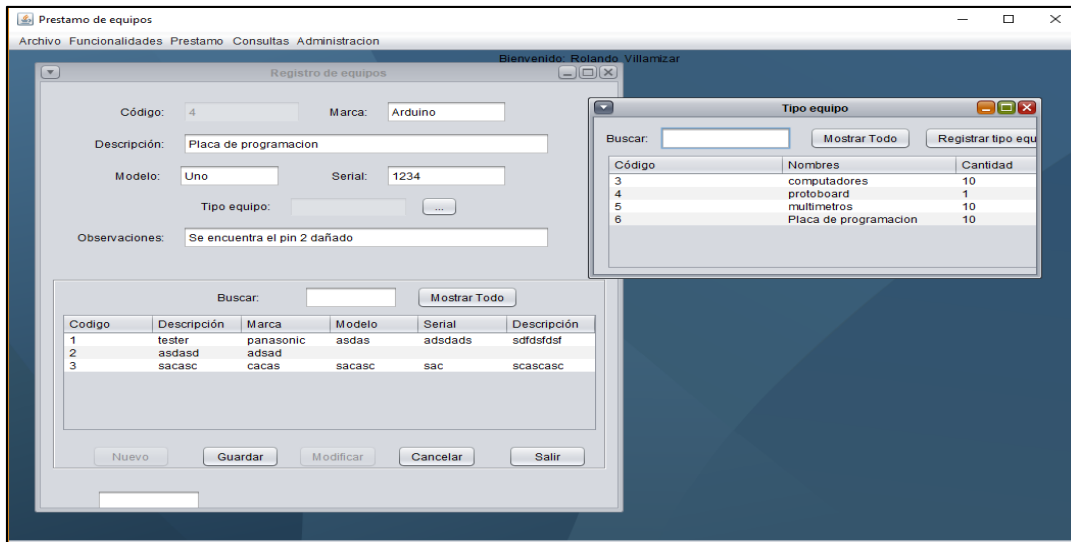


Figura 18. Registro de un nuevo equipo

Como se observa en la Figura 18 los datos manejados en la tabla de registro de equipos, en campo de modelo y serial no se colocó restricción con respecto a los caracteres, ya que en el mercado hay seriales que son alfanuméricos.

4.2.2.3. Tipo de Equipo

Esta opción permite el registro de un nuevo tipo, o la modificación de este, los datos manejados son, código, descripción y cantidad como observamos en la Figura 19. El código que se genera automáticamente con la creación de un nuevo tipo, una descripción del tipo de equipo y la cantidad de equipos que cumplen con la descripción, esta última es de importancia, ya que esta permite a los usuarios ver cuantos equipos están disponibles al momento de una solicitud.

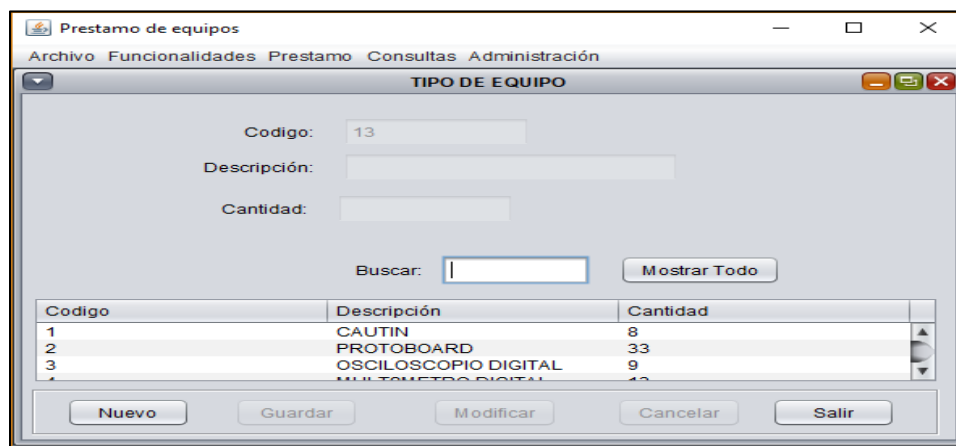


Figura 19. Registro tipo de Equipo



4.2.3. Prestamos

Como su nombre indica, permite realizar prestamos desde el rol – administrador-, esta opción tiene como finalidad de agilizar el proceso en casos que el estudiante tenga un inconveniente con el rol –estudiante-, o casos que se requieran añadir préstamos que por algún motivo no hayan podido ser registrados con anterioridad; también facilita al administrador tomar las solicitudes realizadas de forma verbal.

4.2.3.1. Inicio Sesión

Existen dos opciones para realizar el préstamo, una ingresando el documento y una segunda opción con ayuda del lector de huella del identificador dactilar que agilizará el ingreso del estudiante al aplicativo, esta ventana de inicio la podemos apreciar en la Figura 20. Ésta función comparte código en los roles de usuario y administrador por lo que el proceso realizado será igual

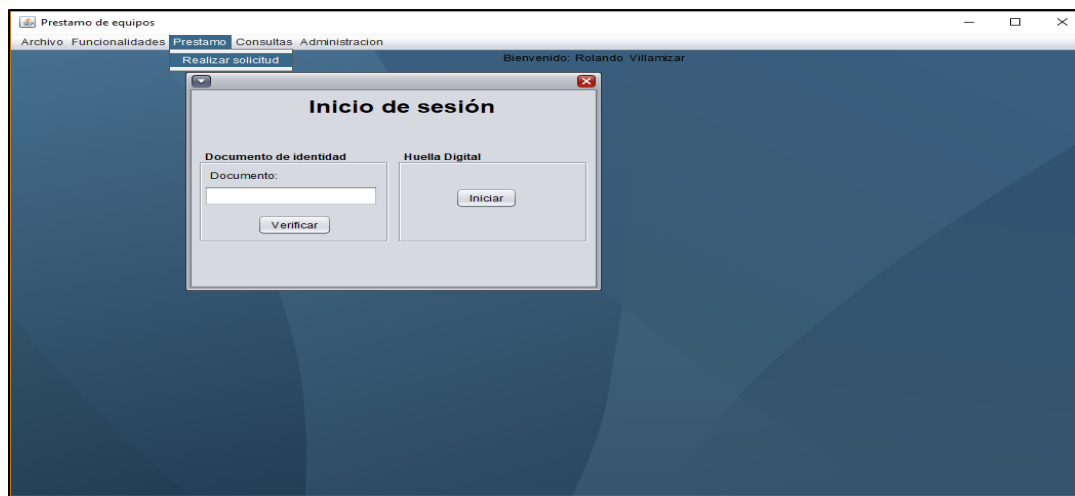


Figura 20. Solicitud de préstamo Rol –administrador-

4.2.3.2. Proceso de solicitud

La siguiente Figura 21 muestra que los datos del estudiante permanecen en la base de datos una vez son ingresados por lo que aparecerán por defecto. Para realizar una solicitud el requisito mínimo es la selección de un equipo; la adición de un equipo para préstamo se hace de forma sencilla, dando clic en el botón “Agregar Equipo”

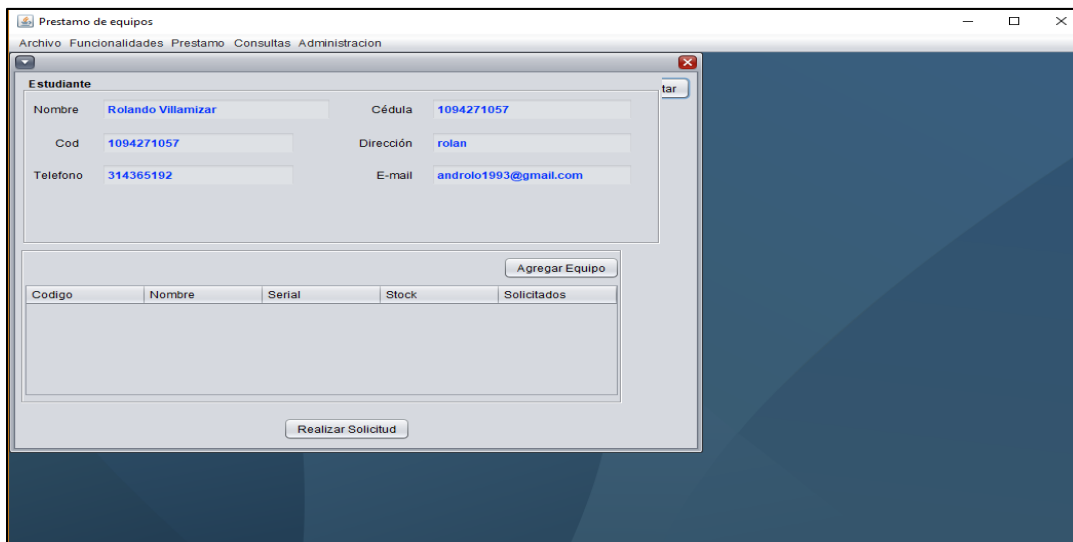


Figura 21. Realizar una solicitud rol –administrador-

La siguiente ventana ver Figura 22, muestra la lista de equipos, la cual tiene la opción de buscar por descripción, mostrando las coincidencias del nombre del equipo buscado.

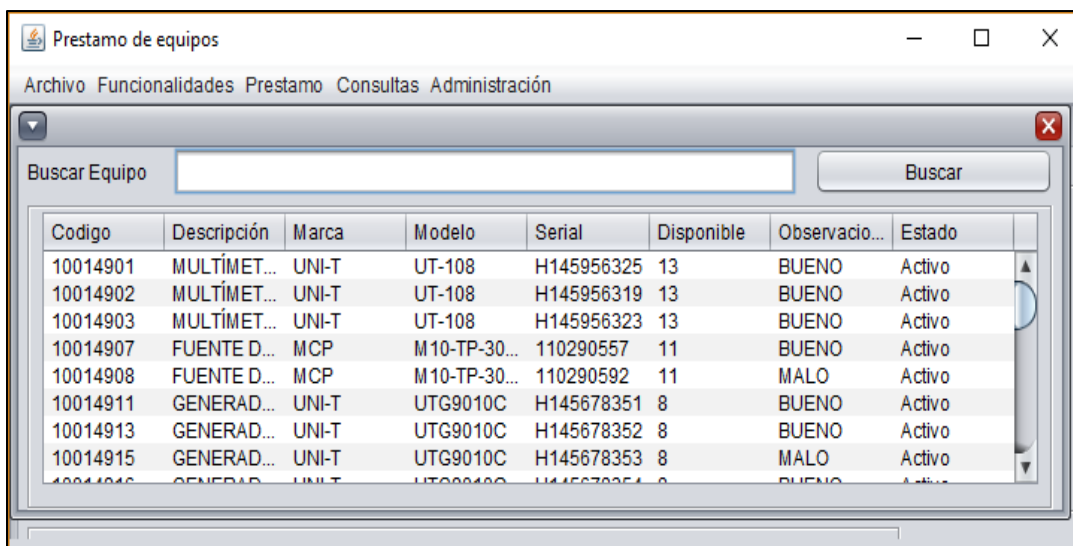


Figura 22. Adición de un equipo solicitud rol –administrador-

Para la adición de un equipo, se debe dar clic derecho sobre la selección del equipo y elegir la opción “Enviar datos”. Finalmente, seleccionados los equipos, presionamos el botón “realizar Solicitud” y la solicitud de préstamo quedara registrada en sistema y queda en estado de espera.



4.2.4. Consultas

En este apartado se tiene como finalidad la revisión de datos; aquí se pueden encontrar los datos de los estudiantes inscritos en el sistema o el listado de equipos. También se cuenta con la función de revisar las solicitudes realizadas por estudiantes considerándose una de las funciones más relevantes del aplicativo.

4.2.4.1. Estudiantes

Permite la visualización de los estudiantes, mostrando la cantidad de registros. El recuadro inferior cantidad de registros que se ve en la

Figura 23, muestra la cantidad de los registros que se visualizan en la tabla actualmente, al realizar una búsqueda el valor mostrado será la cantidad relacionada con la coincidencia de la búsqueda.



Figura 23. Estudiantes Inscritos en el sistema de préstamo

4.2.4.2. Consulta de Equipos

En este apartado la ventana de consulta de equipos nos muestra los equipos que están actualmente registrados, como se ve en la Figura 24, en la tabla se muestra la cantidad total que hay, también permite una búsqueda por la descripción de equipo, filtrando los equipos que coincidan con el valor de búsqueda.



Figura 24. Consulta de equipos

4.2.4.3. Solicitudes

La opción “consultar solicitudes” permite al administrador aprobar, recibir y rechazar solicitudes de los estudiantes, el sistema muestra por defecto en orden de fechas, de las más recientes a las más antiguas como se observa en la Figura 25, también se cuenta con la opción de realizar búsqueda por, cedula, fecha y por huella esta última le permite al administrador identificar al estudiante de forma rápida al momento de realizar un préstamo.

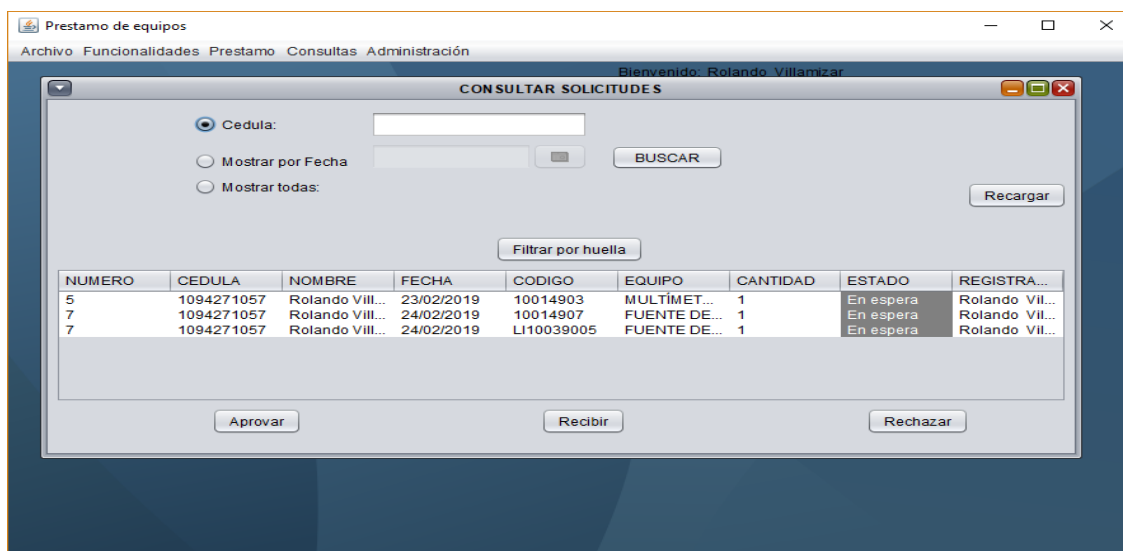


Figura 25. Consulta y control de solicitudes rol-administrador-



Para ver con más detalle una solicitud, el sistema tiene algunas opciones adicionales, al dar clic derecho sobre una de las solicitudes, se despliegan las opciones de “ver detalle” y “eliminar”; la primera opción hace visible una ventana emergente con la información detallada del equipo solicitado y la segunda lo elimina.

Las solicitudes en el sistema se muestran por equipo, con el fin de que, si es necesario eliminar o rechazar un equipo de la solicitud, evitando una nueva solicitud al usuario.

4.3. Rol –Estudiante-

El rol –estudiante- cuenta algunas similitudes con el rol –administrador-, ya que el diseño de la interfaz y funciones se trabajó bajo el mismo código, la diferencia radica en que el código para el rol –usuario- solo tiene funciones de conexión con la base de datos, y la función de solicitudes.

Para el uso del aplicativo el estudiante no tiene necesidad de ingresar usuario y contraseña. El ingreso a la opción “realizar solicitud” del aplicativo cuenta con dos opciones de inicio, la primera digitando el documento de identidad, y la segunda opción que es identificador dactilar; esta última opción solo puede ser usada si se cuenta con el lector de huellas.

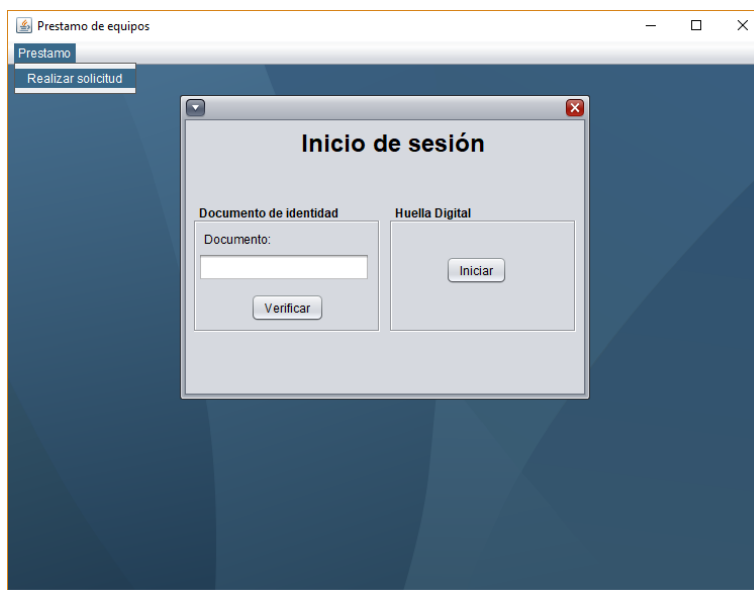


Figura 26. Inicio de sesión para préstamos rol-estudiante-



Al iniciar el sistema se abrirá otra ventana la cual permitirá al estudiante realizar la solicitud de los equipos, como se mencionó anteriormente la interfaz de solicitud de equipos es similar al rol –administrador-, contando con las mismas funciones de cómo se agrega un equipo a la solicitud.

En la siguiente imagen se puede apreciar la ventana de una solicitud, ya con un equipo agregado, y encontraremos un botón que nos permitirá quitar un equipo de la solicitud, cabe mencionar que realizada la solicitud ya no se puede modificar y en caso de que el usuario no dese uno de los equipos solicitados el administrador en momento de aprobar la solicitud puede rechazar de la solicitud el equipo.

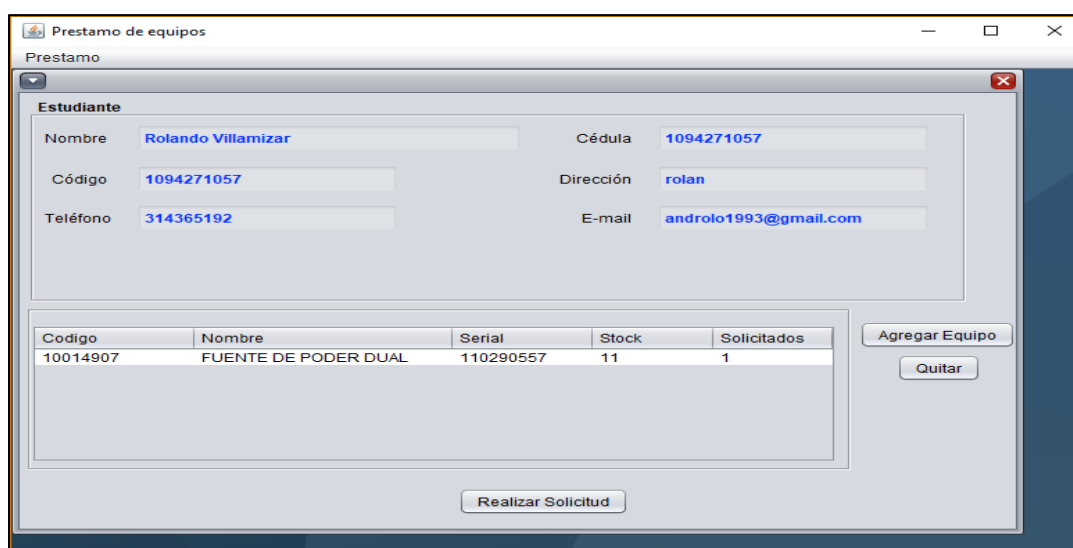


Figura 27. Ventana de solicitud de equipos rol-estudiante-

Como se observa en la ventana se ven los equipos que se van agregando, con la idea del que el usuario –estudiante- pueda ver con detalle la solicitud que este realiza. El cual puede agregar más equipos a la solicitud, o quitarlo según sea el caso.

4.4. Pruebas del aplicativo

Se realiza una prueba general, donde se desea poner a prueba la integración de todas las funciones, para eso se alimentó la base de datos con información de equipos y usuarios con el fin de realizar varias solicitudes, de esta prueba se revisa el procesamiento de las solicitudes, y ver el comportamiento del aplicativo como son aprobar, rechazar o eliminar;

Las pruebas tienen como objetivo observar el comportamiento de la función que realiza el conteo de la cantidad de equipos que hay por tipo de equipo. A continuación, se detalla el proceso.

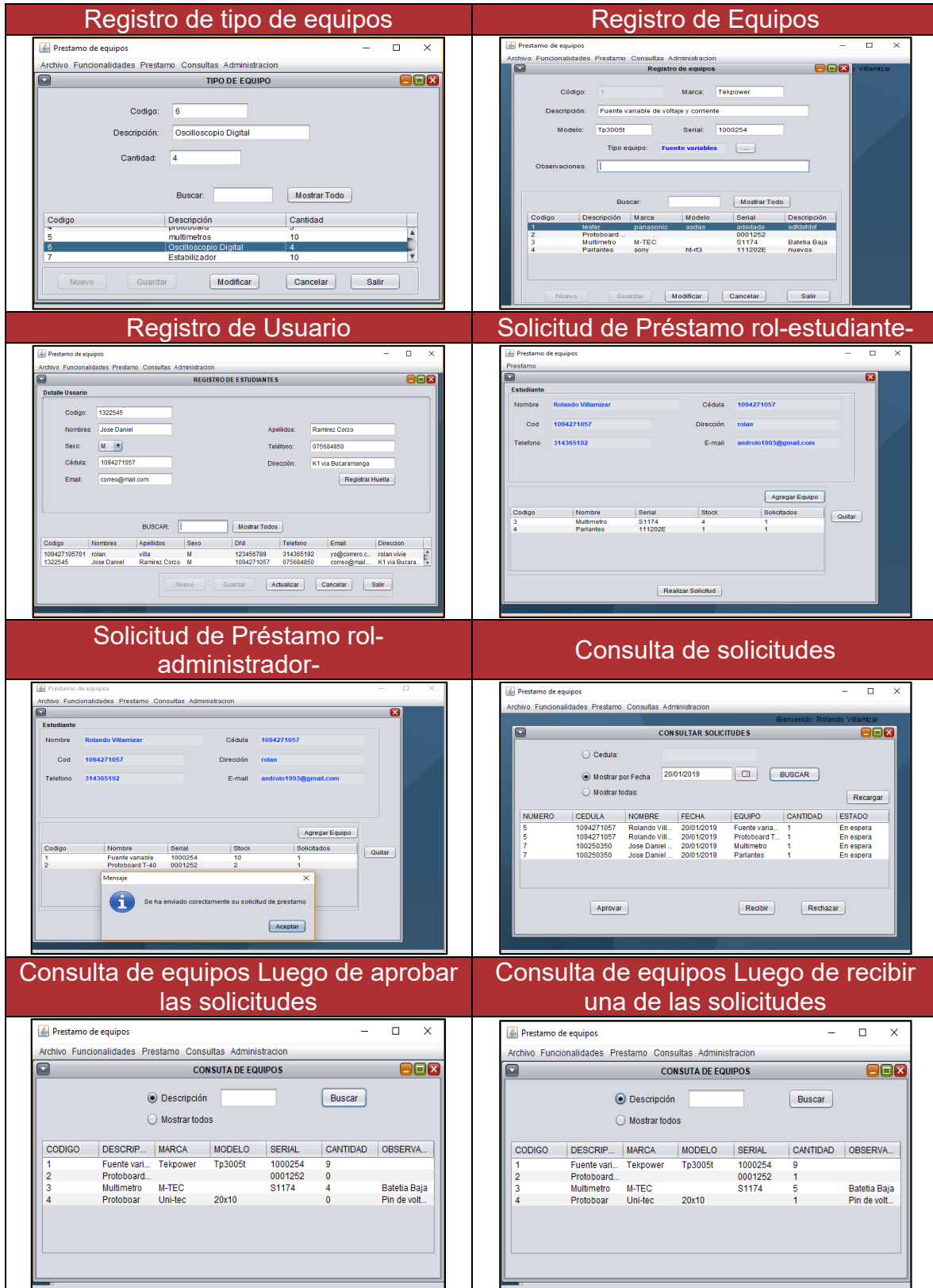


Figura 28. a. Registro de un nuevo tipo equipo b. Registro de equipo c. Registro de usuario d. Solicitud de préstamo rol – estudiante- e. Solicitud de préstamo rol –administrador- f. Consulta de solicitudes realizadas g. Consulta de equipos luego de aprobar solicitudes h. Consulta de equipos luego de recibir las solicitudes



Como vemos en la Figura 28, una serie de imágenes, se realizó el proceso de inscripción de 4 equipos, y 2 usuarios, para el registro de equipos se realizó el registro de varios tipos de equipos con el fin de relacionarlos al momento de registrar un equipo. Teniendo los nuevos registros se procede a realizar la prueba de solicitudes, una desde el rol *-administrativo-* y la otra del rol *-estudiante-*, con el fin de validar la función de solicitud desde los dos roles, como se aprecia en las imágenes a cada usuario se le registro dos equipos en los pedidos, estos pedidos tienen en común la solicitud de un protoboard, con el fin de visualizar en la consulta de equipos la disponibilidad

De esta prueba se destaca el funcionamiento para las solicitudes, permitiendo que tanto el administrador o un estudiante puedan realizarlas. Se aprecia que la función de solicitud desde el rol *-estudiante-* no están limitada con la conexión de la base de datos, también se destaca que el aplicativo se ejecuta sin problema desde cualquier computador que tenga instalado Java, y este en la misma red LAN donde está la base de datos.

4.5. Errores detectados y solución

Durante la programación de las objetos con sus respectivas funciones, se fueron ejecutando a medida de su programación, con el fin de evitar o dar soluciones inmediatas o parciales según el caso, todo esto fue referente al lenguaje, para errores de lógica que se encontraron al momento de ejecutar las pruebas fue necesario realizar modificaciones al código de la función, para llegar a una solución viable, estos errores fueron solucionados a la facilidad de uso de lenguaje y las consultas realizadas.

pero a medida que se integraron las funciones, destacaron algunos errores más complejos, entre ellos se destacaron los errores que se menciona en la siguiente Tabla 3:



PROCESO	ERROR	DETALLE	SOLUCION
Consulta Equipos	Cantidad de equipos no se reflejaba correctamente	Al realizar una consulta de un equipo, el sistema no reflejaba correctamente la disponibilidad de equipos	Crear una nueva entidad que relacionara la cantidad del total de un conjunto de equipos de las misma función
Solicitud/agregar Equipos	No se despliega la ventana para agregar equipos	Al momento de seguir la ruta para agregar un equipo a una nueva solicitud, el sistema marcaba error, y no desplegaba la ventana, error marcado era que el tamaño no coincidida, el error se presentó cuando se agregó más de 4 equipos en la base de datos	La tabla para mostrar los equipos estaba limitada a solo 4 filas, solución, al momento de crear una tabla desde la parte de diseño no colocar el número de filas, para que el sistema automáticamente las asigne según los registros encontrados
Solicitud/agregar Equipos	Variable no admite el tipo de dato	Al momento de solicitar los equipos para una solicitud la ventana de agregar los equipos no se desplegaba, en los registros de ejecución del programa, marcaba que la variable tipo Null no coincidía.	Al revisar la base de datos en busca de variables que eran tipo Null, se encontró que al crear un tipo de equipo nuevo, un valor no se asignaba y colocaba el campo en tipo Null, Fue necesario cambiar las propiedades de la celda en la base de datos.

Tabla 3. Errores destacados y solución



4.6. Otros Softwares presentes en la universidad

4.6.1. Academusoft

La Universidad de Pamplona cuenta con el centro soporte tecnológico CIADTI, este centro tiene implementado en las bibliotecas el aplicativo ACDEMUSOFT, el cual permite el funcionamiento del servicio de consultas, reserva, préstamos y devoluciones de activos y recursos físicos pertenecientes a la institución.

El servicio de este aplicativo da apoyo a la biblioteca en el registro de libros y publicaciones, consulta de materiales digitalizados, en el procedimiento de préstamo de libros y publicaciones a través de la web,

4.6.1.1. Ventajas que presenta ante el sistema de préstamos desarrollado

El sistema implementado por el CIADTI cuenta con diversas funciones, como llevar el control de préstamo de los libros, cuenta con un lector de códigos agilizando el proceso de selección de libro, del modo que el usuario no tiene que realizar una solicitud previa, la base de datos está alojada en un servidor permitiendo a los estudiantes realizar consultas de la existencia de un libro en línea, y a los encargados de la biblioteca revisar la existencia de un libro en una de las sedes. El sistema esta cruzado la lista de matrícula de los estudiantes permitiendo a los estudiantes realizar consultas sin un registro previo.

4.6.1.2. Desventajas ante el sistema de préstamo propuesto

El sistema de préstamo propuesto, permite al usuario final realizar solicitudes, y el sistema le muestra si el insumo está disponible o en préstamo, adicional las observaciones relevantes de los equipos, con el fin de que el usuario pueda tomar una decisión al momento de solicitar un equipo según la información disponible.

El sistema funciona como servidor local permitiendo configurar equipos como clientes; cada ingreso de usuario tiene que ser validado por el lector de huellas



permitiendo agilizar el proceso de inicio de sesión, y da mayor seguridad al momento de realizar las solicitudes, no necesita de una conexión a internet, ya que el sistema funciona en una red LAN.

5. Conclusiones y Recomendaciones

5. Conclusiones.....	54
6. Recomendaciones	55



5. Conclusiones

- El desarrollo de una aplicación para el control de préstamo de equipos en el laboratorio de Electrónica Digital ubicado en el edificio Enrique Rochereaux de la Universidad de Pamplona, puede impactar de forma positiva en proceso de préstamos, y permite aprovechar las herramientas tecnológicas presentes en el laboratorio; de ser implementada de forma correcta, impactará en la eficiencia del proceso, la administración de equipos y la actualización y organización de la información de los mismos.
- Durante el desarrollo del aplicativo, se hizo un estudio sobre el actual sistema de préstamos manejado que permitió desarrollar una solución acorde a las necesidades laboratorio de Electrónica Digital, con funcionalidades de fácil comprensión para los usuarios del mismo.
- Dado que el software (interface) fue creado en JAVA y la base de datos con un administrador de base de datos XAMP, el desarrollo mejoras o correctivos al software original se puede realizar sin afectar los registros guardados en la base de datos.
- El uso de software libre (gratuito) en el desarrollo del aplicativo permite que este sea de uso libre en la institución, lo cual permitiría incluir un equipo cliente adicional, para llevar el control de préstamo en el laboratorio ER106, que este usa los mismos insumos del laboratorio ER105.



6. Recomendaciones

- Realizar respaldos periódicos de la base de datos para salvaguardar la información de los préstamos e inventario de equipos.
- Se recomienda tener cuidado con el manejo de contraseñas de los usuarios, ya que es muy perjudicial el que personas no autorizadas tengan acceso a la información que maneja el sistema.
- Realizar capacitación con los ayudantes de laboratorio que vayan ingresando en nuevos periodos, para un buen uso del sistema.



7. Bibliografía

- [1] T. GROUSSARD, Java 8 Los fundamentos del lenguaje Java, F. J. P. JUAN, Ed., Barcelona: Ediciones ENI, 2014.
- [2] C. Date, «Introduccion a Los Sistemas de Bases de Datos (Spanish Edition),» MEXICO, S.A. ALHAMBRA MEXICANA, 2001, pp. 5-25.
- [3] J. M. P. Gómez, Bases de datos relacionales y modelado de datos, Cimapres: Ediciones Paraninfo, S.A., 2013, pp. 3-10.
- [4] C. Coronel, S. Morris y P. Roa, «Bases de datos Diseño, Implementacion y administración.,» Novena ed., Cengage Learnig, 2011, pp. 10-35.
- [5] G. L. O. Rivera, «Base de datos relacionales teoria y practica,» Medellín Colombia, ITM, 2008, pp. 15-20.
- [6] W. R. Rodriguez Poma, «Originalidad de las huellas dactilares,» 2009. [En línea]. Available: http://www.revistasbolivianas.org.bo/scielo.php?script=sci_arttext&pid=S1997-40442009000200027&lng=es&nrm=iso. [Último acceso: 20 Julio 2018].
- [7] M. Almaguer Ml., «PREVALENCIA DEL PATRÓN DACTILAR EN LOS DERMATOGLIFOS DE ESTUDIANTES,» *Gaceta Internacioal de Ciencias Forenses*, vol. 27, Abril 2018.
- [8] S. Zorita, Reconocimiento Automático Mediante Patrones Biométricos De Huella Dactilar, Madrid: Universiada Politecnica de Madrid, 2003.
- [9] A. M. Vargas, Sistema biometrico de reconocimiento de hullas dactilares en cortrol de acceso y salida, Bofota D.C, 2013.
- [10] A. Angélica Gonzáles, J. Gomez Marin, J. García Sanchez, L. Alsonso Romero, A. L. Sanchez Lázaro y V. Lopez Batista, «Universiada de Salamanca,» 2003. [En línea]. Available: <http://www.lsi.us.es/redmidas/Capitulos/LMD32.pdf>. [Último acceso: Julio 2018].
- [11] R. Juste Meco, Módulo de identificación biométrica mediante huellas dactilares para sistemas empotrados, Madrid: Universidad Carlos III de Madrid. Departamento de Tecnología Electrónica, 2010.
- [12] L. M. Baez Moyano, «CNEISI,» 23 Abril 2007. [En línea]. Available: <http://www.cneisi.frc.utn.edu.ar/viewabstract.php?id=23>. [Último acceso: 28 6 2018].