



Universidad de las Fuerzas Armadas ESPE

**Departamento de Ciencias de la
Computación (DCC0)**

Carrera de Ingeniería Tecnología de la
Información
(En línea)

NRC 5451: Ingeniería de Software II

Integrantes del Grupo 5:

Karla Cazares, Fernando Carrera, Lizeth Iza,
Erick Maldonado, Jaime Rojas.

Docente:

Dr. Efraín Rodrigo Fonseca Carrera

29 de junio del 2022



Actividad Entregable 3

Introducción:

Actualmente es primordial para cualquier institución utilizar herramientas de software, en especial las aplicaciones web, ya que permite resguardo de la información de manera segura y el fácil acceso a la información desde cualquier punto de acceso de una red.

Las nuevas tecnologías de la información, traen consigo la innovación de nuevas estrategias para la educación, en este proyecto se propone la implementación y el monitoreo de una aplicación multimedia para usuarios, mediante herramientas de diseño básicas para potenciar su desarrollo digital y acrecentar su crecimiento formativo como lo marca el programa de educación escolar.

Y está enfocado para toda la comunidad educativa, personal administrativo, docentes y estudiantes que puedan acceder a su portal de una manera más ágil y veras y en tiempo real acceder a toda su información, calificaciones y otros datos, así como la generación de reportes e impresión de las mismas, facilitando al usuario una interfaz amigable para la interacción con la aplicación y un manejo sencillo.

Objetivo General:

Realizar un prototipo de sistema ejecutable, con el propósito de validar la arquitectura de software planteado.

Objetivos Específicos:

- Diseñar el software con las interfaces requeridas basados en el paradigma de reutilización.
- Depurar la arquitectura del sistema propuesta.
- Proponer actualizaciones en el software y realizar el mantenimiento
- Proponer casos de prueba de caja blanca y de caja negra

Desarrollo.

- 1. Plantee la problemática a ser abordada dando a conocer las características de su contexto y los requerimientos generales del sistema, considerando una posible evolución de estos.**

La Unidad Educativa requiere una aplicación web que le permita gestionar todos los procesos académicos, administrativos, y reportería enmarcados en el contexto de los lineamientos del Mineduc. Los requerimientos del aplicativo son los siguientes: Control de Usuarios, Gestión de calificaciones, Gestión de asignaturas, Gestión de periodos, Gestión de activos fijos, Reportería General.



2. Diseñe el software con las interfaces requeridas basados en el paradigma de reutilización.

Ingreso al Sistema

Usuario

Contraseña

INGRESAR

Registro de docentes al sistema

REGISTRO DE DOCENTES
Complete todos los campos para que el docente sea registrado

Nombres del docente *	Apellidos del docente *
<input type="text"/>	<input type="text"/>
Nivel de estudio *	Teléfono celular *
<input type="text"/>	<input type="text"/>
Dirección *	Correo electrónico *
<input type="text"/>	<input type="text"/>
Paralelo *	Cédula Docente *
<input type="text"/>	<input type="text"/>

Nombre de la Materia que imparte *	Año Lectivo *
<input type="text"/>	<input type="text"/>
Número de materias que imparte *	Salario por materia (\$) *
<input type="text"/>	<input type="text"/>
Nombre de usuario *	Contraseña *
<input type="text"/>	<input type="text"/>
Tipo de usuario *	Status *
<input type="text"/>	<input type="text"/>

REGISTRAR DOCENTE

Registro de materias al sistema

REGISTRO DE MATERIAS
Complete todos los campos para que la materia sea registrada

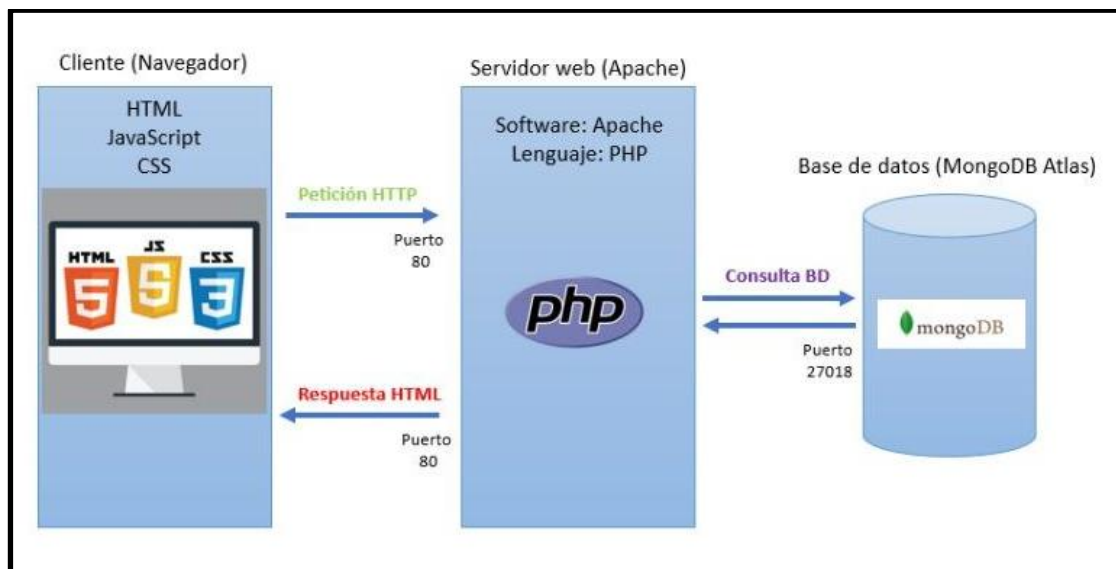
Nombre de la Materia *	Año Lectivo *
<input type="text"/>	<input type="text"/>
Departamento *	Nivel *
<input type="text"/>	<input type="text"/>
Paralelo *	Cédula Docente *
<input type="text"/>	<input type="text"/>

REGISTRAR MATERIA

Registro de estudiantes al sistema

REGISTRO DE ESTUDIANTES	
Complete todos los campos para registrar al Estudiante	
Nombres del estudiante *	Apellidos del estudiante *
<input type="text" value="Nombres del Estudiante"/>	<input type="text" value="Apellidos del Estudiante"/>
Teléfono celular *	
<input type="text" value="Ej: 0983525003"/>	
Dirección *	Correo electrónico *
<input type="text" value="Ej: Machala, Urbanización Unio"/>	<input type="text" value="ej: mathew2012@gmail.com"/>
Paralelo *	Cédula Estudiante *
<input type="text" value="ej: A-B-C-D"/>	<input type="text" value="ej: 0701990836"/>
Año Lectivo *	
<input type="text" value="ej: 2021-2022"/>	
Número de materias que estudia *	
<input type="text" value="Ej: 6"/>	
Nombre de usuario *	Contraseña *
<input type="text" value="Inicial del nombre y apellido, Ej:"/>	<input type="text" value="Número de cédula, Ej: 160089"/>
Tipo de usuario *	Status *
<input type="text" value="Ej: docente, alumno"/>	<input type="text" value="Ej: activo, inactivo"/>
<input type="button" value="REGISTRAR ESTUDIANTE"/>	

3. Depure la arquitectura del sistema propuesta.



La aplicación utilizará del lado del Cliente a JavaScript para lógica de las operaciones, HTML para crear la vista que se presentará al usuario, CSS para los estilos de la aplicación. PHP como un servidor web y base de datos MongoDB Atlas para almacenar información.

4. Proponga actualizaciones en el software y realice el mantenimiento inherente a las mismas.

En la actualización del software lo que se tiene previsto realizar son cambios en los estilos es por ello que se debe realizar un Mantenimiento de estilos donde se trabajaría solamente en el componente del CSS.

Adicionalmente se plantea la implementación de una API para el proceso de transacciones con la base de datos.

5. Proponga casos de prueba de caja blanca y de caja negra y realice las pruebas (automáticas o manuales).

Dado que el sistema debe gestionar los cursos y contenidos a ser accedidos por los usuarios el aplicativo debe permitir el ingreso al personal administrador, docentes y alumnos. Para realizar las actividades según el tipo de usuario al que pertenezcan en este caso se plantea las pruebas de caja blanca y caja negra para validar el acceso al aplicativo.

En este proyecto se plantean pruebas manuales mediante un análisis dinámico donde se analizará los fragmentos de código y procesos de comunicación entre los módulos según las variables que se le asigne a cada elemento, siguiendo este proceso podemos verificar los bucles, funciones y declaraciones en el código fuente.

Prueba de Caja Blanca – Acceso al aplicativo

Requisito Planteado N:001			
Usuario	Administrador	Nombre Historia	Acceso a la aplicación
Prioridad	Alta	Programador Responsable	Fernando Carrera
Descripción	Iniciar la aplicación e ingresar usuario y contraseña, los usuarios tendrán perfiles definidos como el de administrador, docentes y estudiantes.		
Validación	El sistema muestra la interfaz principal		

Validación de Usuario – Acceso a la aplicación

- Código Fuente : Lenguaje PHP – IDE: Visual Studio Code

```

06-Code > controller > validarusuario.php
1  <?php
2  require_once "../db/db.php";
3  $usuario = "jfcarrera3"; //$_POST['usuario'] ;
4  $password = "1711990836"; //$_POST['password'] ;
5  $query = new MongoDB\Driver\Query(array('user' => $usuario, 'password' => $password));
6
7  //Output of the executeQuery will be object of MongoDB\Driver\Cursor class
8  $cursor = $manager->executeQuery('gestionEducativa.usuarios', $query);
9  // Convert cursor to Array and print result
10 $users = $cursor->toArray();
11 $result = count($users);
12
13 if ($result != 0) {
14     echo 1;
15 }else{
16     echo 0;
17 };
18 ?>

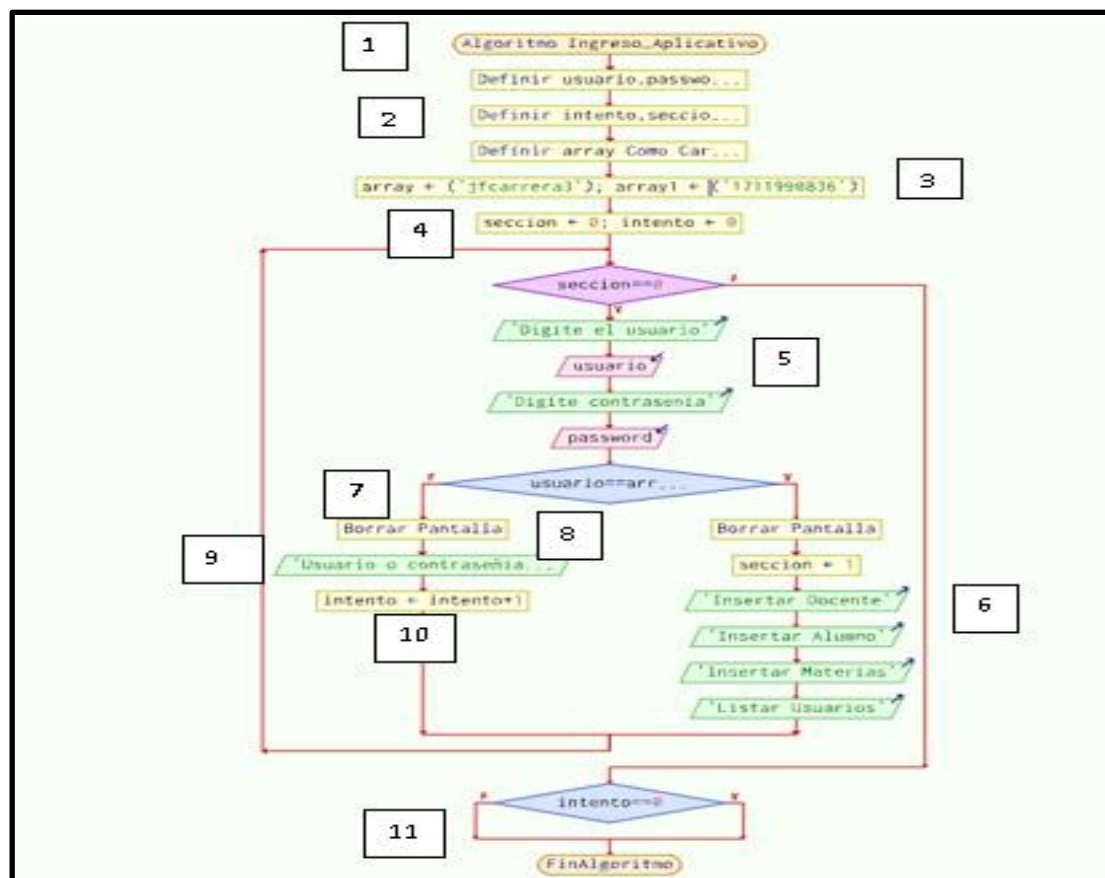
```

- Pseudocódigo

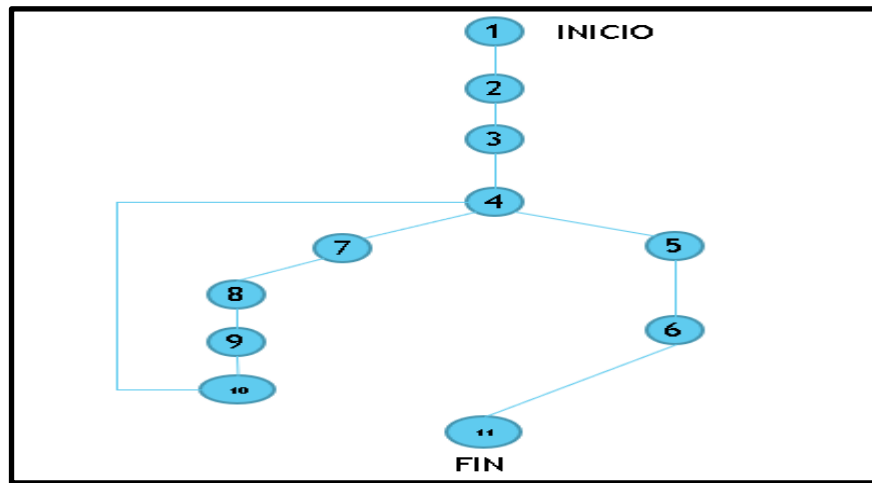
```

Ingreso_Aplicativo.psc* X
1  Algoritmo Ingreso_Aplicativo
2  Definir usuario,password Como Caracter
3  Definir intento, seccion Como Entero
4  Definir array Como Caracter
5  array ← ['jfcarrera3']; array1←['1711990836']
6  seccion ← 0 ; intento ← 0
7  Mientras seccion==0 Hacer
8  Escribir 'Digite el usuario'; Leer usuario
9  Escribir 'Digite contrasenia'; Leer password
10 Si usuario==array y password==array1
11     Borrar Pantalla
12     seccion ←1
13     Escribir 'Insertar Docente'
14     Escribir 'Insertar Alumno'
15     Escribir 'Insertar Materias'
16     Escribir 'Listar Usuarios'
17 Sino
18     Borrar Pantalla
19     Escribir 'Usuario o contraseña equivocado - Intente de nuevo'
20     intento← intento+1
21 FinSi
22 FinMientras
23 Si intento==0 Entonces
24     FinSi
25 FinAlgoritmo
    
```

- Diagrama de Flujo



- Grafo



Rutas: R1: 1, 2, 3, 4, 5, 6, 11
R2: 1, 2, 3, 4, 5, 7, 8, 9, 10, 11

Complejidad Ciclomática

Fórmula: $V(G) = A - N + 2$
$V(G) = \text{Aristas} - \text{Nodos} + 2$
$V(G) = 11 - 6 + 2 = 7$
Donde:
$V(G) = \text{Número de nodos predichados} = 6 + 1 = 7$

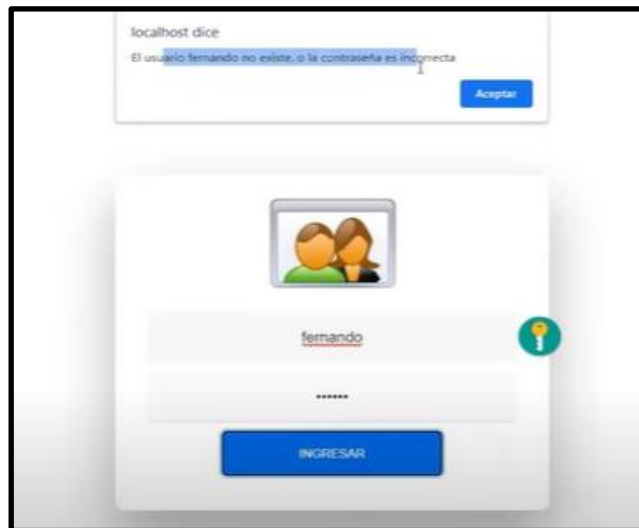
En conclusión, la complejidad ciclomática de la función validarusuario.php al encontrarse en una escala de 1-10, se considera que tiene una simplicidad Simple



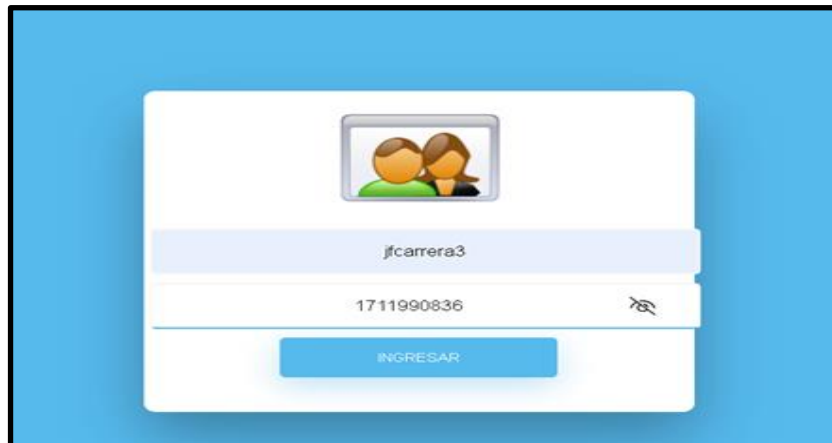
Prueba de Caja Negra

Variable	Clase de equivalencia	Estado	Representante
Usuario	EC1:document.form.login.v alue=='jfcarrera'	Válido	'jfcarrera'
	EC2:document.form.login.v alue!=='fernando'	No válido	Error Login opción no válida fernando Usuario no registrado
Contraseña	EC1:document.form.passw ord.value=='1711990836'	Válido	'1711990836'
	EC1:document.form.passw ord.value!=='1711990830'	No válido	Error en el ingreso contraseña ' 1711990830' Usuario no registrado
Validación	EC1:document.form.login.v alue==' jfcarrera3' && EC1:document.form.passw ord.value==' 1711990836'	Válido	'jfcarrera3' '1711990836'
	EC1:document.form.login.v alue=='fernando' && EC1:document.form.passw ord.value==' 1711990836'	No válido	Error en el ingreso usuario 'fernando' Usuario no registrado
	EC1:document.form.login.v alue=='1jfcarrera3' && EC1:document.form.passw ord.value=='1711990830'	No válido	Error password opción no válida '1711990830' Usuario no registrado
	EC1:document.form.login.v alue=='fernando' && EC1:document.form.passw ord.value=='1711990830'	No válido	Error Login y error password 'fernando' '1711990830' Usuario no registrado

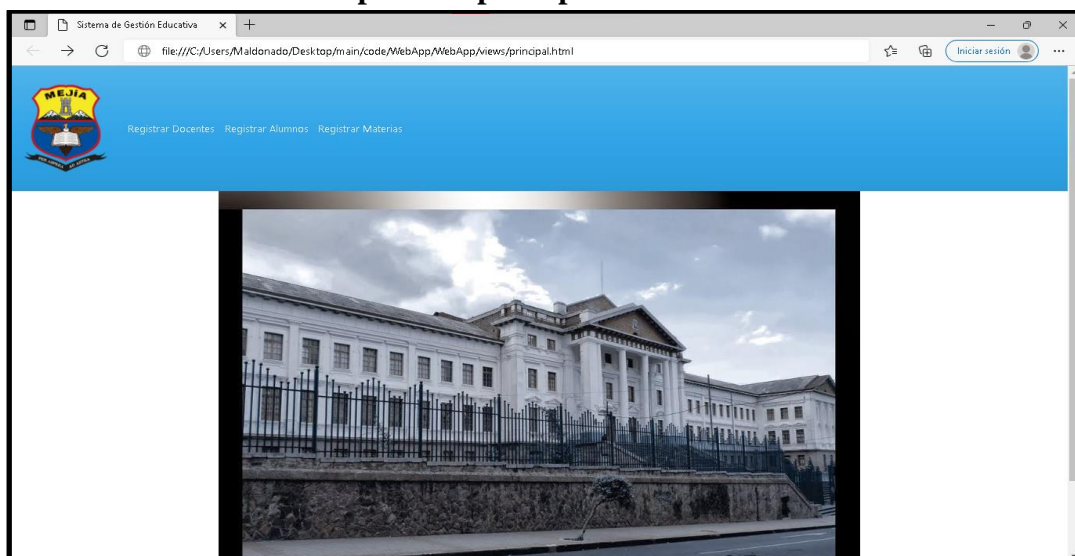
Captura Ingreso de Datos Incorrectos



Captura Ingreso de Datos Correctos



Captura Acceso Correcto a la pantalla principal





6. Realizar la presentación en línea del caso práctico.

<http://34.125.165.209/>

Usuario: jfcarrera

Contraseña: 1711990836

Conclusiones:

- Las pruebas de caja blanca se desarrollaron para verificar que la relación entre la entrada y la salida es correcta en las funciones empleadas para esta técnica, no sólo evaluando el comportamiento del usuario hacia la interfaz, sino también buscando errores en el código fuente lo que permitirá la retroalimentación en la etapa de desarrollo.
- Las Apps que fomentan los campos formativos deben ser evaluadas metódicamente antes de ponerse en marcha en niños de preescolar, pues cada niño aprende de distinta manera y saber llevar el aprendizaje móvil a este campo es una nueva forma de aprovechar las Tics para acceder al conocimiento mediante dispositivos móviles y mejorar el desempeño de los alumnos, pero deben habilitarse dispositivos adecuados a los pequeños, así como adecuar actividades educativas favorables que pueden soportar éstas herramientas digitales y sean de gran ayuda para el estudiante.
- En el paradigma de monitoreo e implementación de las Apps, refiere a que M-learning o aprendizaje móvil, debe aprovechar al máximo las nuevas tecnologías móviles para facilitar, apoyar y mejorar el proceso de enseñanza-aprendizaje y control de los usuarios.
- La creación de nuestra Aplicación educativa contiene grandes rasgos, un nivel de programación por parte del desarrollador que sea sofisticada y de calidad, puesto que las situaciones didácticas a desarrollar e implementar, deberán tener calidad en el diseño y abarcar contenidos de acuerdo a la administración del sistema

Bibliografía.

- *Introducción con MVC*. (2022, April 11). Microsoft Docs. Retrieved July 2, 2022, from <https://docs.microsoft.com/es-es/aspnet/mvc/overview/older-versions-1/getting-started-with-mvc/>
- *Manual de PHP - Manual*. (n.d.). PHP. Retrieved July 2, 2022, from <https://www.php.net/manual/es/index.php>

Enlace del video:

<https://www.youtube.com/watch?v=k3oysOqLGZU>