



Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

Sistema Web para Intercambio de Productos (iTrueque)

Tesis para obtener el título de Licenciada en Computación

Presenta

Evangelina Anzueto Juárez

Asesor

Dr. Pedro Bello López

Junio 2022

Agradecimientos

Mi primer agradecimiento es a Dios, creador del Universo y de la Humanidad, él ha caminado conmigo a lo largo de mi vida y ha manifestado su Gloria en mí, al darme la mejor Madre, el mejor Esposo y el mejor Hijo, además de que en cada momento de mi vida he sentido su presencia, su amor, su protección y su misericordia.

A mi Madre le agradezco y dedico mi proyecto, mi trabajo y toda mi vida, sin ella no estaría escribiendo estos agradecimientos, le agradezco por darme la vida, alimentación, techo, abrigo, educación, formación, valores, amor, motivación y admiración. Sus consejos, aplausos, trabajo y sacrificio hicieron de mí el ser humano que soy. Gracias Infinitas Mamá

Agradezco a los Profesores que tuve a lo largo de mi vida, ellos fueron un punto clave en mi formación académica, cada uno me dejó enseñanzas invaluable. Gracias por tan noble labor que desempeñan día a día educando a miles de estudiantes.

Gracias a mis hermanos y Padre, por haberme tolerado y motivado a continuar con mis estudios.

Gracias Infinitas a mi eterno novio (ahora esposo), pues durante los años que estudié mi carrera, siempre conté con su apoyo, su amor incondicional y su motivación para el cumplimiento de mis metas.

Dedico principalmente este proyecto a mi hijo y le agradezco el amor, paciencia y apoyo que me ha brindado, mostrando disciplina, responsabilidad y autonomía en sus actividades escolares, liberándome tiempo suficiente para sacar adelante este proyecto, y todos mis proyectos laborales. Él es mi motivo para seguir creciendo cada día como profesionista y como ser humano para poder ser su modelo a seguir.

Finalmente agradezco a mis profesores del diplomado por sus grandes enseñanzas y profesionalismo, principalmente gracias a mi asesor por su gran apoyo para la realización de mi proyecto de tesis y mi proceso de titulación.

Tabla de contenido

1. Introducción.....	5
1.1 Antecedentes.....	5
1.2 Planteamiento	6
Capítulo 2. Marco Teórico.....	8
2.1 Metodología RUP.....	8
2.2 UML (Lenguaje Unificado de Modelado)	9
2.3 Base de datos.....	9
2.4 SQL (Lenguaje de Consulta Estructurada).....	10
2.5 Arquitectura de Desarrollo.....	11
2.6 Integración de Herramientas para la plataforma de desarrollo	12
Capítulo 3. Análisis y Diseño del Sistema	13
3.1 Análisis del Sistema	13
3.2 Diseño del sistema.....	15
3.2.1 Descripción general del sistema	15
3.2.1.1 Arquitectura Proceso General.....	15
3.2.1.2 Componentes de la Aplicación.....	16
3.2.2 Diseño del sistema con UML	19
3.2.2.1 Módulo de Usuarios	19
3.2.2.1.1 Diagrama General de Casos de Uso	19
3.2.2.1.2 Diagrama General de Clases	19
3.2.2.1.3 Tablas Descriptivas de Casos de Uso Registro Usuario / Acceso al Sistema / Cerrar Sesión	20
3.2.2.2 Módulo de Productos.....	22
3.2.2.3 Módulo de Trueques	24
3.2.3 Diseño de Pantallas Principales del Prototipo de iTrueque	25
3.2.3.1 Diseño Pantallas Modulo Usuarios	25
3.2.3.2 Diseño Pantallas Módulo Productos.....	28
3.2.3.3 Diseño Pantallas Módulo Trueques	30
3.2.4 Diseño de Base de Datos iTrueque	33
Capítulo 4. Implementación y Pruebas del Sistema.....	35
4.1 Definición de la arquitectura.....	35
4.1.1 Integración de herramientas para la plataforma de desarrollo iTrueque.....	35
4.1.2 Plataforma de Desarrollo (WampServer)	35

4.2 Desarrollo de la Base de Datos	36
4.2.1 Entidad Usuarios	36
4.2.2 Entidad Productos	37
4.2.3 Entidad Cambios	38
4.2.4 Llaves Primarias, Llaves Foráneas y Relaciones entre Entidades	39
4.3 Pruebas e Implementación de las Principales Funciones del prototipo de iTrueque.	40
4.3.1 Funciones Internauta	40
4.3.1.1 Página Principal (Menú Internauta)	40
4.3.1.2 Consultar Productos	42
4.3.1.3 Registrarse como Usuario del Sistema	43
4.3.2 Funciones del Usuario Registrado / Usuario Administrador	46
4.3.2.1 Acceso a iTrueque con cuenta de usuario registrado	46
4.3.2.2 Agregar productos nuevos para trueque (Función Nuevos)	49
4.3.2.3 Consulta, Modificación y Baja de Productos (Función Productos)	52
4.3.2.4 Crear y Enviar Solicitudes de Trueque (Función Intercambiar)	56
4.3.2.5 Consultar y Aceptar /Rechazar Solicitudes de Trueque de otros usuarios (Función solicitudes)	60
4.3.2.5.1 Consultar Solicitudes de Trueque de otros usuarios	60
4.3.2.5.2 Aceptar Solicitud de Trueque (“Aceptar cambio”)	62
4.3.2.5.3 Rechazar Solicitud de Trueque (“rechazar cambio”)	63
4.4 Poner en Marcha el Prototipo de iTrueque	65
5. Conclusión	66
5.1 Trabajo Futuro	66
6. Referencias.	68
6.1 Bibliografía	68
6.2 Mesografía	68

1. Introducción

El trueque es una práctica milenaria que apareció cuando las personas necesitaban cosas que no tenían y para obtenerlas las intercambiaban por cosas que sí tenían. Más tarde apareció el dinero con lo que las cosas se intercambiaban por el dinero y luego el dinero por las cosas. [15]

La idea de hacer un Sistema de Intercambio de Productos (Trueque), surge al observar el auge de las aplicaciones web, las apps móviles, y las redes sociales y se identifica que se puede dar solución a problemas tales como el de la falta de poder adquisitivo debido la crisis financiera en México provocada por la pandemia y algunos otros factores que provocan que algunas personas no tengan acceso a comprar todos los productos que necesitan, pero tiene productos en buen estado que ya no ocupan y que podrían intercambiar por algún otro producto que les esté haciendo falta con personas que por el contrario tienen la posibilidad de realizar muchas compras, la mayoría de productos no indispensables o también tienen productos que no utilizan y que acumulan a modo de estorbo o terminan siendo basura, provocando con esto último un impacto muy negativo para el planeta ya que mientras más productos se produzcan se utilizan más materia prima (otros productos), entre más productos se compren más basura se genera y más contaminación ambiental y deterioro del planeta.

1.1 Antecedentes

Como antecedentes se revisaron algunas aplicaciones web existentes para intercambio de productos entre las cuales se encuentran las siguientes:

 **Hula Trading App** <https://hula.trading/es/>


Una plataforma confiable que permite a los usuarios intercambiar productos que tienen y no utilizan por productos que desean que les sobran a otras personas, además esta app permite conectar a las personas que quieren intercambiar cosas.

La funcionalidad básica está disponible de forma gratuita y para desbloquear funciones más avanzadas es a través de compras dentro de la aplicación. Además, permite tomar un video de las cosas en tiempo real para que todos conozcan el estado del producto antes de obtenerlo. [16]

 **GivU App mexicana para Trueque de Productos.**

<https://laverdadnoticias.com/tecnologia/Tienes-cosas-que-ya-no-usas-o-ya-no-quieres-Haz-trueque-con-esta-app-mexicana-20200127-0297.html>

Es una app mexicana cuya intención es revolucionar la forma de adquirir productos nuevos a través de intercambios online, su funcionamiento es muy sencillo: instalar la aplicación, crear una cuenta o conectarse con Facebook u otras redes sociales, encontrar un producto y tener a la mano algún otro producto que sea apto para ser intercambiado por el artículo de interés. [17]

 **Truekeo, App para trueque de productos y servicios** <http://www.truekeo.com/>,
2017 truekeo.com - Todos derechos reservados | Website by Pulpo

Aplicación destinada a apoyar el trueque y el rehusó, intercambiar objetos, conocimientos y/o servicios por otros objetos, conocimientos y servicios de otras personas. [18]

1.2 Planteamiento

Derivado de lo anterior, se plantea crear un sistema web para la publicación de productos de un grupo de personas para ser intercambiados por otros productos al que llamaremos **iTrueque**.

Se creará una aplicación Web de fácil acceso, amigable e intuitiva para el intercambio de productos considerando los siguientes elementos:

- Interfaz gráfica para el intercambio de productos con acceso restringido (Usuario/Contraseña).
- Establecer contacto entre usuarios para intercambio de productos vía correo electrónico.
- Manual de usuario del funcionamiento del sistema.
- Código fuente del sistema completo.

Para la construcción del **Sistema de Intercambio Web** se analizarán los requerimientos y alcances necesarios para poder efectuar un diseño e implementación exitosos.

El **Sistema de Intercambio Web** deberá permitir al usuario:

- Poder ofertar su producto a intercambiar, especificando características del producto tales como, caducidad, percedero, ubicación del producto, área geográfica de intercambio, etc.
- Debe presentar un listado de productos ofertados por otros usuarios que permitan al usuario poder seleccionar algunos para intercambiar por su producto, generando un listado por prioridad de trueque.

Se plantea crear un prototipo funcional que cumpla con las siguientes características:

- Análisis y diseño del sistema usando la metodología de software del proceso unificado. [11][14]
- Diseñar [1][5] e implementar la base de datos [5] usando el modelo relacional con MySQL.
- Implementar el sistema usando PHP [4] y su conexión con MySQL [5] para crear un prototipo funcional.
- Agregar seguridad al prototipo funcional y realizar las pruebas necesarias para su puesta en marcha. [12]

Capítulo 2. Marco Teórico

A continuación, se describe los conceptos teóricos metodológicos básicos para el desarrollo del proyecto.

2.1 Metodología RUP

Para la construcción del Sistema de Intercambio de Productos iTrueque se utiliza la metodología Rational Unified Process o Proceso Unificado de Racional (RUP), es una metodología para el desarrollo de Software de calidad cuya característica es ser iterativo e incremental, está centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos tangibles como los casos de uso, diagramas de clases, diagramas de secuencia, diagramas de procesos además de roles, los cuales corresponden al papel que desempeña cada persona en un proceso, en un momento determinado, también una persona puede desempeñar más de un rol a lo largo de un proceso. [2][11]

En el ciclo de vida **RUP** se establecen tareas en 4 fases secuenciales en las cuales se realizan varias iteraciones en número variable, en donde al terminar cada fase se realiza una evaluación para determinar si se ha cumplido o no con los objetivos de ésta [2][14].

Las fases son:

Fase de inicio

En donde se define el modelo del negocio y el alcance del proyecto. Se identifican todos los actores y Casos de Uso, además de que se diseñan los Casos de Uso más esenciales (aproximadamente el 20% del modelo completo). Se desarrolla, un plan de negocio para determinar que recursos deben ser asignados al proyecto. [11] [13]

Fase de elaboración

El propósito de esta fase es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto eliminando los mayores riesgos. En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final. Este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio. [11] [13]

Fase de construcción

Durante esta fase todos los componentes, características y requisitos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto. [11] [13]

Fase de transición

La finalidad de esta fase es poner el producto en manos de los usuarios finales, por lo que se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, capacitar a los usuarios en el manejo del producto, y tareas relacionadas con ajustes, configuración, instalación y facilidad de uso del producto. [11] [13]

2.2 UML (Lenguaje Unificado de Modelado)

Para las diferentes etapas de la construcción del Sistema **iTrueque** se crearán modelos de análisis y diseño **UML**, el cual es un lenguaje estándar de modelado de sistemas que se ha adoptado a nivel internacional por diversos organismos y empresas para crear esquemas, diagramas y documentación relativa a los desarrollos de software

Como **RUP** es un proceso de desarrollo orientado a objetos basado en componentes, utiliza herramientas **UML** para definir una forma eficiente de trabajar que permite crear esquemas, diagramas y documentación referente a los desarrollos de software.

UML es un modelado independiente del lenguaje de implementación, de tal forma que los diseños realizados usando **UML** se puedan implementar en cualquier lenguaje que soporte las funciones de **UML** que son principalmente lenguajes orientados a objetos. [23]

UML contiene una amplia variedad de diagramas para visualizar el sistema desde varias perspectivas, los que utilizaremos para el análisis de requerimientos y diseño del sistema serán los **diagramas de casos de uso, diagrama de clases, diagrama de secuencia, etc.**

2.3 Base de datos

Parte de construir un producto de Software como el Sistema de Intercambio de Productos **iTrueque**, corresponde al **almacén de datos**, esto es; las estructuras de memoria en donde se almacenarán la información de usuarios, productos, solicitudes de trueque e intercambios de productos; así como el manejo y administración de estos datos y la interacción de estos con el usuario, de tal forma que es necesario investigar, estudiar, aprender y poner en práctica todo lo referente a Bases de Datos.

Definimos una base de datos como una colección organizada de información estructurada, almacenada electrónicamente en un sistema de computadora. Una Base de Datos normalmente es controlada por un sistema de gestión de base de datos (DBMS), los datos y el DBMS, junto con las aplicaciones que están asociados con ellos, se conocen como un **sistema de base de datos**, que a menudo se reducen a solo base de datos. [4]

Modelo Entidad Relación

Para diseñar y crear una **Base de Datos** se utiliza el **modelo entidad relación**, que es una **herramienta** que permite representar de manera simplificada los componentes de datos que participan en un **proceso de negocio** y el modo en el que estos se relacionan entre sí. [4]

El **Modelo Entidad Relación** tiene tres elementos principales:

- **Entidades** (cada uno de los componentes del proceso de negocio),
- **Atributos** (propiedades de las entidades)
- **Relaciones** (vínculos entre parejas de entidades)

2.4 SQL (Lenguaje de Consulta Estructurada)

Los datos se modelan en filas y columnas en una serie de tablas para que el procesamiento y la consulta de datos sean eficientes. Luego se puede acceder, administrar, modificar, actualizar, controlar y organizar fácilmente los datos. La mayoría de las bases de datos utilizan lenguaje de consulta estructurado (**SQL**) para escribir y consultar datos. [7][8]

Dada su fuerte conexión con la teoría del modelo relacional, SQL es un lenguaje de alto nivel orientado a conjuntos de registros. Esto implica que un solo comando SQL puede equivaler a decenas o cientos de líneas de código que se tendrían que utilizar en un lenguaje de más bajo nivel orientado a registros. Como resultado de lo anterior, SQL permite lograr con mayor rapidez y facilidad la definición y manipulación de los objetos de base de datos, permitiendo alcanzar una mayor eficiencia y productividad en el desarrollo. [10]

SQL es un estándar ANSI, por lo que las bases del lenguaje son las mismas en la mayoría de los motores de bases de datos relacionales. [10]

Algunas funciones de SQL están basadas en el cálculo relacional, algunas en el álgebra relacional que provienen del modelo relacional y otras a ninguno de los dos, sino que son parte de SQL. [10]

SQL posee herramientas para

- Definición de datos o DDL
- Manipulación de datos o DML
- Integridad
- Definición de vistas
- Funciones de agrupamiento
- Control de Transacciones
- Autorización o Control de Acceso
- SQL incorporado y dinámico

2.5 Arquitectura de Desarrollo

La arquitectura seleccionada para el desarrollo de **iTrueque** es una arquitectura de modelo de 3 capas que es el tipo de arquitectura ideal para una aplicación web que gestionara información a almacenar en un servidor de base de datos, en donde tendremos:

1. Capa de presentación

Representada por el cliente (el navegador, explorador o visualizador)

La aplicación cliente corresponde a la **Capa de presentación** y se refiere a la interacción entre el usuario y el software. Su principal responsabilidad es mostrar información al usuario, interpretar los comandos de este y realizar algunas validaciones simples de los datos ingresados. El cliente puede ser cualquier navegador web como **Google Chrome, Firefox, Edge, Opera, etc.**

2. Capa de Negocio

Representada por el servidor Web

La capa de negocio corresponde al Servidor en donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas.

El Servidor Web que se utilizara es Apache, el cual es un popular servidor web multiplataforma de fuente abierta. El trabajo básico de Apache es aceptar solicitudes de clientes (por ejemplo, el navegador web de un visitante) y luego enviar la respuesta a esa solicitud (por ejemplo, los componentes de la página que el visitante quiere ver).

[19]

3. Capa de datos

Representada por el Servidor de Datos.

Esta capa se encargará de gestionar los datos de la aplicación. Será el almacén de todos los datos de la aplicación y entre sus objetivos principales están, asegurar la disponibilidad de los datos y organizar los datos de tal forma que el acceso a éstos sea lo más rápido posible. El Servidor de Datos para **iTrueque** será MySQL.

La Figura 2.1 presentada a continuación muestra la arquitectura Web Cliente-Servidor de 3 capas que se utilizara para la implementación de **iTrueque**.

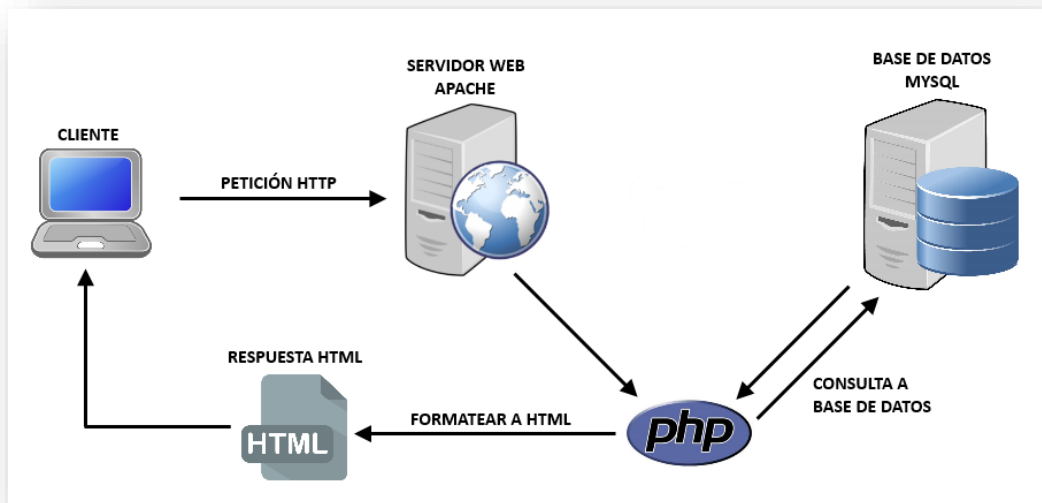


Figura 2.1 Arquitectura Aplicación Web Cliente-Servidor de 3 Capas

<http://diymakers.es/wp-content/uploads/2014/08/PROCEDIMIENTOWEBSEVER.png>

2.6 Integración de Herramientas para la plataforma de desarrollo

Como herramientas adicionales para el desarrollo de **iTrueque** del lado del cliente se utilizará el lenguaje de marcado **HTML**, y para el diseño de la presentación el lenguaje **CSS**.

Para el desarrollo y funcionamiento de **iTrueque** se integrarán los lenguajes **HTML**, **PHP**, **CSS** y el sistema de base de datos **MySQL** en donde cada herramienta tiene una función diferente explicada brevemente a continuación. [6][8]

HTML: Lenguaje con el que se define el contenido de las páginas web. Básicamente se trata de un conjunto de etiquetas que sirven para definir el texto y otros elementos que compondrán una página web. [9]

CSS: Es un lenguaje de diseño gráfico que nos permite definir la presentación de un determinado documento estructurado. Si con HTML definimos que es cada elemento y donde se sitúa, **con CSS vamos a definir como queremos presentarlo.** [9]

PHP: Es un lenguaje de programación del lado del servidor que nos permite generar documentos HTML de forma dinámica. De este modo podemos generar distintos documentos en función de variables pasadas por el usuario, ya sea desde la propia URL o desde un formulario. [3] [9]

MySQL: Es un sistema de base de datos relacional y su función es la de almacenar y administrar datos de forma perdurable en el tiempo y entre sesiones. [9]

Capítulo 3. Análisis y Diseño del Sistema

Como se plasmó en el documento de Especificación de Requisitos del Software tanto para el análisis como para el diseño de **iTrueque** se utiliza la metodología de software del proceso unificado **RUP (Proceso Unificado de Racional)**, utilizando **UML (Lenguaje Unificado de Modelado)** para el modelado del análisis y diseño del sistema.

3.1 Análisis del Sistema

De acuerdo con la metodología UML para el análisis de requerimientos, se siguen los siguientes pasos mostrados:

- Descripción detallada del sistema
- Análisis textual
- Localización de sustantivos, estados y las relaciones entre estos
- Definición de Actores y Casos de Uso

Los artefactos obtenidos durante el análisis se muestran en las Figuras 3.1 y 3.2 en donde se parte primero de los requerimientos del usuario, en donde a partir de las especificaciones y requisitos se realiza un análisis textual, se determinan los actores (funciones y usuarios), las restricciones, las dependencias y la relaciones entre actores, obteniendo de esta forma una descripción general del Sistema a construir, mostrada en la Figura 3.4

Requerimiento Usuario

Se requiere un sistema web gratuito para intercambio de productos, que permita a cualquier persona internauta buscar productos disponibles para intercambio y en caso de estar interesados, pueda registrarse como usuario del sistema para poder **publicar/ingresar sus productos** para **intercambiar** por otros productos, considerando algunos elementos como la caducidad de productos perecederos, el estado de los productos y el área geográfica para establecer un punto de encuentro. Considerando además la prioridad del trueque, es decir, el tipo de productos por el que se desea intercambiar y el orden de preferencia. El usuario que publica o registra sus productos debe también poder hacer cambios a las características de sus productos y cambiar los otros productos y su prioridad de intercambio. También debe poder eliminar un producto si ya no está interesado en intercambiarlo. El sistema notificara a los usuarios cuando otro usuario esté interesado en intercambiar alguno de sus productos por otro producto y establecerá comunicación entre ambas personas interesadas en realizar el intercambio. Ambos usuarios establecerán contacto y negociaran el trueque y pueden o no finalizarlo con éxito. Debería existir un super usuario que tenga acceso a todas las funciones del sistema, que pueda depurar productos y usuarios que nunca hicieron transacciones de trueque o que llevan mucho tiempo sin hacer transacciones de trueque.

Figura 3.1 Requerimientos iniciales del usuario

Análisis Textual

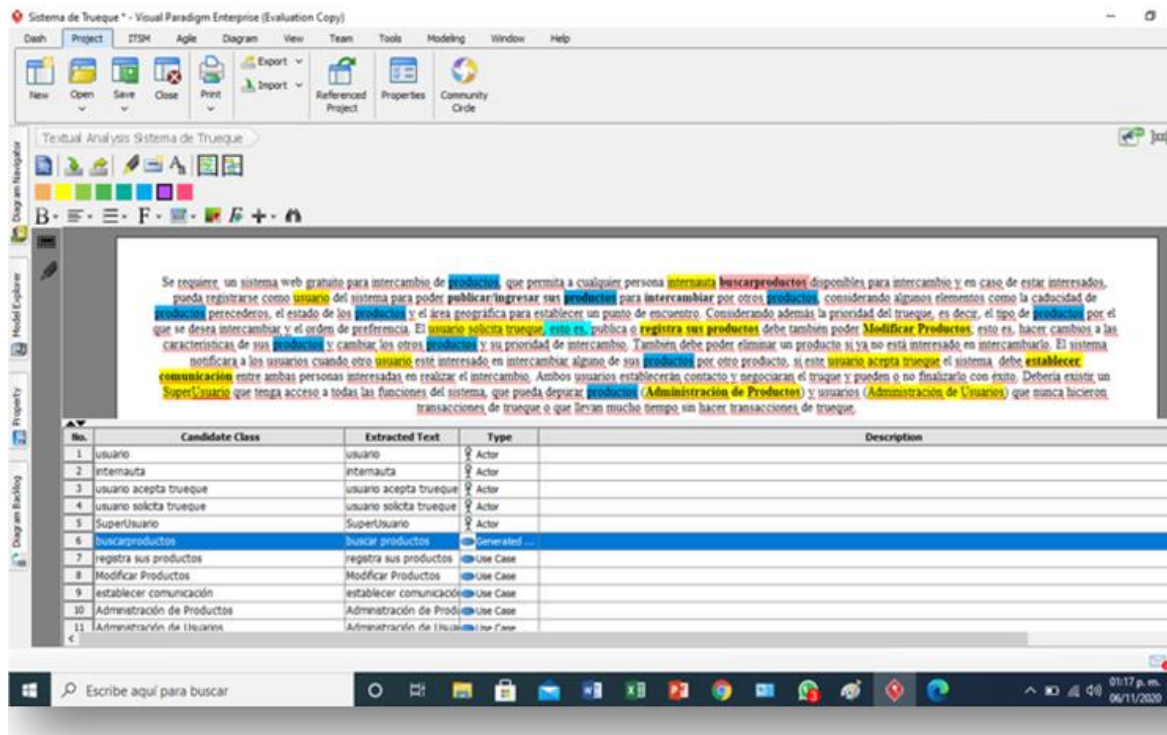


Figura 3.2 Análisis Textual

- Actores : Internauta, Administrador, usuario registrado

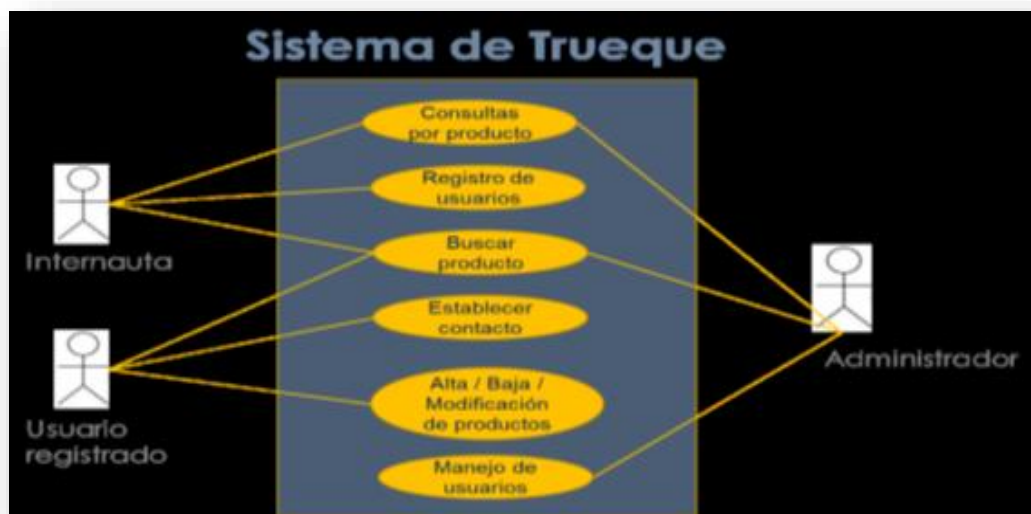


Figura 3.3 Descripción General del Sistema

3.2 Diseño del sistema

Analizados los requerimientos y alcances necesarios del sistema en la especificación de requisitos del software, se efectúa el proceso de diseño de Software en donde se diseña la arquitectura, el diseño de las funciones y los datos del Sistema por medio de diagramas generales y detallados, diagramas de proceso, diagramas de clases y de secuencia, así como la descripción detallada de todos los requisitos funcionales y no funcionales.

3.2.1 Descripción general del sistema

Se describe en forma general los elementos necesarios en el desarrollo del sistema web de intercambio de productos.

3.2.1.1 Arquitectura Proceso General

Se diseña la arquitectura del proceso general del Sistema de Trueque de Productos **iTrueque** como se presenta en la figura 3.5.

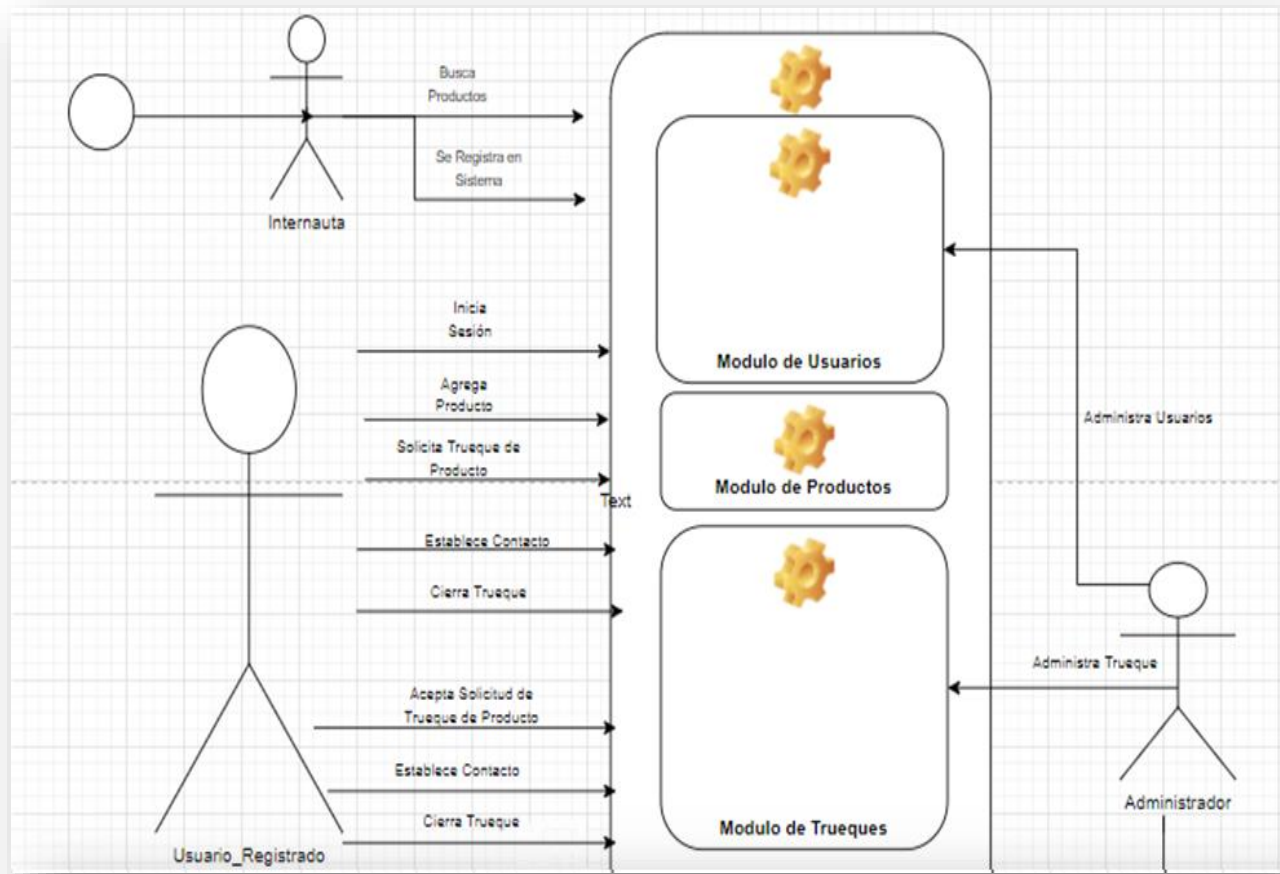


Figura 3.5 Proceso General de iTrueque

3.2.1.2 Componentes de la Aplicación

El Sistema **iTrueque** tendrá los siguientes Módulos:

- **Usuarios**
- **Productos**
- **Trueques**

Módulo Usuarios

El Módulo de Usuarios será utilizado por los Roles de **Internauta**, **Usuario_Registrado** y **Administrador** cada uno con sus respectivos privilegios de acuerdo con la **tabla 3.1**, **tabla 3.2** y **tabla 3.3** presentada a continuación.

Rol	Módulo Usuarios Sistema iTrueque				
	Registra Usuario	Iniciar Sesión	Cerrar Sesión	Cambiar Contraseña	Recuperar Contraseña
Internauta	Cualquier Internauta que navegue en iTrueque solo podrá consultar productos ofertados para trueque por otros usuarios. Si desea registrar sus propios productos y ofertarlos para Trueque a cambio de productos de otro usuario deberá registrarse en iTrueque . De tal forma que tendrá acceso a la función Registrar Usuario y llenado un formulario podrá quedar registrado como Usuario-Registrado para poder operar en el Sistema.				

Tabla 3.1 Modulo Usuarios – Rol Internauta

Rol	Módulo Usuarios Sistema iTrueque				
	Registra Usuario	Iniciar Sesión	Cerrar Sesión	Cambiar Contraseña	Recuperar Contraseña
Usuario_Registrado		Podrá iniciar sesión en iTrueque y podrá operar con los privilegios que le corresponden al Rol Usuario_Registrado . Entre otras cosas podrá, registrar sus productos, Solicitar Trueque de Productos, Establecer Contacto para intercambio con otros usuarios y cerrar transacción de Trueque de Productos.	Acceso a Cerrar Sesión para finalizar sus procesos y su usuario de sesión, función asociada a la seguridad de los datos del usuario.	Función en donde podrá cambiar su contraseña proporcionand o su contraseña anterior	Función en donde podrá recuperar su contraseña siempre y cuando proporcione información de recuperación de contraseña

Tabla 3.2 Modulo Usuarios – Rol Internauta

Rol	Módulo Usuarios Sistema iTrueque				
	Registra Usuario	Iniciar Sesión	Cerrar Sesión	Cambiar Contraseña	Recuperar Contraseña
Administrador		Podrá iniciar sesión en iTrueque y podrá operar con los privilegios que le corresponden al Rol Administrador . Entre otras cosas podrá, agregar/modificar/eliminar productos no truequeados. Agregar/modificar/eliminar Solicitudes de Trueque Abiertas. Agregar/Modificar/Eliminar Ciudades, Estados, Colonias	Acceso a Cerrar Sesión para finalizar sus procesos y su usuario se sesión función asociada a la seguridad de los datos del usuario.	Función en donde podrá cambiar su contraseña, proporcionand o su contraseña anterior	Función en donde podrá recuperar su contraseña siempre y cuando proporcione información de recuperación de contraseña

Tabla 3.3 Modulo Usuarios – Rol Administrador

Módulo Productos

En este módulo se administrarán los productos (Altas, Bajas, Cambios y Consultas) de acuerdo con los privilegios de cada rol.

Usuario Internauta: En el Módulo de Productos este usuario tendrá acceso solo a la búsqueda de Productos Ofertados para Trueque por otros usuarios y con estatus “Libre”.

Usuario Registrado: En el Módulo de Productos este usuario tendrá activas las siguientes funciones:

- **Búsqueda de Productos** pendientes de trueque ofertados por otros usuarios registrados.
- **Consulta de Productos** propios, pendientes de trueque, consulta de productos de otros usuarios pendientes de trueque.
- **Alta de Productos:** Este usuario podrá agregar productos que quiera intercambiar por productos ofertados por otro usuario en donde deberá especificar entre otras cosas, el **nombre del producto**, la **descripción**, **categoría**, **estado** del producto, **Fecha de Caducidad**, **foto** del producto, etc.
- **Modificación de Productos:** El usuario podrá modificar cualquier campo de sus productos siempre y cuando los productos se encuentren con estatus “Libre”, esto es; no asignados a alguna Solicitud de Trueque, (estatus “EnProceso”) ni tampoco ya intercambiados (estatus “Intercambiado”)
- **Baja/Eliminación de Productos.** Esta operación será lógica, solo cambiara el estatus del producto a “Baja”, permitirá al usuario dar de baja sus productos, siempre y cuando el estatus del producto sea “Libre”.

Usuario Administrador: Para el módulo de Productos el usuario Administrador tendrá los mismos privilegios que el Usuario_Registrado y adicional podrá:

- **Buscar / Consultar Productos propios y de otros usuarios.**
- **Modificación de Productos propios y de otros usuarios.**
- **Baja / Eliminación de Productos propios y de otros usuarios.**

Módulo Trueques

Usuario Internauta: No tiene acceso a este módulo.

Usuario Registrado: Este usuario realizara las siguientes operaciones:

- **Solicitud de Trueque de Producto.** El **usuario_ registrado** que va a ofertar su producto para intercambiar por algún otro producto deberá levantar una Solicitud de Trueque en donde seleccionara su producto a ofertar, además de especificar el producto de otro usuario que desea a cambio y quedará la solicitud en espera hasta el usuario dueño del producto candidato a trueque acepte o rechace la solicitud de Trueque. En caso de aceptar el intercambio podrán de establecer contacto ambos usuarios, quedando cerrada con éxito la solicitud de trueque.

Al registrar una nueva solicitud de trueque los productos de ambos usuarios quedaran con estado “**Libre**” y al aceptar el trueque y realizar el intercambio el estado del producto cambiara a “**Intercambiado**”. En caso de rechazar la solicitud el estado del producto regresara a “**Libre**” para que su dueño pueda en un futuro colocarlo en nueva solicitud de trueque.

- **Baja de Solicitudes de Trueque:** En caso de que el usuario registrado no pueda intercambiar su producto, o simplemente lo haya agregado por equivocación o ya no quiera intercambiarlo podrá dar de baja la solicitud y de esta forma el estado del producto regresara a “**Libre**”, liberando el producto para otra solicitud.
- **Aceptar/Rechazar Solicitud de Trueque**

Ambos usuarios involucrados en la solicitud de trueque podrán rechazarla y el sistema devolverá el estado de sus productos de “**EnProceso**” a “**Liberado**”, para que queden listos para futuras nuevas solicitudes de trueque.

Usuario Administrador: Para el módulo de Trueques el usuario Administrador tendrá los mismos privilegios que el Usuario_Registrado y adicional podrá:

- **Buscar / Consultar Trueques propios y de otros usuarios.**
- **Solicitar Trueques propios de otros usuarios.**
- **Baja de Solicitud de Trueques de otros usuarios.**
- **Aceptar / Rechazar Trueques de otros usuarios.**

3.2.2 Diseño del sistema con UML

En este punto se pinta la funcionalidad de los principales módulos que tendrá **iTrueque** utilizando herramientas de modelado UML como los Diagramas de Casos de Uso, Diagrama de Clases y/o Diagramas de Secuencia y Tablas Descriptivas de Casos de Uso con el fin de obtener un diseño claro de las funciones de cada módulo.

3.2.2.1 Módulo de Usuarios

Para el módulo de usuarios, se utiliza el modelado UML para representar el diagrama general de Casos de Uso y el Diagrama general de Clases.

3.2.2.1.1 Diagrama General de Casos de Uso

El Diagrama General Casos de Uso se presenta en la Figura 3.6, en donde se muestran las funciones del Módulo de Usuarios.

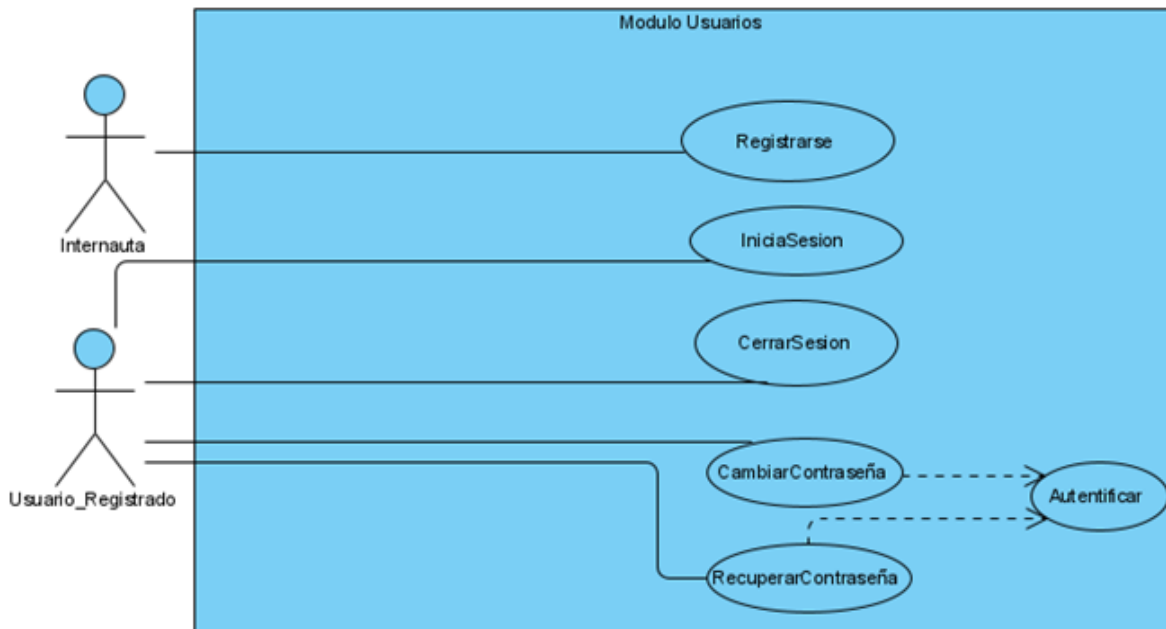


Figura3.6 Diagrama General Casos de Uso Módulo Usuarios de iTrueque

3.2.2.1.2 Diagrama General de Clases

De acuerdo con las funciones de usuarios detectadas durante el análisis se crea el siguiente diagrama de clases correspondiente al módulo usuarios de la Figura 3.7.

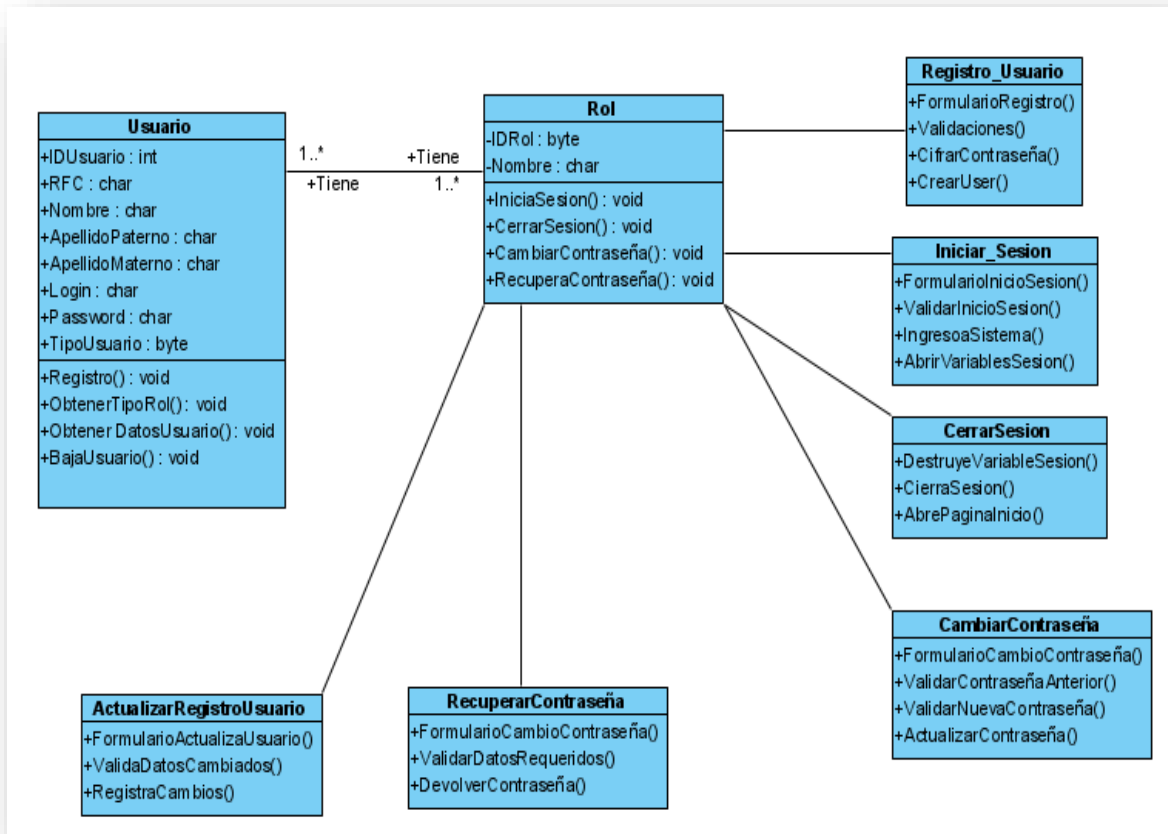


Figura 3.7 Diagrama de clases del módulo Usuarios

3.2.2.1.3 Tablas Descriptivas de Casos de Uso Registro Usuario / Acceso al Sistema / Cerrar Sesión

Tabla Descriptiva Caso de Uso Registro de Usuario

A continuación, se presenta la Tabla 3.4 descriptiva que muestra el flujo del caso de uso **Registro de Usuario**.

Nombre:	Registro de Usuario en Sistema de Trueque	
Actor:	Internauta	
Descripción	Persona que navega en iTrueque interesada en realizar trueque de un producto y para realizar esta operación debe registrarse como usuario del sistema	
Flujo Principal	Eventos Actor	Eventos Sistema
	1) Internauta solicita al sistema opción para Registrarse como Usuario	2) El Sistema muestra formulario de registro

	3) Internauta llena formulario para alta de registro incluyendo usuario y contraseña	4) El sistema valida que los campos requeridos estén correctamente capturados, para el caso de la contraseña valida contraseña segura (al menos una mayúscula, al menos un carácter especial, longitud mínima 8 caracteres, longitud máxima 45 caracteres).
		5) Ya validados todos los datos del formulario de registro de usuario, el sistema guarda el registro en la base de datos de usuario cifrando la contraseña con cifrado SHA1, mediante un Trigger en SQL.
Flujo Alternativo:	Eventos Actor	Eventos Sistema
Precondición:	Internauta navegando en Sistema de Trueque Busca productos de su interés para intercambiar por su producto, para solicitar trueque debe registrarse en iTrueque	
Postcondición:	Internauta Inicia Sesión en iTrueque con su nuevo Usuario validando el acceso y posteriormente podrá dar de alta su producto para poder ofertarlos por medio de una solicitud de trueque.	

Tabla 3.4 Descriptiva Caso de Uso Registro de Usuario

Tabla Descriptiva caso de uso "Acceso al Sistema"

La Tabla descriptiva 3.5 muestra el flujo del caso de uso "Acceso al Sistema"

Nombre:	Acceso Usuario en Sistema de Trueque	
Actor:	Usuario_Registrado/Usuario_Administrador	
Descripción	Persona registrada en iTrueque ya sea con rol de Usuario_Registrado o Usuario_Administrador	
Flujo Principal	Eventos Actor	Eventos Sistema
	1) Usuario ejecuta opción para acceso al Sistema	2) El Sistema muestra formulario para acceso al sistema
	3) Usuario Captura Usuario, Contraseña y ejecuta botón de acceso	4) El sistema valido que los campos requeridos estén correctamente capturados, para el caso de la contraseña aplica cifrado SHA1 y busca usuario y contraseña en la Base de Datos de Usuarios.
		5) Si el usuario y contraseña existen en la Base de Datos, le da acceso al Sistema, presentándole la pantalla principal de usuario de acuerdo al Rol de este. Si el usuario y/o contraseña no existen, limpia campos para que el usuario intente nuevamente. Después de 3 intentos erróneos, cierra la pantalla de inicio de sesión y muestra la página de Inicio.
Flujo Alternativo:	Eventos Actor	Eventos Sistema

Precondición:	Usuario debe estar previamente registrado en el Sistema iTrueque
Postcondición:	Ya con acceso a iTrueque dependiendo del Rol del Usuario, este podrá, Agregar Productos y Solicitar Trueque.

Tabla 3.5 Descripción Caso de Uso Acceso Usuario

Tabla Descriptiva Caso de uso Cerrar Sesión

La Tabla 3.6 muestra el flujo del caso de uso "Cerrar Sesión".

Nombre:	Caso de Uso Cerrar Sesión	
Actor:	Usuario_Registrado/Usuario_Administrador	
Descripción	Usuario de iTrueque ya sea con rol de Usuario_Registrado o Usuario_Administrador, una vez terminada sus tareas en iTrueque debe cerrar su sesión y a modo de seguridad después de 10 minutos sin actividad el sistema cerrara la sesión automáticamente.	
Flujo Principal	Eventos Actor	Eventos Sistema
	1) Usuario ejecuta opción "Cerrar Sesión"	2) El Sistema destruye las variables de sesión activas 3) Muestra Página de Inicio
Flujo Alternativo:	Eventos Actor	Eventos Sistema
Precondición:	Usuario debe estar operando en el Sistema iTrueque	
Postcondición:	Usuario de iTrueque podrá iniciar sesión nuevamente desde la página de Inicio.	

Tabla 3.6 Caso de Uso Cerrar Sesión

3.2.2.2 Módulo de Productos

- **Diagrama de Casos de Uso y Diagrama de Clases para Módulo de Productos**

Siguiendo con el diseño a partir del análisis, se crea el del módulo de productos presentado en la Figura 3.8 que muestra las funciones del módulo de productos mediante un diagrama de casos de uso y la figura 3.9 mediante un diagrama de clases.

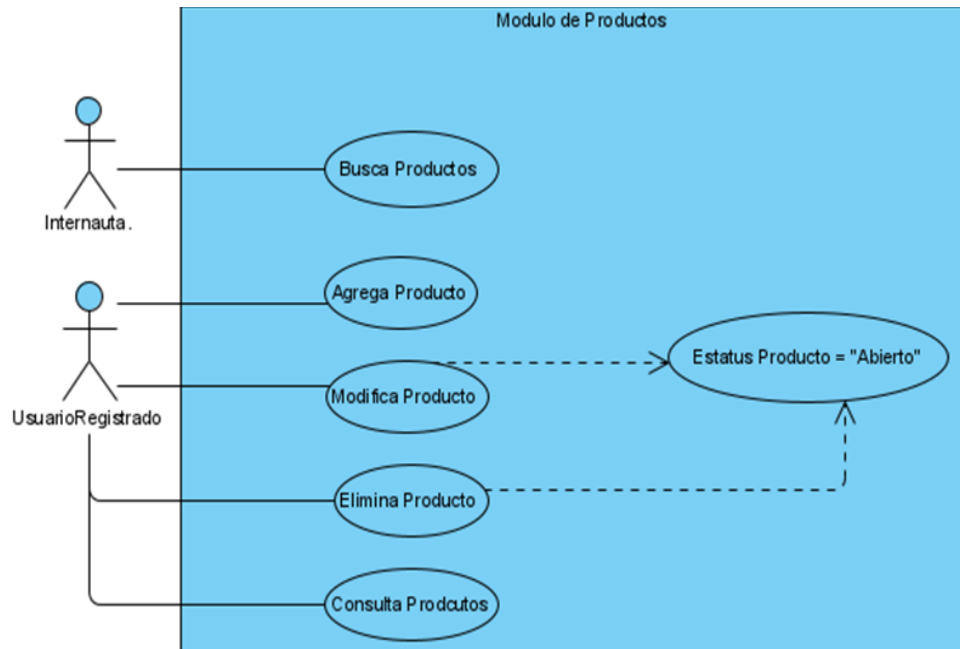


Figura 3.8. Diagrama General de Casos de Uso Módulo Productos

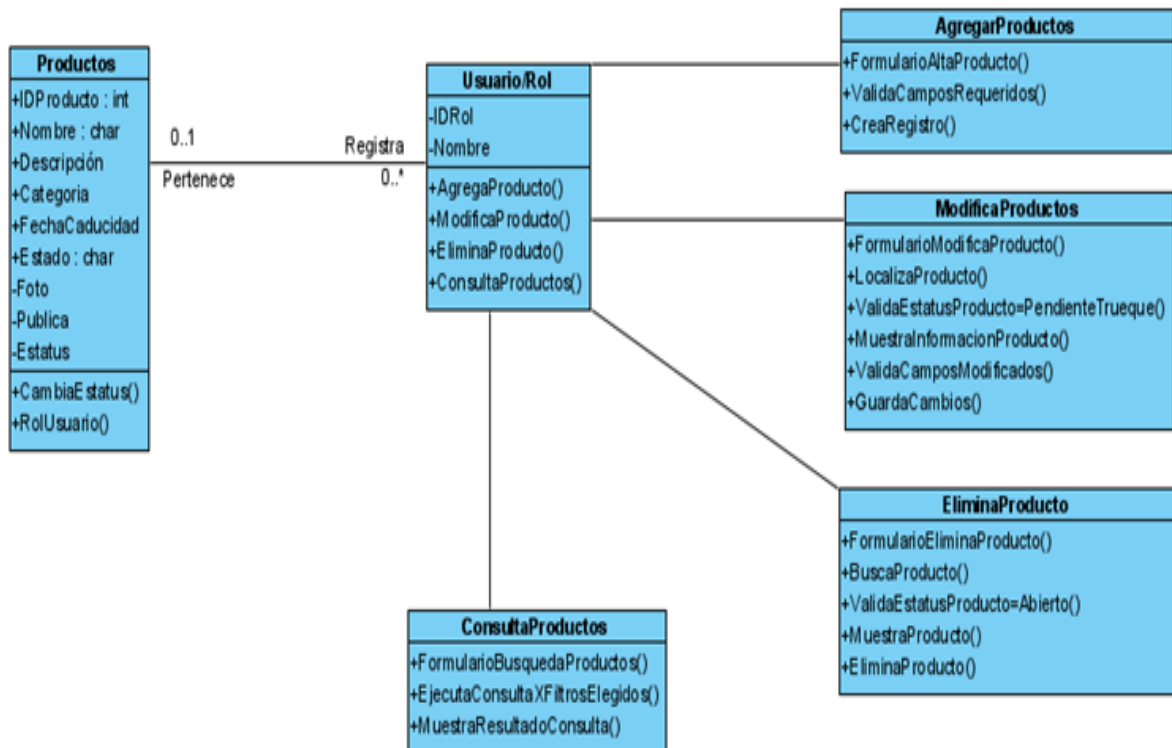


Figura 3.9. Diagrama General de Clases Módulo Productos

3.2.2.3 Módulo de Trueques

El Módulo de Trueques contendrá las funciones principales para poder ofertar e intercambiar productos entre usuarios registrados en **iTrueque**.

3.2.2.3.1 Diagrama de Casos de Uso Módulo de Trueques.

El diagrama de casos de uso de la **Figura 3.10** muestra de manera general las funciones del módulo de Trueques que son Crear Solicitud de Trueque de Productos, Modificar Solicitud de Trueque, Eliminar Solicitud de Trueque, Aceptar Solicitud de Trueque, Confirmar Intercambio, Establecer, Contacto, Efectuar Trueque de Productos y consultar trueques. En este diagrama el **Usuario A** representa al usuario que levanta una solicitud de trueque y el usuario B representa al usuario(s) que aceptan la solicitud de trueque.

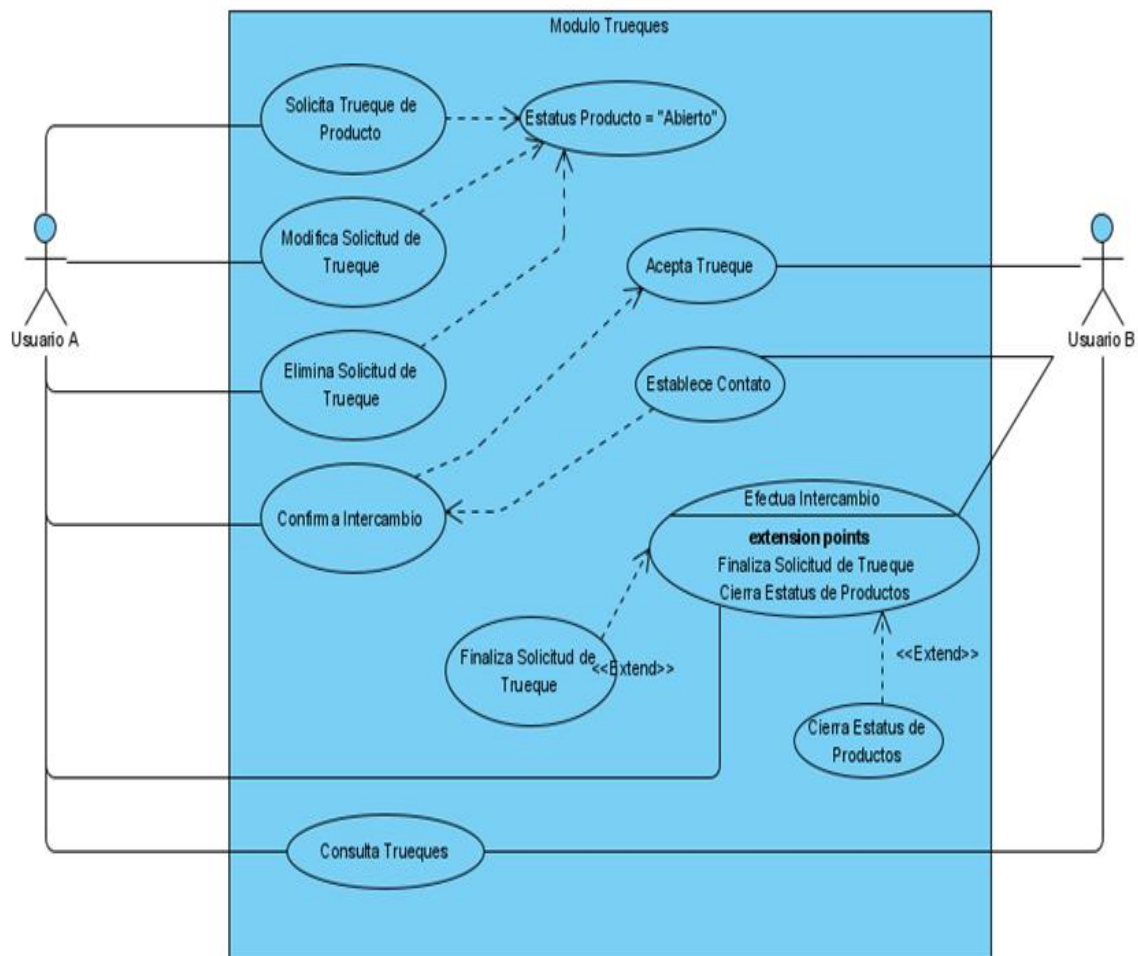


Figura 3.10 Diagrama de Casos de Uso Modulo Trueques

3.2.2.3.2 Diagrama de Clases Módulo Trueques

El Diagrama de Clases de la Figura 3.11 representa las Funciones del Módulo de Trueques y como se interrelacionan estas.

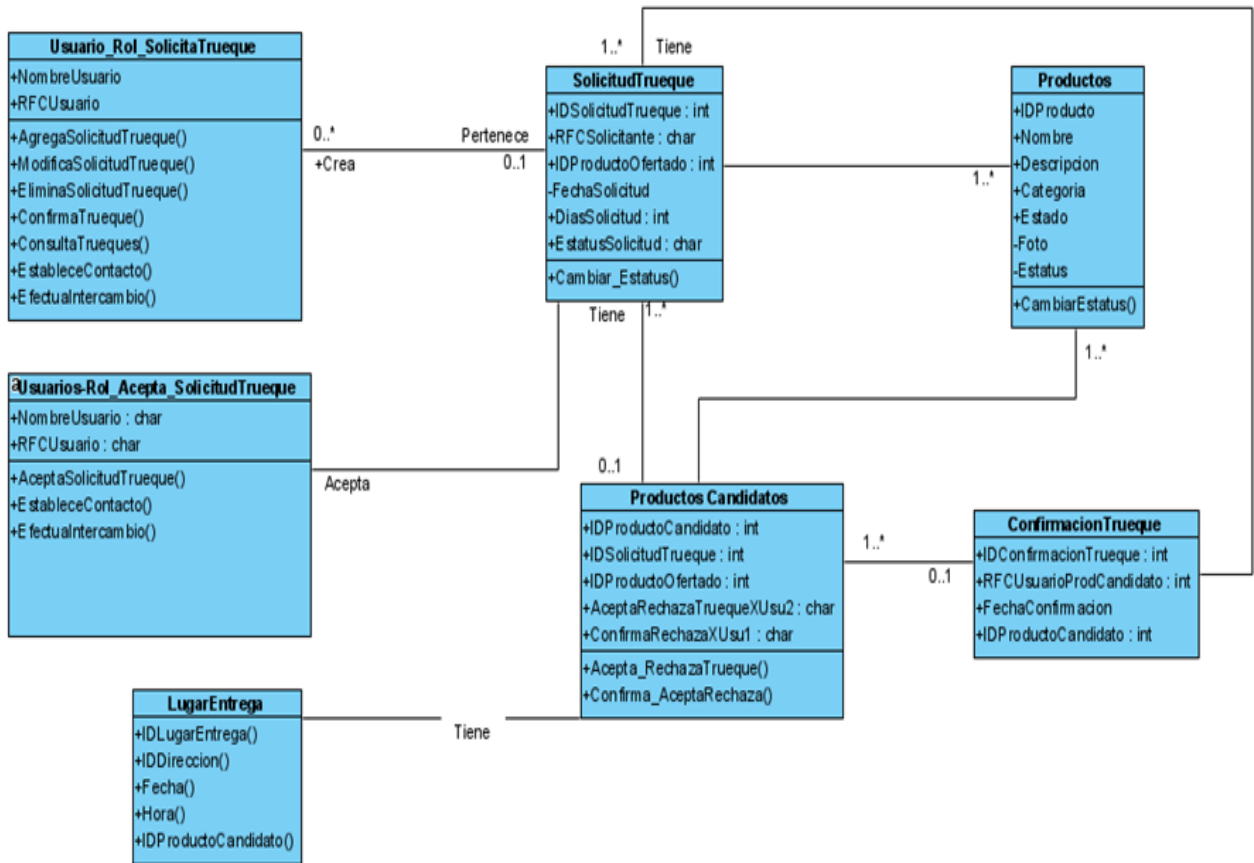


Figura 3.11 Diagrama de Clases Modulo Trueques

3.2.3 Diseño de Pantallas Principales del Prototipo de iTrueque

A continuación, se presentan el diseño pantallas de cada uno de los módulos de **iTrueque** con una breve explicación de cómo deberían funcionar, estos mismos puntos se retomarán para elaborar el manual de usuario.

3.2.3.1 Diseño Pantallas Modulo Usuarios

- **Prototipo de Pantalla Registro de Usuario**

La pantalla en donde se registrará el internauta para tener acceso a las funcionalidades del Sistema presentara un formulario como el de la pantalla de la figura 3.12, en donde

se validará que registre cada uno de los campos solicitados, que el correo electrónico sea un dato valido y se validará también que la contraseña cumpla con los requisitos de contraseña segura (Mínimo 8 caracteres, al menos una mayúscula, y al menos un carácter especial). Además, al insertar el registro del usuario en la tabla de usuarios se ejecutará un Trigger desde MySQL que cifrará la contraseña con SHA1.

The screenshot shows the user registration interface for iTrueque. The header includes the logo and navigation buttons. The registration form includes fields for RFC, Name, Surnames, Address, Location, and Contact Information. A password field with a security requirement note is also present.

Figura 3.12 Pantalla Registro Usuario-iTrueque

- **Prototipo de Pantalla de Acceso de Usuario al Sistema iTrueque**

La pantalla de inicio de sesión a **iTrueque**, será como la de la **Fig. 3. 13**, en donde se solicitará el Usuario y Contraseña y los datos proporcionados se validara en la tabla de usuario y en caso de haber coincidencia se iniciaran las variables de inicio de sesión con los datos del usuario y se le dará acceso a la pantalla principal del tipo de usuario que corresponda (**Administrador, UsuarioRegistrado**).



Figura 3.13 Inicio de Sesión

- **Diseño de Pantallas de Salida/Cerrar Sesión**

Para cerrar sesión el usuario simplemente desde la página principal de sus sesiones elegirá el botón salir y el sistema eliminará la información de las variables de sesión y cargará la página de inicio de **iTrueque**, en donde podrá navegar cualquier internauta con acceso solo a buscar productos, registrarse e iniciar sesión. La Figuras 3.14, muestra como cerrar sesión.

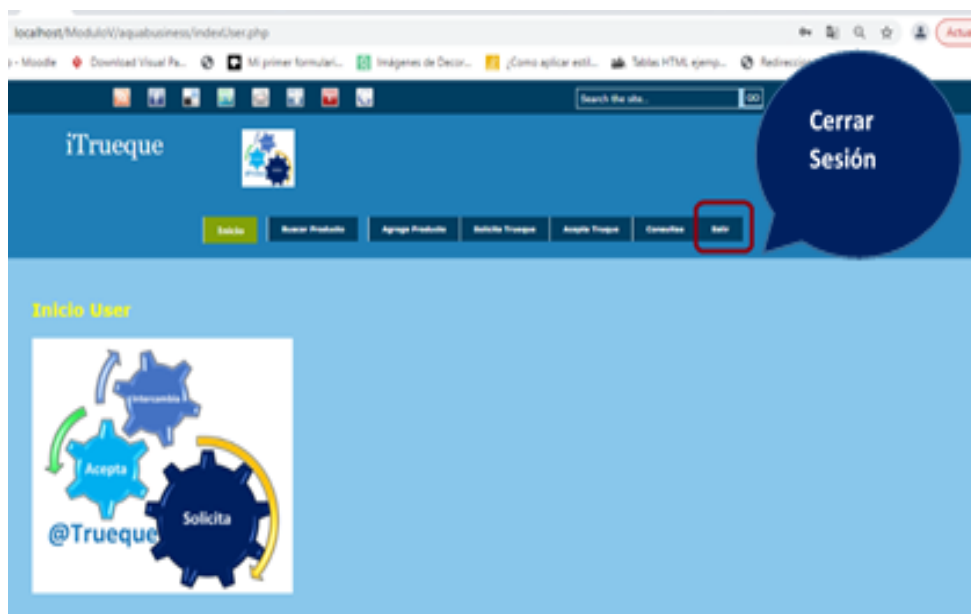


Figura 3.14. Cerrar sesión

3.2.3.2 Diseño Pantallas Módulo Productos

- **Diseño de Pantalla para función Agregar Producto**

La función **Agrega Producto** se encuentra en el módulo de Productos y permitirá al usuario poder agregar los productos que ofertará para intercambio posteriormente por medio de una Solicitud de Trueque. Se accederá a esta función desde la pantalla inicial del usuario en el Menú Productos se muestra en la Figura 3.15.



Fig. 3.15 Acceder a Agregar Producto en iTrueque

Una vez ingresado a la función se deberán llenar el formulario de **alta de productos** con los datos requeridos para agregar el producto a la tabla de productos de la base de datos de Trueques, en donde además del **nombre, descripción y estado del producto** se elegirá una **categoría del producto, la fecha de caducidad** y se **subirá una foto del producto**. Una vez completa la captura correcta, se clic al **botón Registrar Producto** y si toda la información es correcta el sistema grabara la información del producto, registrando además **la fecha de alta, el estatus inicial del producto** como **“Libre”**, esto es; listo para trueque, este campo no es editable para el usuario, el sistema cambiara los estados dependiendo del estado en el que se encuentre el producto (**“Libre”, “EnProceso”, “Baja”, “Intercambiado”**)

La Foto del Producto se subirá mediante código PHP a una carpeta de imágenes del Servidor de Datos y en el campo foto de la tabla productos se almacenará la ruta de esta foto.

Search the site... GO

iTrueque

Inicio Producto Solicita Trueque Acepta Trueque Consultas Salir

Alta Producto para Trueque

Nombre Producto: Descripción Producto:

Categoría: Estado del Producto:

Fecha Caducidad:

Foto Producto: Estado Producto:

La Figura 3.16 muestra el diseño de la pantalla alta de producto.

- **Diseño de Pantalla para función Elimina Producto**

La **figura 3.17** muestra el diseño pantalla para **eliminar o dar de baja un producto**. Esta función solo mostrara los productos del usuario activo y que además tienen estatus “Libre”, esto es; que no se han intercambiado y que no se encuentran en una solicitud de trueque, para este último caso deberá bajarse el producto de la solicitud de trueque antes de eliminar para que el sistema permita hacer la operación. Al eliminarse el producto el estado de este cambiará a “Eliminado” y no se podrá utilizar en alguna otra transacción del sistema.



Fig. 3.17 Diseño de Pantalla Elimina Producto

3.2.3.3 Diseño Pantallas Módulo Trueques

La pantalla de la Figura 3.18 muestra las diferentes funciones del módulo de trueques que son:

Solicitud de Trueques: Iniciar esta operación el usuario que creará la solicitud para intercambio de su producto por algún producto de su interés de otro usuario.

Modificar Solicitud de Trueques: Cambios a la solicitud, siempre que su producto se encuentre “EnProceso”

Eliminar Solicitud de Trueques: Eliminar una solicitud, siempre que su artículo se encuentre “EnProceso”

Acepta/Rechaza Solicitud de Trueque: Esta función la ejecutará cualquiera de los dos usuarios involucrados en la solicitud de trueque.

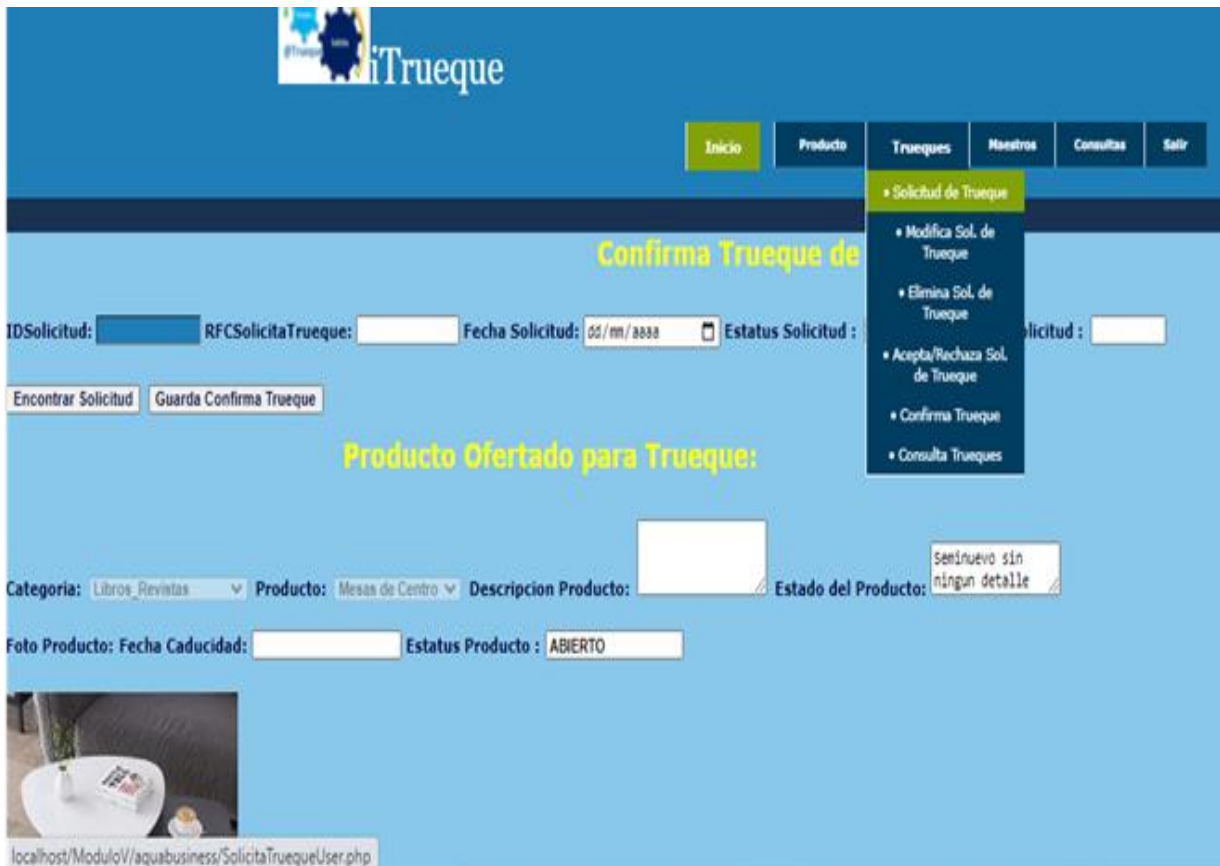


Figura 3.18 Menú Trueques

▪ **Diseño de Pantalla para función Solicita Trueque de Producto**

La pantalla de Solicitud de trueque será ejecutada por el usuario que desea ofertar su producto a cambio otro producto de una lista de productos pendientes de trueque de otros usuarios, seleccionados por el mismo, llamados “**productos candidatos**”.

La figura 3.19 es un diseño aproximado de la Solicitud de Trueque en donde el usuario seleccionara de su lista de productos pendientes de trueque el Producto a ofertar, **iTrueque** desplegara las características del producto y la sección de “**Productos Candidatos para Trueque**”, para que el usuario de acuerdo a su preferencia seleccione los productos de su interés de otros usuarios para trueque. Al guardar la solicitud, el sistema creara cambiara el estatus de ambos productos a “**EnProceso**”.



Figura 3.19 Pantalla Solicitud de Trueque

Diseño de Pantalla “Aceptar/Rechaza” Solicitud de Trueque

La Figura 3.20, ejemplifica un diseño aproximado de la pantalla que utilizara el usuario dueño del producto de interés seleccionado por el usuario que creo la solicitud de trueque en donde confirmara o rechazara la Solicitud de Trueque.

El usuario podrá activar la casilla “**Aceptar Trueque**” para confirmar el intercambio y la dejará desactivada para rechazar el intercambio. En caso de confirmar el intercambio el estatus de ambos productos cambiara a “**Cerrado**” y ambos usuarios establecerán contacto para proceder al intercambio físico y en caso de rechazar la solicitud el estado del producto cambiara de “**EnProceso**” a “**Libre**”, quedando listos para posibles futuros trueques.



Figura 3.20 Pantalla Solicitud de Trueque (Aceptar Trueque)

3.2.4 Diseño de Base de Datos iTrueque

Partiendo del **Modelo Entidad-Relación** se diseña la estructura de los datos utilizando el **Modelo Relacional** para el Sistema de Intercambio de productos **iTrueque**, se genera el **Modelo Entidad-Relación-Extendido** (Figura 3.21), en donde se representan entidades, relaciones, atributos, llaves primarias, llaves foráneas, etc. que implementadas en un sistema administrador de bases de datos como MySQL, darán solución a las estructuras de información para **iTrueque**.

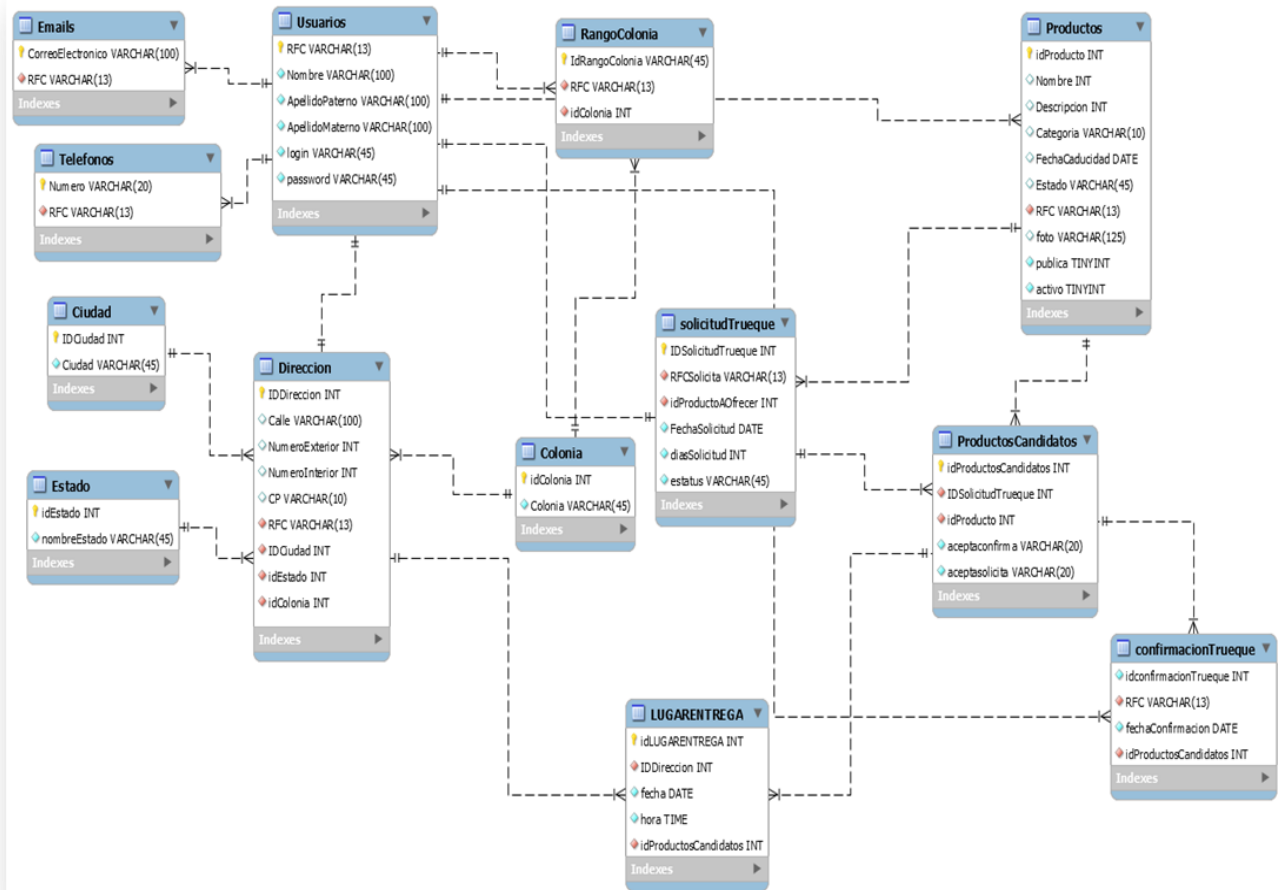


Figura 3.21 Diseño del Modelo Entidad Relación para iTrueque

Capítulo 4. Implementación y Pruebas del Sistema

Se describe el proceso de implementación del sistema en base al diseño previo. Se inicia con la arquitectura del sistema hasta la programación en PHP con MySQL, incluyendo pruebas y funcionamiento.

4.1 Definición de la arquitectura

La arquitectura seleccionada para el desarrollo de **iTrueque** es una arquitectura de modelo de 3 capas, como se mencionó en el marco teórico, que es el tipo de arquitectura ideal para una aplicación web que gestionara información a almacenar en un servidor de base de datos, en donde tendremos:

- **Capa de presentación**, representada por **el cliente** (el navegador, explorador o visualizador), que puede ser cualquier navegador web como **Google Chrome, Firefox, Edge, Opera, etc.**
- **Capa de Negocio**, representada por el **Servidor Web**, en este caso, será el servidor **Apache**.
- **Capa de datos**, representada por el **Servidor de Datos. MySQL**.

4.1.1 Integración de herramientas para la plataforma de desarrollo iTrueque

Como herramientas adicionales para el desarrollo de **iTrueque** del lado del cliente se utilizará el lenguaje de marcado **HTML**, y para el diseño de la presentación el lenguaje **CSS**.

Para el desarrollo y funcionamiento de **iTrueque** se integrarán los lenguajes **HTML, PHP, CSS** y el sistema de base de datos **MySQL**.

4.1.2 Plataforma de Desarrollo (WampServer)

Para la implementación de una arquitectura cliente–servidor de 3 capas cómo será la implementada en este caso, se utilizará la plataforma **WampServer** que es un entorno de desarrollo web para Windows con el que se pueden crear aplicaciones web con **Apache, PHP** y bases de datos **MySQL**.

La Figura 4.1 muestra los componentes incluidos en **WampServer** de los cuales se utilizarán para la implementación de **iTrueque**, las versiones:

- ✚ **WAMP SERVER 3.2.0 – 64 BITS**
- ✚ **phpAdmin 4.9.2**
- ✚ **Apache 2.4.41**

- + PHP 7.3.12
- + MariaDB 10.4.10



Figura 4.1 Herramientas de WampServer <http://diymakers.es/wp-content/uploads/2014/08/LAMP.png>

4.2 Desarrollo de la Base de Datos

De acuerdo a la información obtenida durante el análisis y diseño, se plasma en MySQL el diseño de la base de datos del prototipo funcional para **iTrueque**.

Se crea en MySQL la base de datos Trueque, utilizando la sentencia SQL:

```
create database trueque;
```

Se crean las entidades **Usuarios, Productos, cambios, Lugar de Entrega**.

4.2.1 Entidad Usuarios

Se crea la estructura para la **tabla usuarios** dentro de la base de datos **Trueque**, en donde la llave primaria o identificador único de cada instancia es el **campo id_usuario**, el cual es tipo `auto_increment`, con lo que se garantiza que su valor siempre sea único, la siguiente sentencia SQL crea la tabla usuarios:

```
CREATE TABLE usuarios (
  id_usuario INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(128) NOT NULL,
  celular varchar (12),
  mail varchar (128) NOT NULL,
  password varchar (128) NOT NULL,
  direccion varchar (250),
  ubicacion varchar (64),
  tipo int(1)
```

);

La Figura 4.2 muestra la Tabla usuarios desde phpMyAdmin



#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	id_usuario			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/>	2	nombre	latin1_swedish_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	3	celular	latin1_swedish_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	4	mail	latin1_swedish_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	5	password	latin1_swedish_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	6	direccion	latin1_swedish_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7	ubicacion	latin1_swedish_ci		Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	8	tipo			Sí	NULL			Cambiar Eliminar Más

Figura 4.2 Estructura de Entidad Usuarios en MySQL

4.2.2 Entidad Productos

Se crea la estructura para la **tabla productos** dentro de la base de datos **Trueque**, en donde la llave primaria o identificador único de cada instancia es el campo **id_producto**, el cual es tipo `auto_increment`, con lo que se garantiza que su valor siempre sea único, el campo `estatus` tomara solo 3 posibles valores, los cuales serán cambiados por el sistema y no por el usuario y se dispararan cuando sucedan los diferentes eventos del usuario (valor 1 cuando el usuario crea el producto e indicara que esta **“Libre”**, valor 2 cuando el usuario crea una solicitud de trueque e indicara que esta **“EnProceso”**, valor 3 cuando el usuario al que le están solicitando trueque Acepta el trueque e indicara que el producto esta **“Intercambiado”**, valor 4 cuando el usuario da de baja el producto e indicara que el producto esta **“Eliminado”**). El campo **xcambio** contendrá el **id_producto** del producto de otro usuario, al que el usuario que genero la solicitud de trueque selecciono para trueque. La siguiente sentencia SQL crea la tabla productos:

```
CREATE TABLE productos (  
  id_producto INT NOT NULL AUTO INCREMENT PRIMARY KEY,  
  id_usuario INT,  
  producto varchar(64),  
  descripcion VARCHAR(100) NOT NULL,  
  xcambio varchar(120),  
  clasificacion varchar(30),
```

foto varchar(64),
estatus int(1),
FOREIGN KEY(id_usuario) REFERENCES usuarios(id_usuario)
);

La Figura 4.3 muestra la estructura de la Tabla **productos** desde phpMyAdmin

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1	id_producto			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/>	2	id_usuario			Si	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	3	producto	latin1_swedish_ci		Si	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	4	descripcion	latin1_swedish_ci		No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	5	xcambio	latin1_swedish_ci		Si	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	6	clasificacion	latin1_swedish_ci		Si	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7	foto	latin1_swedish_ci		Si	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	8	estatus			Si	NULL			Cambiar Eliminar Más

Figura 4.3 Estructura de Entidad Productos en MySQL

4.2.3 Entidad Cambios

Se crea la estructura para la **tabla cambios** dentro de la base de datos **Trueque**, en esta tabla el identificador único de cada fila o llave primaria será el campo **id_cambio**, el cual es de tipo **auto_increment**, esto es, **MySQL** será el responsable de crear un consecutivo único para cada instancia de esta tabla. Cada registro de esta tabla se creará cuando un usuario agrega una nueva solicitud de trueque de alguno de sus productos por un producto de su interés de otro usuario previamente ofertado para Trueque.

Al crearse una nueva solicitud de trueque el **campo Validar** de la **tabla cambios** tomara el **valor de 0** que indica que la solicitud es recién creada, si el usuario al que se le está haciendo la oferta de trueque la acepta el **campo Validar será igual a 1**, pero si este usuario rechaza la solicitud de trueque el **campo validar tendrá el valor de 2**.

La siguiente sentencia SQL creara la tabla cambios en MySQL:

```
CREATE TABLE cambios (
  id_cambio INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  id_usuario INT NOT NULL,
  id_remitente INT NOT NULL,
```

id_productoSolicitado INT NOT NULL,
 id_productoAcambio INT NOT NULL,
 validar int(1),
 fechaAplica date,
 FOREIGN KEY(id_usuario) REFERENCES usuarios(id_usuario)

La Figura 4.5 muestra la estructura de la tabla **cambios** desde phpMyAdmin.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/>	1 id_cambio	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/>	2 id_usuario	int(11)			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	3 id_remitente	int(11)			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	4 id_productoSolicitado	int(11)			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	5 id_productoAcambio	int(11)			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/>	6 validar	int(1)			Sí	NULL			Cambiar Eliminar Más
<input type="checkbox"/>	7 fechaAplica	date			Sí	NULL			Cambiar Eliminar Más

Figura 4.4 Estructura de la Entidad Cambios en MySQL

4.2.4 Llaves Primarias, Llaves Foráneas y Relaciones entre Entidades

Como vimos en la creación de tablas, para la **tabla Usuarios** su identificador único o **Llave Primaria** es el **campo id_usuario** y se **relaciona** su llave primaria con el **campo id_usuario en la tabla productos**, el cual en esta tabla es la **llave foránea**; por otro lado, el mismo **campo id_usuario de la tabla usuarios** se **relaciona** con la **tabla cambios** con el campo con el mismo nombre (**id_usuario**) que es una **llave foránea** en esta tabla.

Para la **tabla productos** su **llave primaria** es **id_producto** y para la **tabla cambios** su **llave primaria** es **id_cambio** y para **ambas** su **llave foránea** es **id_usuario**.

De esta forma se establecen las relaciones en las tres tablas, las cuales permitirán que los usuarios de la tabla usuarios puedan tener uno o más productos de la tabla de productos y que puedan ofertar sus productos por productos de otros usuarios y estas solicitudes se registren en la tabla cambios y por medio de un estatus se sabrá si la solicitud está en proceso, aceptada o rechazada. La Figura 4.5 muestra la estructura de las tablas de la base de datos Trueque, con sus respectivas llaves primarias, llaves foráneas y las relaciones entre estas explicadas previamente.

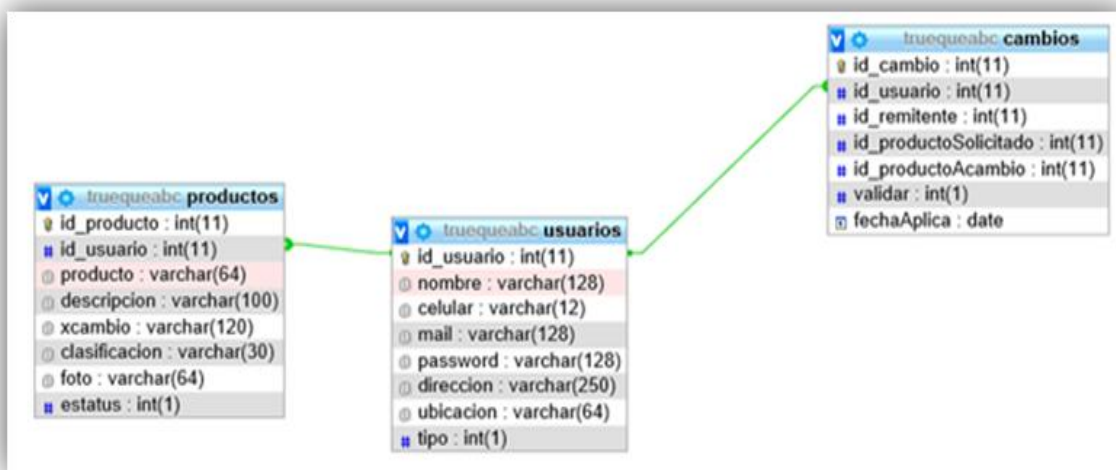


Figura 4.5 Estructura de las Tablas Principales y sus relaciones Base de Datos Trueque

4.3 Pruebas e Implementación de las Principales Funciones del prototipo de iTrueque.

Para el desarrollo del prototipo del Sistema de Intercambio de Productos **iTrueque**, se partirá de la descripción de funciones y de cada uno de los **modelos UML** creados durante el proceso de diseño, esto es; elementos tales como actores, diagrama de casos de usos, diagrama de clases, diagramas de secuencia, etc.

4.3.1 Funciones Internauta

Las primeras funciones para implementarse son las del internauta, que es cualquier navegador web que llega por primera vez a visitar el sitio **iTrueque**. El internauta podrá:

- **Tener acceso a una Página Principal (Menú Internauta)**
- **Consultar Productos** Ofertados para Trueque por usuarios registrados en el sistema.
- **Registrarse como Usuario del Sistema**

4.3.1.1 Página Principal (Menú Internauta)

Para la creación del menú del usuario internauta se utiliza del lado del cliente lenguaje **HTML** y lenguaje **PHP**, para el diseño de la presentación el lenguaje **CSS**, tal como se definió la arquitectura de desarrollo en el punto 4.1. El código escrito quedara como se muestra en la figura 4.6

```

<body>
<div id="header">
<ul id="menu">
<li><a href="index.php" accesskey="1">Inicio</a></li>
<li><a href="usuarios.php" accesskey="2">usuarios registrados</a></li>
<li><a href="registro.php" accesskey="3">registro de usuarios</a></li>
<li><a href="productosInternauta.php" accesskey="4">productos</a></li>
<li><a href="contacto.php" accesskey="5">Contacto</a></li>
</ul>
</div>
<div id="content">
<div id="colOne">
<div id="logo">
<h1><span class="Estilo2"> iTrueque</span><span class="Estilo1">que</span></h1>
<h2>&nbsp;</h2>
<h2>Facultad de ciencias de la Computación </h2>
<br></br>
</div>
</div>
<div id="colTwo">
<h2>Bienvenidos al sistema de intercambio de productos </h2>

<p class="indent">El intercambio de productos o trueque es una práctica milenaria en diferentes regiones del país, y ante el auge de las nuevas tecnologías y el acceso cada vez más sencillo a internet, se plantea este sistema web para la publicación de productos de un grupo de personas para ser intercambiados por otros productos.</p>
<p class="indent">Este sistema es dedicado a la publicación de artículos varios para ser intercambiados por otros usuarios principalmente en la ciudad de Puebla. Se tienen

```

Figura 4.6 Código Fuente para Pantalla Principal de Internauta en iTrueque

Se realizan **Pruebas de Unidad** (para mostrar que el código funciona) y **Pruebas de Cliente** (para demostrar que cumple con las necesidades del cliente) resultando satisfactorias. La Página Principal del internauta se muestra en la figura 4.7 en donde podrá Consultar Productos Ofertados para trueque por usuarios registrados y podrá también registrarse en el sistema. Así queda implementada la función Página Principal Internauta.



Figura 4.7 Página Principal iTrueque para Internauta

4.3.1.2 Consultar Productos

La Función Productos permitirá al internauta poder consultar los productos ofertados para trueque por usuarios registrados en el sistema. Sin embargo, para poder contactar a algún usuario, para poder registrar un producto, o para poder crear una solicitud de trueque el internauta deberá registrarse en el sistema.

El código fuente escrito con las herramientas de desarrollo **PHP**, **HTML** y **CSS** para la consulta de productos por parte del internauta se muestra en la figura 4.8.

```
43 <?php
44 include("conexion.php");
45 $link=Conectarse();
46 $result=mysqli_query($link,"select * from productos where estatus=1");
47 ?>
48
49 <table border="1">
50
51 <tr><th> Imagen </th><th> Id Producto </th><th> Producto </th><th> Descripción </th></tr>
52
53
54 <?php
55
56 while($row = mysqli_fetch_array($result))
57 {
58     $id=$row["id_producto"];
59     $pro=$row["producto"];
60     $des=$row["descripcion"];
61     $fot=$row["foto"];
62     $cad="<img src='imagenesProductos/$fot' width='60' height='80'>";
63     echo"<tr><td> $cad </td><td><center> $id </center></td><td> $pro </td><td> $des </td></tr>";
64 }
65
66 mysqli_free_result($result);
67 mysqli_close($link);
68
69 </table>
```

Se realiza conexión al Servidor MySQL y se consultan todos los artículos publicados por usuarios registrados que no se encuentran en alguna solicitud de trueque.

Se muestra la información de los productos consultados en la página de consulta.

Se cierra conexión con la base de datos MySQL

Figura 4.8 Código Fuente de Consulta de Productos en Pagina Internauta, sistema iTrueque

Se realizan **Pruebas de Unidad** (para mostrar que el código funciona) y **Pruebas de Cliente** (para demostrar que cumple con las necesidades del cliente) y se hacen las correcciones necesarias hasta obtener resultandos satisfactorios.

La implementación de la **consulta de productos** que realiza el internauta se muestra en la figura 4.9 en donde se mostraran todos los Productos Ofertados para trueque por usuarios registrados, el código queda registrado en el archivo **productosinternauta.php**.

Imagen	Id Producto	Producto	Descripción
	1	Silla	Silla de madera
	2	Mesa	Mesa de plástico para 4 personas
	3	Bocinas	Bocinas para computadora Logitech
	4	Cafetera/Una taza/vaso	Cafetera de una sola taza y taza de viaje de 14 onzas. Portátil y ligero personal de goteo cafetera y vaso con función avanzada de apagado automático y filtro ecológico reutilizable (lavanda)
	5	Bolso de Mano Azul	BOLSO DE MANO PEQUEÑO DE PIEL CON ASA Y MEDUSA COLOR AZUL Exterior 100% Beccro, Interior 100% Corduro. Altura 20cm Ancho 25cm Profundidad 12cm
	7	Anillo	Anillo Oro Blanco 14 K con 10 puntos de diamante.

Figura 4.9 Consulta de Productos de otros usuarios ofertados para trueque

4.3.1.3 Registrarse como Usuario del Sistema

Finalmente se realiza el desarrollo de la última función del internauta de **iTrueque**, que es la de registrarse al sistema para poder tener un usuario y una contraseña para ingresar, ofertar productos para trueque, solicitar trueque de sus productos por productos de su interés de otros usuarios. La pantalla de registro se muestra en la figura 4.10 y el código fuente se muestra en las figuras 4.11 y 4.12.

Inicio usuarios registrados registro de usuarios productos Contacto	
iTrueque Facultad de ciencias de la Computación	
Registro de nuevos usuarios	
Nombre completo:	<input type="text"/>
Dirección:	<input type="text"/>
Teléfono:	<input type="text"/>
Mail:	<input type="text"/>
Ubicación que te es más cercana	Plaza Dorada <input type="button" value="v"/>
<input type="button" value="Registrar usuario"/>	
	

Figura 4.10 Pantalla Registro de Usuario



Figura 4.11 - Código Fuente para Formulario de Registro a iTrueque

El código del archivo **registro2.php**, es el encargado de tomar la información capturada por el internauta en el formulario de registro (Figura 4.11), toma el correo electrónico como identificador del usuario de registro, valida que no se encuentre ya registrado, en caso de que ya este registrado manda mensaje de error, en caso contrario registrara al usuario.

Contraseña Segura

Como parte del registro del usuario, mediante un algoritmo se genera una contraseña formada por caracteres elegidos aleatoriamente, y para reforzar la seguridad de la contraseña, al crear el registro del usuario en la base de datos **Trueque**, se desarrolló un Stored Procedure en MySQL que inserta el registro del usuario en la base de datos cifrando su contraseña con el algoritmo de cifrado SHA1, este Stored Procedure será llamado desde php. El código SQL del Stored Procedure queda de la siguiente forma:

```
CREATE PROCEDURE InserU (IN nom varchar(128), IN cel varchar(12), IN mail
varchar(128), IN pass varchar(128), IN dir varchar(250), IN ubi varchar(64), tipo int(1))
insert into usuarios (nombre, celular, mail, password, direccion, ubicacion, tipo)
values(nom, cel, mail, sha1(pass), dir, ubi, tipo)
```

```

include("conexion.php");
$link=Conectarse();
$result = mysqli_query($link,"SELECT mail FROM usuarios WHERE mail='$mai'");
$row = mysqli_num_rows($result);

if ($row>0) {
    echo "<h3>La cuenta de correo ya esta en uso </h3><br>";
    echo " * Intenta un nuevo registro de usuario <br><br>";
}
else{
    //mysqli_query($link,"insert into usuarios(nombre,celular,mail,direccion,password,tipo,ubicacion)
    //          values ('$nom','$tel','$mai','$dir','$cad',1,'$lugar')");

    $query="call InsertU ('$nom','$tel','$mai','$cad','$dir','$lugar',1)";

    mysqli_query($link,$query);

    echo "<br>REGISTRO VÁLIDO <BR>";
    echo "<h4><B>Usuario:</B> $mai </h4>";
    echo "<h4><B>Contraseña :</B> $cad </h4><br>";

    echo " * Bienvenido ya puedes hacer uso del sistema <br>";
}

```

Figura 4.12 Inserta Registro de Usuario Nuevo

Una vez que el internauta registra sus datos, el sistema valida la información ingresada y genera contraseña segura y cifrada, guarda el registro en la base de datos y confirma registro. Las pantallas de las figuras 4.13 y 4.14, muestran el registro del usuario.

Inicio usuarios registrados registro de usuarios productos Contacto

iTrueque
Facultad de ciencias de la Computación

Registro de nuevos usuarios

Nombre completo:	<input type="text" value="Robin Rojas"/>
Dirección:	<input type="text" value="20 de Noviembre 1232 Centro, Puebla, Pue"/>
Teléfono:	<input type="text" value="2223430918"/>
Mail:	<input type="text" value="robinH@hotmail.com"/>
Ubicación que te es más cercana:	<input type="text" value="Plaza Dorada"/>

 El intercambio de productos debe ser gratuito, no se permite comprar o vender los productos.

Figura 4.13 Formulario Registro



Figura 4.14 Confirmación de Registro

Para que quede funcionando el desarrollo del registro de usuario se realizan Pruebas de Unidad (para mostrar que el código funciona) y Pruebas **de Cliente** (para demostrar que cumple con las necesidades del cliente) y se hacen las correcciones necesarias hasta obtener resultados satisfactorios.

4.3.2 Funciones del Usuario Registrado / Usuario Administrador

El usuario registrado y el usuario administrador tendrán acceso a las mismas funciones, la diferencia es que el usuario registrado solo podrá trabajar con sus productos y el administrador con los de todos los usuarios.

4.3.2.1 Acceso a iTrueque con cuenta de usuario registrado

El usuario registrado en **iTrueque** tendrá usuario y contraseña para iniciar sesión en el Sistema, de tal forma que la primera función a desarrollar, probar e implementar es el acceso al sistema.

El código fuente para crear el formulario de acceso a **iTrueque** se muestra en la figura 4.15 y la figura 4.16 muestra el formulario de acceso

```

usuarios.php validarUsuario.php
Código Dividir Diseño Título: sistema de trueque
32 </div>
33 </div>
34 <div id="colTwo">
35 <h2>Usuarios registrados </h2>
36 <p class="indent">
37 <form action="validarUsuario.php" method="post">
38 Correo:
39 <br />
40 <input type="text" name="correo" required />
41 <br /><br />
42 Password:
43 <br />
44 <input type="password" name="passwd" required />
45 <br /><br />
46 <input type="submit" name="enviar" value=" Entrar " />
47 </form>
48 </p>
49 <p>&nbsp;</p>
50 <p>* Si aun no tienes cuenta puedes crear una en &quot;registro de usuarios&quot; </p>
51 <p align="left" class="indent">&nbsp;</p>
52 <hr>
53 <h3></h3>
54 <p>El intercambio de productos debe ser gratuito, no se permite comprar o vender los productos. </p>
55 <p>&nbsp;</p>
56 <p>&nbsp;</p>

```

Figura 4.15 Código Fuente para Formulario de acceso a iTrueque



Figura 4.16 Formulario Acceso a iTrueque

Una vez que el usuario ingresa con sus credenciales de acceso, el sistema valida que el usuario (correo electrónico) exista buscándolo en la base de datos de **iTrueque**, si no lo encuentra manda mensaje de error (función **errorlogin.php**) si existe valida que la contraseña sea correcta, si no lo es, manda mensaje de error (función **errorpassword.php**) y limpia formulario de acceso, si la contraseña es correcta llena las variables de sesión con los datos del usuario y muestra pantalla de bienvenida con el menú de funciones de **iTrueque**.

La pantalla de la figura 4.17 muestra el código fuente desarrollado para validar las credenciales del usuario y dar acceso a las funciones de **iTrueque** y la figura 4.18 muestra la página principal de **iTrueque** con la bienvenida al usuario.

```

1 <?PHP session_start();
2 $correo=$_REQUEST['correo'];
3 $pas=$_REQUEST['passwd'];
4
5 $link=mysqli_connect("localhost","root","");
6 mysqli_select_db($link,"trueque");
7 $result = mysqli_query($link,"SELECT password,mail,tipo,nombre, id_usuario FROM usuarios WHERE mail='".$correo."'");
8
9 // $result = mysqli_query($link,"select * from usuarios where mail='sofyanroto@gmail.com' and password=sha1('n6qix0o2')");
10
11 if($row = mysqli_fetch_array($result))
12 {
13     $result2 = mysqli_query($link,"SELECT password,mail,tipo,nombre, id_usuario FROM usuarios WHERE mail='".$correo.'" and password=sha1('$pas')");
14     // $row2 = mysqli_num_rows($result2);
15     // if($row["password"] == $pas)
16     if($row2 = mysqli_fetch_array($result2))
17     {
18         $_SESSION["usuario"] = $row['mail'];
19         $_SESSION["id_usuario"] = $row['id_usuario'];
20         $_SESSION["tipo"] = $row['tipo'];
21         echo "Has sido logueado correctamente: ".$_SESSION['usuario'];
22         if($row['tipo']==1) header("Location:indexUsuarios.php");
23         // if($row['tipo']==0) header("Location:indexADM.php");
24     }
25 }
26
27 else header("Location:errorPassword.php");
28
29 }
30
31 else
32
33 header("Location:errorLogin.php");
34
35 ?>

```

Figura 4.17 Código fuente para validar credenciales de usuario y darle acceso a itrueque



Figura 4.18 Página Principal de iTrueque para un usuario registrado

El usuario registrado en el sistema podrá agregar nuevos productos para trueque, consultar sus productos ofertados para trueque, crear solicitudes de trueque, aceptar solicitudes de trueque de otros usuarios. El menú de la página principal se muestra en la figura 4.19.



Figura 4.19 - Menú Principal iTrueque para Usuario Registrado

4.3.2.2 Agregar productos nuevos para trueque (Función Nuevos)

Se desarrolla el código fuente en php y MySQL para la función que agrega nuevo producto, el cual posteriormente el usuario podrá ofertar en una solicitud de trueque por algún producto de otro usuario. El código fuente de esta función se muestra en las Figuras 4.20 y 4.21.

```

<form action="altaProductos2.php" method="post" ENCTYPE="multipart/form-data">
Nombre: <input type="text" name="producto"> <br><br>
Descripción:
<TEXTAREA COLS="80" ROWS="4" NAME="descrip">
... describir el producto ...
</TEXTAREA>
<br><br>
Selecciona como lo clasificas:
<SELECT NAME="clasifica">
<OPTION VALUE="muebles" SELECTED> Muebles
<OPTION VALUE="ropa"> Ropa
<OPTION VALUE="zapatos"> Zapatos
<OPTION VALUE="jardineria"> Jardineria
<OPTION VALUE="electronico"> Electrónicos
<OPTION VALUE="papeleria"> Papeleria
<OPTION VALUE="artesanias"> Artesanias
<OPTION VALUE="juguetes"> Jugetes
<OPTION VALUE="otros"> Otros (varios)
</SELECT>

<br><br>
Porque producto te gustaria intercambiar: <input type="text" name="acambio" size="80">
<br><br>
<br>
Cargar imagen del producto:
<INPUT TYPE="file" NAME="foto">
<br><br>

```

Figura 4.20 Código Fuente para generar formulario de Registro de Producto

```

<?PHP
$pro=$_REQUEST['producto'];
$des=$_REQUEST['descrip'];
$aca=$_REQUEST['acambio'];
$cla=$_REQUEST['clasifica'];
$fot=$_FILES['foto']['name'];
$susu=$_SESSION['id_usuario'];
echo "$pro <br>";
echo "$des <br>";
echo "$aca <br>";
echo "$cla <br>";
echo "$fot <br>";

echo "Usuario: $usu <br>";

$ruta=../../trueque/imagenesProductos/".$fot;
if(is_uploaded_file($_FILES['foto']['tmp_name']))
copy($_FILES['foto']['tmp_name'],$ruta);

$link=mysqli_connect("localhost","root","");
mysqli_select_db($link,"trueque");

mysqli_query($link,"insert into productos(id_usuario, producto, descripcion, xcambio, clasificacion, foto, estatus)
values ($usu,$pro,$des,$aca,$cla,$fot,1)");
?>

```

Figura 4.21 - Código Fuente para validar y Registrar un Producto Nuevo

La Figura 4.22 muestra el formulario de alta de producto, resultado del código de la figura 4.20 ya con un artículo (producto) capturado por un usuario. La Figura 4.23 muestra la pantalla de confirmación de que el artículo ya se grabó en la base de datos de productos,

operación generada con la ejecución del código fuente (Figura 4.21) ejecutada mediante el botón “**Enviar Datos**”.

The screenshot shows a web form titled "Nuevos productos:" with the following fields and elements:

- Navigation bar: Inicio, Nuevos, Productos, Solicitudes, Intercambiar, Salir
- Header: iTrueque, Facultad de Ciencias de la Computación
- Form fields:
 - Nombre:
 - Descripción:
 - Selecciona como lo clasifica: - Porque producto te gustaría intercambiar:
 - Cargar imagen del producto: ZapatillasRosas.jpg
- Submit button:

Figura 4.22 – Pantalla de ejemplo de Alta de un Nuevo Producto en iTrueque

The screenshot shows a confirmation page with the following content:

- Navigation bar: Inicio, Nuevos, Productos, Solicitudes, Intercambiar, Salir
- Header: iTrueque, Facultad de Ciencias de la Computación
- Section: **Producto almacenado con éxito**
- Text: * Si deseas almacenar mas productos ve a la opción productos
- Product details:
 - ZapatillasRosas
 - Zapatillas De Plataforma Color Rosa Pastel Para Dama Talla 24
 - Mesa
 - Zapatos
 - ZapatillasRosas.jpg
 - Usuario: 3

Figura 4.23 Sistema Confirma Alta de Producto

Cuando un nuevo producto se registra en la base de datos de productos el **estatus del producto es = 1**, que significa que el producto solo está dado de alta y no se encuentra en alguna solicitud de trueque.

Para que quede funcionando el desarrollo de la función Nuevos se realizan Pruebas de Unidad (para asegurarse que el código funciona) y Pruebas **de Cliente** (para demostrar que

cumple con las necesidades del cliente) y se hacen las correcciones necesarias hasta obtener resultados satisfactorios.

4.3.2.3 Consulta, Modificación y Baja de Productos (Función Productos)

Esta función muestra Todos los Productos del Usuario Activo que están Pendientes de Trueque. Desde aquí además se pueden **modificar los datos de los artículos**, además de poder **darlos de baja**; en ambos casos siempre y cuando no se encuentren en alguna solicitud de trueque o ya en un trueque realizado.

Consulta de Productos

La Figura 4.24 muestra el código fuente que consulta en MySQL los productos libres, esto es; los productos que no están en alguna solicitud de trueque y que no están ya intercambiados. Una vez obtenida el resultado de la consulta la información se despliega en la tabla de productos cargada en la página de Productos, como se muestra en la Figura 4.25.

The image shows a PHP code snippet for querying products in MySQL. The code includes a database connection, a query to fetch products for an active user, and a loop to display the results in an HTML table. Three callouts provide additional context:

- Top Callout:** "Consulta en MySQL los productos libres (no intercambiados ni en solicitud de trueque)" - This points to the SQL query: `$result=mysqli_query($link,"select * from productos where id_usuario=$idusuario and estatus=1");`
- Middle Callout:** "MySQL devuelve información solicitada y se coloca esta en variables" - This points to the `while($row = mysqli_fetch_array($result))` loop where data is assigned to variables like `$id=$row["id_producto"];`
- Bottom Callout:** "Cada registro obtenido de la consulta de vacía en la tabla para que el usuario pueda ver la consulta" - This points to the `mysqli_free_result($result);` and `mysqli_close($link);` lines at the end of the code.

Figura 4.24 - Código Fuente para Consultar en MySQL los productos listos para trueque del usuario activo y los muestra en una tabla de consulta.

	Id Producto	Producto	Descripción	Estatus	Eliminar	Modificar
	4	CafeteraUnaTazaViaje	Cafetera de una sola taza y taza de viaje de 14 onzas. Portátil y ligero personal de goteo cafetera y vaso con función avanzada de apagado automático y filtro ecológico reutilizable (lavanda)	Libre		
	5	Bolso de Mano Azul	BOLSO DE MANO PEQUEÑO DE PIEL CON ASA Y MEDUSA COLOR AZUL Exterior 100% Becerro, Interior 100% Cordero. Altura 20cm Ancho 25cm Profundidad 12cm	Libre		
	7	Anillo	Anillo Oro Blanco 14 K con 10 puntos de diamante.	Libre		
	8	ZapatillasRosas	Zapatillas De Plataforma Color Rosa Pastel Para Dama Talla 24	Libre		

Figura 4.25 Resultado de Consulta de Productos

Modificación de Productos

Dentro del código fuente que consulta los productos, mostrado anteriormente en la figura 4.24, en el recuadro azul; por cada producto mostrado se coloca el botón “**Modificar**”, en la columna Modificar de la tabla de productos este permitirá al usuario modificar las características de su producto. Cuando el usuario utiliza este botón para modificar el producto se ejecuta el código php del archivo **actualizaProductos.php**, mostrado en la Figura 4.26, el cual se encarga de mostrar un formulario con la información del producto, para que el usuario realice los cambios a la información del producto y al actualizar los cambios se ejecuta el código php del archivo **actualizaProductos2.php**, mostrado en la Figura 4.27, el cual se encarga de guardar los cambios realizados por el usuario en la base de datos de productos.

```

<?php
include("conexion.php");
$link=Conectarse();
$id=$_REQUEST['id_producto'];

echo '<FORM ACTION="actualizaProductos2.php" Method= "post">';

$result=mysqli_query($link,"select producto, descripcion, clasificacion, foto from productos where id_producto='$id'");
$row = mysqli_fetch_array($result);
$pro=$row["producto"];
$des=$row["descripcion"];
$cla=$row["clasificacion"];
$fot=$row["foto"];

echo "<img src='imagenesProductos/$fot' width='120' height='160'>";
echo "<br>";
echo "Producto : <INPUT TYPE='text' NAME='producto' value='$pro' SIZE='50'>";
echo "<br>";
echo "<br>";
echo "Descripción: <br>";
echo "<input type='text' name='descrip' value='$des' size='300'>";

echo "<br><br>";
echo "Clasificación: $cla <br>";
echo "<input type='hidden' name='id' value='$id'>";
echo "<br>";
echo "<br>";

```

Figura 4.26 – Código Fuente en php que crea formulario con datos del Producto para modificación por parte del usuario dueño del producto

```

<?php
include("conexion.php");
$link=Conectarse();

$pro=$_REQUEST["producto"];
$des=$_REQUEST["descrip"];
$id=$_REQUEST['id'];

echo "Producto: $pro <br>";
echo "Descripcion: $des <br>";
echo "ID = $id <br>";
mysqli_query($link,"Update productos Set producto='$pro', descripcion='$des' Where id_producto='$id'");
header("Location: productos.php");
?>

```

Figura 4.27 – Código Fuente para actualizar datos cambiados del producto en la base de datos MySQL

Eliminar Productos

Dentro del código fuente que consulta los productos, mostrado anteriormente en la figura 4.24, en el recuadro rojo; por cada producto mostrado se coloca el botón “Eliminar”, en la columna Eliminar de la tabla de productos, este permitirá al usuario eliminar el producto, esto se puede hacer porque la consulta solo muestra productos del usuario que no se han intercambiado y que tampoco se encuentran en una solicitud de trueque.

Cuando el usuario utiliza este botón para eliminar el producto, se creó una función que muestra una pantalla de alerta y/o confirmación (mostrada en la Figura 4.28), para asegurarse de que efectivamente el usuario quiere eliminar el producto, el usuario puede con el botón “**Cancelar**” abortar la operación, pero si confirma con el botón “**Aceptar**”, en seguida se ejecuta el código php del archivo **borrar2new.php**, mostrado en la Figura 4.29, el cual se encarga de forma permanente el producto.

localhost dice
 Está seguro de eliminar este registro?

Aceptar Cancelar

Productos almacenados

	Id Producto	Producto	Descripción	Estatus	Eliminar	Modificar
	4	CafeteraUnaTazaViaje	Cafetera de una sola taza y taza de viaje de 14 onzas. Portátil y ligero personal de goteo cafetera y vaso con función avanzada de apagado automático y filtro ecológico reutilizable (lavanda)	Libre		
	5	Bolso de Mano Azul	BOLSO DE MANO PEQUEÑO DE PIEL CON ASA Y MEDUSA COLOR AZUL Exterior 100% Becerro, Interior 100% Cordero. Altura 20cm Ancho 25cm Profundidad 12cm	Libre		
	7	Anillo	Anillo Oro Blanco 14 K con 10 puntos de diamante.	Libre		
	8	ZapatillasRosas	Zapatillas De Plataforma Color Rosa Pastel Para Dama Talla 24	Libre		

Figura 4.28 Pantalla que solicita confirmación al usuario para eliminar definitivamente un producto no intercambiado y ni en solicitud de trueque

```

1 <html>
2 <head>
3
4
5 </head>
6 <body>
7 <?php
8 session_start();
9 if($_SESSION["tipo"]!=1) header("Location:index.php");
10 include("conexion.php");
11 $link=Conectarse();
12 $id=$_REQUEST['id_producto'];
13 // echo "el valor es : $id";
14 mysqli_query($link,"delete from productos where id_producto = '$id'");
15 header("Location: productos.php");
16 ?>
17 </body>

```

Figura 4.29 Código PHP que elimina producto no intercambiado, ni en solicitud de trueque

Para que quede funcionando el desarrollo de la **función Productos**, para cada una de sus subfunciones: Consulta, Modificación y Eliminación de Productos se realizan Pruebas de Unidad (para asegurarse que el código funciona) y Pruebas **de Cliente** (para demostrar que cumple con las necesidades del cliente) y se hacen las correcciones necesarias hasta obtener resultados satisfactorios.

4.3.2.4 Crear y Enviar Solicitudes de Trueque (Función Intercambiar)

La Función **Intercambiar** está diseñada para que el usuario activo seleccione algún producto de otro usuario para intercambiar por alguno de sus productos mediante una solicitud de trueque.

Crear Solicitud de Trueque

Al entrar a la **función Intercambiar**, el sistema muestra todos los productos ofertados por otros usuarios para trueque y que no se han intercambiado. La **figura 4.30** muestra el código en php que consulta mediante sentencias MySQL los productos ofertados por otros usuarios, sus productos preferentes de intercambio, el punto de encuentro preferido y el contacto de cada usuario dueño de cada producto. El resultado de la consulta se muestra en una tabla, como la de la **figura 4.31**, en donde además se muestra la columna **“Elegir”** y por cada artículo se muestra un enlace en esta columna para que el usuario activo pueda

Cuando el usuario activo selecciona un producto de otro usuario para trueque y entra al link del producto solicitado, el sistema muestra la pantalla que generara la solicitud de trueque (Figura 4.32), en donde el usuario podrá seleccionar de su lista de productos pendientes de trueque el producto que ofrecerá a cambio y enviara la solicitud de trueque con el botón “Enviar”. La Figura 4.33 muestra el código php que genera la pantalla de solicitud de trueque de la figura 4.32, en donde el usuario activo deberá seleccionar uno de sus productos a ofrecer a cambio del producto que selecciono de otro usuario.



Figura 4.32 Pantalla que permite generar la Solicitud de Trueque

```

<?php
$idproducto=$_GET['idpro'];
$link=$_GET['link'];
$costo=$_GET['costo'];
// echo "Producto = $idproducto <br>";

include("conexion.php");
$link=Conectar();
$idusuario=$_SESSION['id_usuario'];
$result=mysql_query($link,"select * from productos where id_producto=$idproducto");
$row=mysql_fetch_array($result);
$pro=$row['producto'];
$des=$row['descripcion'];
$fof=$row['foto'];
$can=$row['cantidad'];
echo "<img src='imagenes/Productos/$fof' width='80' height='100' <br>";
echo "Producto: $pro <br>";
echo "Descripción: $des <br>";
echo "Preferencia del dueño: $can <br>";

echo "Lugar de Intercambio: $lug <br>";
echo "Contacto del dueño: $cont <br>";
echo "<br>";
?>

<?php
$idusuario=$_SESSION['id_usuario'];
echo "<form action='intercambiar.php' Method='POST'>";
$result=mysql_query($link,"select * from productos where id_usuario=$idusuario");
echo "<SELECT NAME='proo'>";
while($row2 = mysql_fetch_array($result2))
{
    $id=$row2['id_producto'];
    $pro=$row2['producto'];
    echo "<option value='$id.'> $pro.<br>";
}
echo "</SELECT>";
echo "<input type='hidden' name='prooSeleccionado' value='$idproducto' > ";
echo "<input type='hidden' name='link' value='$link' > ";
echo "<input type='hidden' name='costo' value='$costo' > ";
echo "<br><br><br><input type='SUBMIT' value='Enviar'>";
echo "</form>";
?>

```

Código encargado de mostrar los datos del producto de otro usuario seleccionado por el usuario activo para trueque

Código que muestra lista de productos del usuario activo, para que este seleccione uno para ofertar y envíe la solicitud de trueque

Figura 4.33 Código PHP que genera pantalla de Solicitud de Trueque con datos del Producto seleccionado por el usuario activo y le producto a ofertar a cambio

Confirmar Solicitud de Trueque (Botón Enviar y Procesar Cambio)

Cuando el usuario activo selecciona su artículo a ofertar en el trueque debe dar clic en el botón Enviar de la Figura 4.32 presentada en la página anterior, para que el sistema muestre la información de la solicitud de trueque, figura 4.34.

La Figura 4.34 muestra la información de ambos productos participantes en la solicitud de trueque y el enlace **“Procesar el Cambio”** para que el usuario confirme la creación de esta solicitud. La Figura 4.35 muestra el código php que crea la pantalla de la Figura 4.34.



Figura 4.34 Pantalla con los datos de ambos productos integrantes en la solicitud de trueque



Figura 4.35 código php que crea la pantalla de la Figura 4.34

Finalmente, una vez que el usuario confirma la creación de la solicitud de trueque se ejecuta el código php mostrado en la Figura 4.36, el cual cambia el estado de ambos productos a 2, que indica que están en proceso de trueque y agrega un registro en la tabla de cambios con los datos de la solicitud de trueque.

```
<?PHP
echo "<h4>Se envió con éxito la solicitud para intercambiar el producto seleccionado. </h4>";
$selec=$_GET['idSeleccionado'];
$sacambio=$_GET['idAcambio'];
/* echo "Producto seleccionado = $selec <br>";
echo "Producto a cambio = $sacambio <br>";*/

include("conexion.php");
$link=Conectarse();
mysql_query($link,"update productos set estatus=2 where id_producto=$selec");
mysql_query($link,"update productos set estatus=2 where id_producto=$sacambio");

$idusuario=$_SESSION["id_usuario"];
$remite=$_GET['remite'];
/* echo "Usuario solicitante= $idusuario <br>";
echo "Usuario remitente = $remite <br>";
echo "ID producto solicitado = $selec <br>";
echo "ID producto a Cambio = $sacambio <br>";*/

$scad="INSERT INTO cambios(id_usuario, id_remitente, id_productoSolicitado, id_productoAcambio, validar, fechaAplica)
VALUES($idusuario,$remite,$selec,$sacambio,0,CURDATE())";
mysql_query($link,$scad);
mysql_close($link);
?>
```

Figura 4.36 Código php que genera la Solicitud de Trueque

Para que quede funcionando el desarrollo de la **función Intercambiar**, para cada una de sus funciones, consulta de Productos ofertados para trueque por otros usuarios y creación y confirmación de Solicitud de Trueque del producto seleccionado se realizan Pruebas de Unidad (para asegurarse que el código funciona) y Pruebas de Cliente (para demostrar que cumple con las necesidades del cliente), además, se hacen las correcciones necesarias hasta obtener resultados satisfactorios.

4.3.2.5 Consultar y Aceptar /Rechazar Solicitudes de Trueque de otros usuarios (Función solicitudes)

La Función Solicitudes, permite al usuario activo poder consultar las solicitudes u ofertas de trueque que crearon otros usuarios interesados en intercambiar sus productos por sus productos para que él pueda aceptar o rechazar el intercambio.

4.3.2.5.1 Consultar Solicitudes de Trueque de otros usuarios

En la solicitud de trueque que se creó anteriormente para ejemplificar la función Intercambiar, el usuario Sofía Torres creó una solicitud de trueque, solicitando un uniforme de karate del usuario José Gilberto a cambio de su kit de teclado y ratón. Cuando el usuario José Gilberto entra a la función Solicitudes, se mostrará una pantalla como la de la Figura 4.37 que muestra la pantalla de Solicitudes del usuario José Gilberto, en donde ya aparece la solicitud de Sofía Torres. Desde esta pantalla José Gilberto puede Aceptar o Rechazar el Trueque de Sofía Torres.

La figura 4.38 presenta el código php que consulta en la base de datos de trueques las solicitudes de pendientes de trueque que realizaron otros usuarios, solicitando productos del usuario activo a cambio de sus productos. Esta información se vacía en una tabla que se presenta en la página de solicitudes mostrada en la Figura 4.37 para que el usuario activo pueda revisar estas solicitudes y poder aceptar o rechazar.

Solicitudes de intercambios

Solicitudes para intercambiar tus productos. Marca la casilla en los productos que aceptes intercambiar



producto solicitado	Solicitante	Producto a Cambio	Descripción	Fecha	Cambios
 Uniforme Karate	sofyanzuelo@gmail.com	 Kit Teclado y Mouse Rosa	Kit Teclado-Mouse color Rosa	2022-02-06	Aceptar el cambio Rechazar el cambio

Figura 4.37 Solicitudes de Trueque de productos del usuario activo realizadas por otros usuarios.

```

<?PHP
include("conexion.php");
$link=Conectarse();
echo "<h3> Solicitudes para intercambiar tus productos. Marca la casilla en los productos que aceptes intercambiar </h3>";
$idusuario=$SESSION["id_usuario"];
// echo "Usuario= $idusuario <br>";

echo"<table class='table' border='1'>";
echo"<tr><th>producto solicitado </th><th> Solicitante </th><th> Producto a Cambio </th><th> Descripción </th>";
echo"<th> Fecha </th><th> Cambios </th></tr>";

$result=mysqli_query($link,"select * from cambios where id_solicitante=$idusuario and validar=0");
while($row = mysqli_fetch_array($result))
{
    $prodSolicitado=$row["id_productoSolicitado"];
    $prodAcambio=$row["id_productoAcambio"];
    $fecha=$row["fechaAlpila"];
    $idCambio=$row["id_cambio"];
    // echo "id producto solicitado: $prodSolicitado <br>";
    $result2=mysqli_query($link,"select producto.foto from productos where id_producto=$prodSolicitado");
    $row2=mysqli_fetch_array($result2);
    $nombreProdSolicitado=$row2["producto"];
    $fotoSolicitado=$row2["foto"];
    $result3=mysqli_query($link,"select producto.foto, id_usuario, descripcion from productos where id_producto=$prodAcambio");
    $row3=mysqli_fetch_array($result3);
    $nombreProdAcambio=$row3["producto"];
    $fotoAcambio=$row3["foto"];
    $idUsuarioSolicitante=$row3["id_usuario"];
    $descripcionSolicitante=$row3["descripcion"];
    // echo "Producto seleccionado: $nombreProdAcambio <br>";
    $result4=mysqli_query($link,"select mail from usuarios where id_usuario=$idUsuarioSolicitante");
    $row4=mysqli_fetch_array($result4);
    $contactoSolicitante=$row4["mail"];

    echo"<tr><td> <img src='imagenesProductos/$fotoSolicitado' width='40' height='80'><br> <nombreProdSolicitado/>";
    <td> <contactoSolicitante </td>";
    <td> <img src='imagenesProductos/$fotoAcambio' width='40' height='80'> <br> <nombreProdAcambio/>";
    <td> <descripcionSolicitante </td>";
    <td> <fecha </td> <td> <a href='aceptarCambio.php?idCambio=$idCambio'>Aceptar el cambio </a><br>";
    <td> <a href='rechazarCambio.php?idCambio=$idCambio&idProdSoli=$prodSolicitado&idProdAcam=$prodAcambio'> Rechazar el cambio </a></td>";
}

echo "</table>";
mysqli_close($link);
?>

```

Obtiene todas las Solicitudes de Trueque que otros usuarios le hicieron al usuario activo

Obtiene los datos de los productos solicitados por otros usuarios al usuario activo

Obtiene los datos de los productos a cambio que ofrecen los usuarios solicitantes

La información obtenida de todas las solicitudes se muestra en una tabla, para que el usuario visualizara y Aceptar o Rechazar las Solicitudes de Trueque.

Figura 4.38 Código php que consulta en la base de datos de trueques las solicitudes de pendientes de trueque creadas por otros usuarios para el usuario activo

4.3.2.5.2 Aceptar Solicitud de Trueque (“Aceptar cambio”)

Una vez que el usuario activo reviso en la página de Solicitudes, las solicitudes de trueque que le realizaron otros usuarios, por cada solicitud mostrada, hay un enlace para “**Aceptar el cambio**” y otro link para “**Rechazar el cambio**”, marcadas en recuadro rojo en la Figura 4.38.

Si el usuario activo decide aceptar alguna solicitud de trueque generada por otro usuario debe entrar al enlace “**Aceptar el cambio**”, marcado en el recuadro azul de la Figura 4.39 para confirmar que acepta la solicitud de trueque, el sistema cambiara el estatus de la solicitud de trueque a **1 = Solicitud Aceptada**, y el estatus del producto del remitente y del producto de solicitante a **0 = Intercambiado** quedando de esta forma confirmado el trueque de los productos.



producto solicitado	Solicitante	Producto a Cambio	Descripción	Fecha	Cambios
 Uniforme Karate	sofyanzuelo@gmail.com	 Kit Teclado y Mouse Rosa	Kit Teclado-Mouse color Rosa	2022-02-06	Aceptar el cambio Rechazar el cambio

Figura 4.39 Solicitud de Trueque – Confirmar Trueque

La Figura 4.40 muestra el código php que realiza los cambios de estado en la solicitud de trueque y en los productos involucrados para que quede realizado el trueque, además de confirmar al usuario la operación del trueque (Figura 4.41).

```
<?PHP
    echo "<h4>Se envió con éxito la Confirmación de aceptación de la Solicitud de Trueque seleccionada. </h4>";

$idcambio=$_GET['idCambio'];
echo "Cambio Aceptado Id Cambio: $idcambio<br>";
$idProdSolicitado=$_GET['idProdSoli'];
$idProdAcambio=$_GET['idProdAcam'];
echo "Producto Solicitado = $idProdSolicitado <br>";
echo "Producto a Cambio = $idProdAcambio <br>";
include("conexion.php");
$link=Conectarse();
mysqli_query($link,"update cambios set validar=1 where id_cambio=$idcambio");
mysqli_query($link,"update productos set estatus=0 where id_producto=$idProdSolicitado");
mysqli_query($link,"update productos set estatus=0 where id_producto=$idProdAcambio");

// header("Location: solicitudes.php");
?>
```

Figura 4.40 Código PHP que realiza la confirmación del trueque



Figura 4.41 Confirmación de Trueque de Productos

4.3.2.5.3 Rechazar Solicitud de Trueque (“rechazar cambio”)

Si el usuario activo decide rechazar alguna solicitud de trueque generada por otro usuario debe entrar al enlace “**Rechazar el cambio**”, marcado en el recuadro azul de la **Figura 4.42** para confirmar que rechaza la solicitud de trueque, el sistema cambiara el estatus de la solicitud de trueque a **2 = Solicitud Rechazada**, y el estatus del producto del remitente y del producto del solicitante a **1 = Libre** quedando libres para una próxima solicitud de trueque.



Figura 4.42 Solicitud de Trueque – Rechazar Trueque

La **Figura 4.43** muestra el código php que realiza los cambios de estado en la solicitud de trueque y en los productos involucrados para que quede rechazada la solicitud de trueque y los productos involucrados queden libres para una futura solicitud. Finalmente, la **Figura 4.44** muestra la pantalla de confirmación de Solicitud de Trueque rechazada.

```

<?PHP

    echo "<h4>Se envió con éxito la Confirmación para Rechazar la Solicitud de Trueque seleccionada. </h4>";

$idcambio=$_GET['idCambio'];
echo "Cambio Rechazado Id Cambio: $idcambio<br>";
$idProdSolicitado=$_GET['idProdSoli'];
$idProdAcambio=$_GET['idProdAcam'];
echo "Producto Solicitado = $idProdSolicitado <br>";
echo "Producto a Cambio = $idProdAcambio <br>";
include("conexion.php");
$link=Conectarse();
mysqli_query($link,"update cambios set validar=2 where id_cambio=$idcambio");
mysqli_query($link,"update productos set estatus=1 where id_producto=$idProdSolicitado");
mysqli_query($link,"update productos set estatus=1 where id_producto=$idProdAcambio");

    //    header("Location: solicitudes.php");
    |
?>

</p>
<p align="left" class="indent">&nbsp;  </p>

```

Figura 4.43 Código PHP que realiza el rechazo del trueque



Figura 4.44 Confirmación de Rechazo de Trueque de Productos

Para que quede funcionando el desarrollo de la **Función Solicitudes (Aceptar o Rechazar Trueques)**, se realizan Pruebas de Unidad (para asegurarse que el código funciona) y Pruebas **de Cliente** (para demostrar que cumple con las necesidades del cliente) y se hacen las correcciones necesarias hasta obtener resultados satisfactorios.

4.4 Poner en Marcha el Prototipo de iTrueque

- 1.1. Registro del dominio ***iTrueque.com.mx***
- 1.2. Contrato de servicio de alojamiento Linux, esto es; el espacio contratado en un servidor para poder poner online la aplicación, con la contratación se obtiene:
 - **Nombre del servidor (Host):** nombre que hay que utilizar para hacer referencia al servidor donde estará hospedada la página (p.e. **ftp.itrueque.com.mx**)
 - **Nombre de Usuario**
 - **Contraseña**
- 1.3. Acceso al panel de control (**cpanel**) en donde se gestiona el alojamiento y la creación de la base de datos.
- 1.4. Utilización del protocolo FTP por medio de la aplicación gratuita **FileZilla** para conectarse al servidor, se requiere el nombre del servidor, el usuario y la contraseña.
- 1.5. Subir todo el contenido de la web **iTrueque** del equipo de desarrollo hacia la carpeta de alojamiento proporcionada por el proveedor del servicio.
- 1.6. Finalmente, solo resta que los internautas accedan a la Web de Intercambio de Productos **iTrueque** desde cualquier computadora con acceso a Internet mediante el link **www.itrueque.com.mx**.

5. Conclusión

El objetivo inicial del proyecto, de poder construir o desarrollar un producto de Software para Intercambio de Productos de forma gratuita queda concluido y plasmado en **iTrueque**, un prototipo funcional del Sistema de Intercambio de Productos, el cual se implementó como un Sistema Web que permite a cualquier internauta consultar productos ofertados para trueque por usuarios registrados en el sistema, si el internauta se interesa por algún producto, deberá registrarse como usuario del sistema para poder agregar sus productos y solicitar el intercambio de estos por productos de otros usuario y en caso de aceptar alguno de estos usuarios la solicitud de trueque, el intercambio se efectúa entre ambos usuarios.

Se cumple además el objetivo de poder contribuir con el cuidado del medio ambiente y la sustentabilidad del planeta al fomentar mediante **iTrueque** la reutilización de productos que ya son basura o estorbo para algunas personas, pero que pueden ser útiles para otras personas, contribuyendo de esta forma a la construcción de una cultura de reciclaje y reutilización que lleva a la reducción de basura y de un consumismo innecesario.

Finalmente se cumple también el objetivo de construir un prototipo de software siguiendo una metodología de software, la metodología RUP (**Rational Unified Process o Proceso Unificado de Racional**), metodología para el desarrollo de Software de calidad que se caracteriza por ser iterativa e incremental. La construcción del prototipo se realizó de forma iterativa en cada una de las fases, **Fase de Inicio** (definición del modelo y alcance del producto), **Fase de Elaboración** (analizar el problema inicial, diseñar la arquitectura y el plan del proyecto), **Fase de Construcción** (desarrollo de Base de Datos, desarrollo y pruebas de cada uno de los componentes del sistema) y **Fase de Transición** (colocar en manos del usuario final el producto). En cada una de estas fases se cumplió también el uso del lenguaje de modelado **UML** (LENGUAJE UNIFICADO DE MODELADO) para plasmar durante el análisis y diseño las funciones principales a implementar para **iTrueque**, estos modelos UML fácilmente se tradujeron en código fuente quedando desarrolladas e implementadas cada una de las funcionalidades de **iTrueque**.

Cabe mencionar además que el prototipo funcional deja actividades pendientes para completar o enriquecer la aplicación en un trabajo futuro, las cuales se enumeran en el punto siguiente y se basan en el diseño inicial del Sistema.

5.1 Trabajo Futuro

1. **Desarrollo Completo del Diseño Inicial de iTrueque** en donde se contempla que:
 - Más de un usuario puede solicitar el intercambio de un producto de otro usuario específico, de tal forma que el usuario dueño del producto solicitado pueda decidir

entre varias solicitudes cual aceptar; esto es, por cual producto intercambiar su producto de una lista de productos candidatos.

- Agregar más puntos de encuentro para efectuar los intercambios físicos.
 - Enviar notificaciones a los usuarios de nuevas solicitudes de trueque vía correo electrónico y/o whatsapp.
2. **Rediseñar los nombres de las tablas de la base de datos de iTrueque** para que no sean tan obvios o explícitos como lo son: “usuarios”, “clientes” y “productos”, de tal forma que un hacker pueda dañar la información, se colocarían nombres de tablas como en clave y para hacerlo claro al administrador de la base de datos o al desarrollador, deberá agregar una tabla de mapeo de los nombres explícitos contra los nombres implícitos, de forma que no se complique el trabajo del equipo técnico y tampoco se arriesgue la seguridad de la información.
 3. **Crear un respaldo de la tabla usuarios por medio de un Disparador (Trigger) desde SQL** de forma que cada que se realice un cambio a algún registro de la tabla de usuarios se respalde el valor original en otra tabla de modo que, en caso de alguna equivocación al actualizar la información, el usuario pueda recuperarla.
 4. **Agregar Herramientas tipo e-commerce a iTrueque**, como por ejemplo un carrito de compras, iTrueque lo permite, pues finalmente es una aplicación tipo **e-commerce** ya que, aunque no se maneje dinero como forma de pago, la forma de pago es con otro producto y este intercambio, se puede considerar como una compra-venta entre dos personas a través de una plataforma en línea.
 5. **Asegurar la Capa de Datos al agregar usuarios en la Base de Datos de MySQL de acuerdo con cada Rol** y gestionar permisos a las tablas de la base de datos de acuerdo con cada rol, esto con el objetivo de proteger la información a la que puede tener acceso cada usuario de acuerdo con su rol desde la capa de aplicación. Se realizará una gestión de permisos por rol como la de la Figura 5.1.

Capa de Datos

Permisos a Base de Datos Intercambio X Rol												
Rol	Internauta				Usuario Registrado				Usuario Administrador			
Tabla	Select	Insert	Update	Delete	Select	Insert	Update	Delete	Select	Insert	Update	Delete
Usuarios					X	X	X		X	X	X	X
Direcciones					X	X	X		X	X	X	X
Productos	X				X	X	X		X	X	X	X
Solicitud Trueque					X	X	X		X	X	X	X
Productos Candidatos					X	X	X		X	X	X	X
Lugar de Entrega					X	X	X		X	X	X	X
Confirmación Trueque					X	X	X		X	X	X	X

Figura 5.1 Gestión de Permisos X Rol para Base de Datos iTrueque en MySQL

6. **Implementar inicio de sesión a iTrueque utilizando el funcionamiento Captcha** (prueba de Turing inversa) para asegurar el sitio contra spam o bots.

6. Referencias.

6.1 Bibliografía

- [1] Carlos Coronel, Steven Morris, Peter Rob. Bases de datos: diseño, implementación y administración, México 2002.
- [2] Braude, Ingeniería de Software, una perspectiva orientada a objetos, Alfaomega, México 2003.
- [3] Abraham Gutiérrez, Ginés Bravo, PHP5, Alfaomega Ra-MA, México 2005.
- [4] Helma Spona, Programación de bases de datos con MySQL y PHP, Alfaomega, México, 2010.
- [5] Silberschatz, Korth, Sudarshan, Fundamentos de Diseño de Bases de Datos, quinta edición, Mc Graw Hill, España, 2007.
- [6] Jacobo Pavon Puertas. Creación de un portal con PHP y MySQL.4ta Edición. Alfaomega. 2010.
- [7] Ian Gilfillan. La biblia MySQL. Editorial ANAYA, 2003.
- [8] Steve Suehring, Janet Valade. PHP, MySQL, JavaScript & HTML5 All-in-One For Dummies. 1st Edición. 2013.
- [9] Elizabeth Freeman & Erick Freeman. Head First HTML with CSS & XHTML–2006.
- [10] Luke Welling, Laura Thomson. Desarrollo Web con PHP y MySQL Editorial Anaya. 2009
- [11] Philippe Kruchten, The Rational Unified Process An Introduction Third Edition, Addison-Wesley
- [12] Gabriel Baca Urbina, Proyectos de Sistemas de Información, Primera Edición Ebook, México 2015 FCC BUAP
SECRETARÍA ACADÉMICA
- [13] Ian Sommerville, Ingeniería del software, Séptima edición, Pearson Educación. S.A. Madrid 2005
- [14] Booch, Grady / Rumbaugh, James / Jacobson, Ivar. E Proceso Unificado De Desarrollo De Software. Addison Wesley, 2000

6.2 Mesografía

- [15] EL TRUEQUE EN LA ACTUALIDAD. ¡Barato... muy barato! Cheap... very cheap! <https://baratomuybarato.wordpress.com/2012/12/19/el-trueque-en-la-actualidad/>. Consultado el 22 de octubre de 2020.
- [16] Hula App LLC. Hula Trading. [2018 © By. Hula App LLC] <https://hula.trading/es/>. Consultado el 22 de octubre de 2020.
- [17] ¿Tienes cosas que ya no usas o ya no quieres? Haz trueque con esta app mexicana. La Verdad Noticias. <https://laverdadnoticias.com/tecnologia/Tienes-cosas-que-ya-no-usas-o-ya-no-quieres-Haz-trueque-con-esta-app-mexicana-20200127-0297.html>. Consultado el 22 de octubre de 2020.
- [18] Truequeo. ©2017 truequeo.com. <http://www.truequeo.com/>. Consultado el 23 de Octubre de 2020. [19] <https://kinsta.com/es/base-de-conocimiento/que-es-apache/>
- [23] Microsoft 365 Team, septiembre 24, 2019 <https://www.microsoft.com/es-mx/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling>