**Documento de Especificación de Requisitos 1**

|  |  |  |
| --- | --- | --- |
| **Grupo** | 3 | |
| **Proyecto** | “TreeSolution” (TS) | |
| **Especificación de Requisitos** | 1 | |
| **Requisito Funcional** | Inicio de Sesión | |
| **Encargado** | Cjuno Chacca, Roggelio | |
| **Control de Versión** | 1.0 | 22/04/2024 |

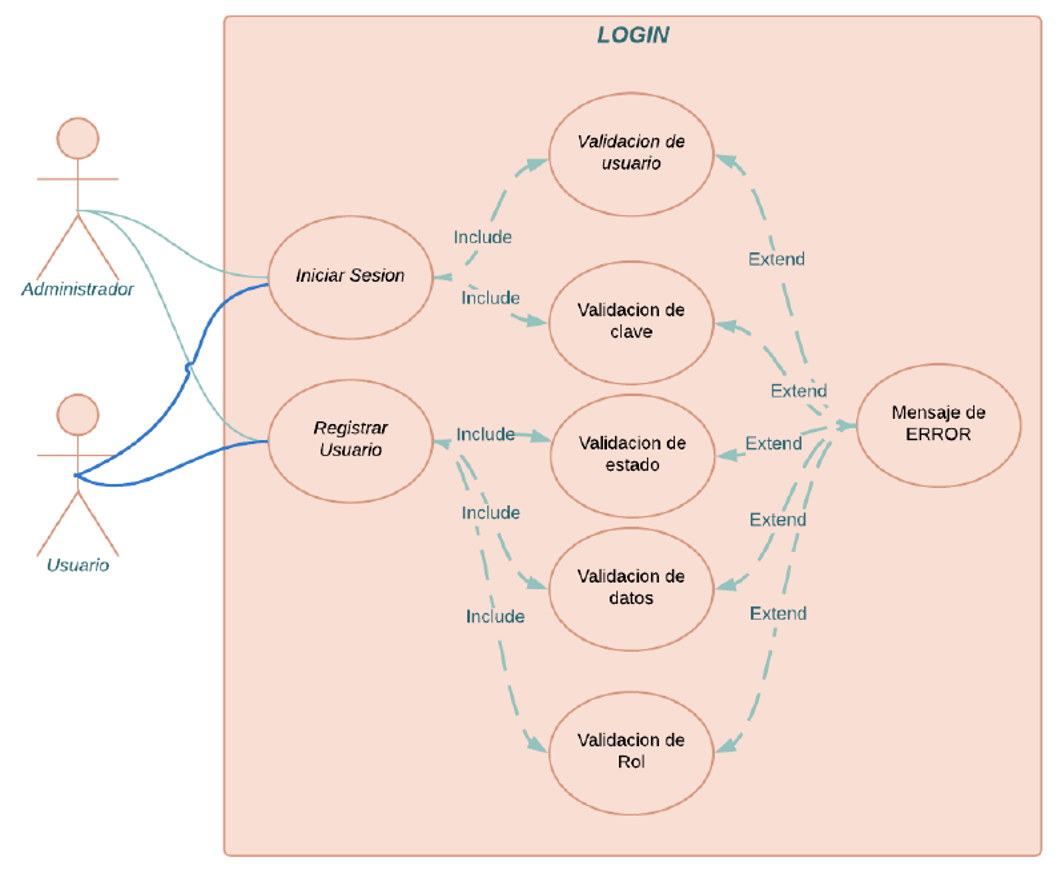
1. **Requisito Funcional**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Usuario** |
| RF001 | Sistema de inicio de sesión | El sistema debe permitir el registro y el inicio de sesión al sistema. | Gerente, Empleado |

1. **Caso de Uso : Sistema de inicio de sesión**
   1. **Descripción**:

El sistema de inicio de sesión permite al gerente y empleados autenticarse de manera segura en la aplicación de software. Este requisito define los comportamientos necesarios para garantizar un acceso controlado y protegido al sistema por parte de ambos tipos de actores..

* 1. **Actores**
     1. **Principales**
        1. **Gerente**: Representa al que administra todo el sistema informático.
     2. **Secundarios**
        1. **Usuario** : Representa a las entidades externas que solicitan productos de la empresa, en este caso ropa deportiva, y están sujetas a ser facturadas por ellos.
  2. **Condiciones**
     1. **Precondiciones**
* El sistema está en un estado operativo y disponible para su uso.
* El empleado tiene credenciales válidas para acceder al sistema .
  + 1. **Postcondiciones**
* El usuario ha sido autenticado exitosamente por el sistema.
* Se ha concedido al usuario acceso a las funciones y datos autorizados según su rol y nivel de permisos.
* Se ha registrado la actividad de inicio de sesión, incluyendo la fecha, hora y la identificación del usuario, para fines de auditoría y seguimiento.
* Se ha cargado la interfaz de usuario principal o la pantalla de inicio correspondiente al usuario autenticado.
  1. **Diagrama de Caso de Uso**



1. **Requisitos No Funcionales**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Nombre** | **Descripción** | **Usuario** |
| RNF1 | Seguridad | * El sistema debe utilizar un protocolo de autenticación seguro, como HTTPS, para proteger las credenciales del usuario durante la transmisión. * Debe implementarse un mecanismo de protección contra ataques de fuerza bruta para evitar intentos repetidos de inicio de sesión con credenciales incorrectas. * Se deben seguir las mejores prácticas de seguridad de contraseñas, como el uso de contraseñas fuertes, el almacenamiento seguro de contraseñas y la implementación de políticas de caducidad y cambio de contraseñas. | Empleado, Cliente |

1. **Codificación del CU1:**
   1. HTML:

|  |
| --- |
| <!DOCTYPE html>  <html lang="es">  <head>  #<!-- fzxczxczczxcqqqqqaaaaaaaaaaaaaaaaa -->  <meta charset="UTF-8">  <meta name="viewport" content="width=device-width, initial-scale=1.0">  <title>TreeSolution</title>  <link rel="stylesheet" href="/Public/estilos/index.css">  </head>  <body>  <header class="header">  <div class="header\_\_logo">TreeSolution</div>  <nav class="header\_\_nav">  <ul class="header\_\_nav-list">  <li class="header\_\_nav-item"><a href="#" class="header\_\_nav-link">Inicio</a></li>  <li class="header\_\_nav-item"><a href="#" class="header\_\_nav-link">Sobre nosotros</a></li>  <li class="header\_\_nav-item"><a href="#" class="header\_\_nav-link">Contáctanos</a></li>  </ul>  </nav>  <div class="header\_\_actions">  <input type="text" class="header\_\_search" placeholder="Buscar productos">  <button class="header\_\_search-button"><img src="/Public/imagenes/lupa.png" alt="Buscar" class="header\_\_search-icon"></button>  <button class="header\_\_button" id="registrarButton">Registrarse</button>  <button class="header\_\_button" id="loginButton">Iniciar Sesión</button>  <button class="header\_\_cart-button"><img src="/Public/imagenes/cart-icon.png" alt="Carrito" class="header\_\_cart-icon"></button>  </div>  </header>  <main class="main">  <div class="main\_\_overlay"></div>  <section class="hero">  <div class="hero\_\_content">  <h1 class="hero\_\_title">Diseñada para los que no se rinden: supera tus metas con cada prenda, llevando estilo y funcionalidad al siguiente nivel</h1>  <p class="hero\_\_text">Descubre en nuestra tienda virtual una selección exclusiva de ropa deportiva diseñada para maximizar tu rendimiento y mantenerte a la vanguardia del estilo y la comodidad</p>  <button class="hero\_\_button" id="exploButton">Explorar productos</button>  </div>  </section>  </main>  <footer class="footer">  <p>&copy; 2024 TreeSolution. Todos los derechos reservados.</p>  </footer>  <script src="/Public/scripts/index.js"></script>  </body>  </html> |

* 1. CSS:

|  |
| --- |
| body {  font-family: "Raleway", sans-serif;  display: flex;  justify-content: center;  align-items: center;  height: 100vh;  margin: 0;  background: url("../images/v24\_166.png") no-repeat center center;  background-color: #e9e9e9;  }    .container {  display: flex;  flex-direction: column;  align-items: center;  width: 500px;  background: #24303c;  background: rgba(36, 48, 60, 0.9); /\* Semi-transparent background \*/  padding: 30px;  margin: auto;  margin-top: 100px;  border-radius: 4px;  font-family: "Raleway";  color: #ffffff;  box-shadow: 7px 13px 37px #000;  text-align: center;  font-size: 14px;  }    .form-container {  flex: 1;  padding: 40px;  }    .form-container h1,  .form-container h2 {  margin-bottom: 20px;  }    .form-group {  margin-bottom: 15px;  }    .form-group label {  display: block;  margin-bottom: 5px;  }    .form-group input {  width: 100%;  padding: 10px;  box-sizing: border-box;  border: 1px solid #1f2124;  border-radius: 5px;  background: #1f2124;  color: #dbdcdb;  font-size: 13px;  }    .password-container {  position: relative;  width: 100%;  }    .password-container input[type="password"],  .password-container input[type="text"] {  width: 100%;  padding-right: 40px; /\* Espacio para el icono \*/  }    .password-container i {  position: absolute;  top: 50%;  right: 10px;  transform: translateY(-50%);  cursor: pointer;  color: #ccc;  }    .form-group button {  width: 100%;  padding: 10px;  background: #ffc340;  border: none;  border-radius: 5px;  color: #000000;  font-size: 14px;  cursor: pointer;  margin-top: 10px;  }    .form-group button:hover {  background: #ffc340;  }    .social-buttons {  display: flex;  justify-content: space-between;  margin-top: 20px;  }    .social-buttons button {  width: 48%;  padding: 15px;  background: #db4437;  border: none;  border-radius: 5px;  color: #ffffff;  font-size: 14px;  cursor: pointer;  display: flex;  align-items: center;  justify-content: center;  }    .social-buttons .facebook {  background: #3b5998;  color: #ffffff;  }    .social-buttons button img {  margin-right: 10px;  width: 20px;  height: 20px;  }    .links {  display: flex;  justify-content: space-between;  margin-top: 20px;  }    .links a {  color: #00aae4;  text-decoration: none;  }    .links a:hover {  text-decoration: underline;  }    .additional-text {  text-align: center;  margin: 20px 0;  }    .register-link {  text-align: center;  margin-top: 20px;  }  .register-link a {  color: #00aae4; /\* Cambia el color del enlace \*/  text-decoration: none;  }    .register-link a:hover {  color: #00aae4; /\* Cambia el color del enlace al pasar el cursor \*/  text-decoration: underline;  } |

* 1. PHP:
     1. Código backend: LoginController.php

|  |
| --- |
| <?php  namespace Controllers;  use MVC\Router;  use Model\User;  use Model\Cliente;  class LoginController{  public static function login(Router $router){    if( $\_SERVER['REQUEST\_METHOD'] === 'POST'){  //creamos un objeto User con los datos enviados a trabes del formulario(post)  $usuariotmp = new User($\_POST);  //buscamos un Usuario donde el email sea igaul al ingresado por form  $usuario = User::where('email', $usuariotmp->email);  if($usuario){  //Comprobamos el password del usuario correspondiante al Email, con la contraseña ingresada en form:  if($usuario->comprobarPassword($usuariotmp->contrasena)){  session\_start();  $\_SESSION['id'] = $usuario->ID\_Usuario;  $\_SESSION['nombre'] = $usuario->nombre . " " . $usuario->apellido;  $\_SESSION['email'] = $usuario->email;  $\_SESSION['login'] = true;  //redireccionar  if($usuario->usuario == "admin"){  $\_SESSION['admin'] = true;  header('Location: /admin');  }else{  header('Location: /dashboard');  }  }else{  User::setAlerta('error', 'Password Incorrecto');  }  }else{  User::setAlerta('error', 'Correo no registrado');  }  }  if (isset($\_GET['estado'])) {  $estado = $\_GET['estado'];  if($estado = "registroExitoso")  User::setAlerta('success', 'Usuario registrado exitosamente');  }  $alertas = User::getAlertas();  $router->render('Login', [  'alertas' => $alertas  ]);  }  public static function logout(){  session\_start();  $\_SESSION = [];  header('Location: /');  }  public static function crearCuenta(Router $router){  if( $\_SERVER['REQUEST\_METHOD'] === 'POST'){  $usuariotmp = new User($\_POST);    //Comprobamos que el correo no exista en otro usuario  $usuario = User::where('email', $usuariotmp->email);    if($usuario){  //alerta, correo ya registrado.  debuguear('Correo ya registrado');  }else{  $resultado = $usuariotmp->guardar();  if($usuariotmp->usuario == 'cliente'){  $datosCliente = [  'ID\_Cliente' => '',  'ID\_Usuario' => $resultado['id'],  'direccion'=> 'casa'  ];  $clientetmp = new Cliente($datosCliente);  $resultado2 = $clientetmp->guardar();    }  if($resultado){  header('Location: /login?estado=registroExitoso');  }  }  }else{  $router->render('CrearCuenta');  }  }  } |

* 1. Resultado:

