

**MANUAL TÉCNICO
PRACTICA [IPC1]PRACTICA2**

**SERGIE DANIEL ARIZANDIETA YOL
CARNET 202000119**

**UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
ESCUELA DE CIENCIAS Y SISTEMAS
Guatemala, 14 de Diciembre del 2020**

INTRODUCCIÓN

El objetivo de este manual es presentar el software “Practica [IPC1]Practica2”.

Practica [IPC1]Practica2 está creado para la organización y asignación de dichos usuarios con acceso al sistema.

Practica [IPC1]Practica2 está desarrollado en Java, tiene métodos para crear reportes en HTML de las operaciones realizadas por usuario.

Practica [IPC1]Practica2 permite la gestión de usuarios y la carga de archivos para la utilización a los métodos creados.

En el manual se encontrará la descripción de los métodos y clases utilizado para el diseño de Practica [IPC1]Practica2

I. Clases por paquete

1. Principal

1.1. Main: Es la clase principal donde solo nos servirá de pivote para poder inicial el programa en la clase **Menu.java**, con la finalidad de mantener un buen orden y gestión del proyecto

2. Paquete menú:

2.1. Menu.java: Es la primera clase a la cual se le hace llamado, donde aquí se encontrará la función principal del programa ya que gestionara las opciones de la aplicación, carga, llamado de métodos que nos ayudaran en el proceso de gestionamiento

3. Paquete obtención

3.1. Asignacion.java: Clase donde se llevará a cabo todas las asignaciones de cursos para alumnos y profesores, junto a la carga de los alumnos por ID de clase.

3.2. Obtenercsv.java: Clase con una de las funciones fundamentales del proyecto el cual es cargar los archivos csv donde serán cargados los registros de alumnos, profesores, cursos, asignación de cursos para alumnos y asignación de alumnos para profesores

4. Paquete Reporte

4.1. Reportes.java: Clase con la finalidad de la creación de todos los reportes asegure las gestiones llevadas a cabo por usuario en un documento .HTML

II. Métodos por clase

1. **Main:** solo se le hace el llamado al método login(); desde el void main para la inicialización del programa
2. **Menú:**
 - 2.1. **Login():** método encargado del ingreso de credenciales usuario y contraseña para acceder al sistema, en caso del ingreso de credenciales invalidas se proporciona la opción de recuperación de contraseña incluida.
 - 2.2. **NuevoUsuario:** método con finalidad de poder agregar nuevos usuarios al registro creado, con un máximo de 10 registros.
 - 2.3. **Ingreso():** Método encargado de la bienvenida el usuario
 - 2.4. **PrimerIngreso():** El método mas cargado e importante de la clase, se despliega el menú con todas las opciones del programa, realizándoles por medio de otros métodos de otras clases, entre las cuales están: GuardarAlumnos(), GuardarProfesores(), GuardarCursos(), AsignarAA(), AsignarAP() y todas los métodos para la creación de reportes
3. **Asignación:**
 - 3.1. **AsignarCursosAlumnos():** método encargado de asignar a los alumnos todas las clases que les corresponden según el resto de cursos que se utilice mediante un arreglo bidimensional.
 - 3.2. **MostrarAsignarCursosAlum():** método que muestra los cursos a los cuales cada uno de los alumnos esta asignado
 - 3.3. **AsignarCursosProfesores():** método encargado de asignar a los profesores todas las clases que les corresponden según el resto de cursos que se utilice mediante un arreglo bidimensional.
 - 3.4. **MostrarAsignarCursosProf():** método que muestra los cursos a los cuales cada uno de los profesores está asignado
 - 3.5. **AsignarCursos():** método encargado de almacenar todos los alumnos con sus distintas clases asignadas con sus respectivas notas en un arreglo tridimensional para su uso
 - 3.6. **MostrarAsignarCursosNotas():** método que muestra los cursos a los cuales cada uno de los cursos junto a los alumnos con sus notas a los cuales cada uno está asignado.
4. **Obtenercsv:**
 - 4.1. **GuardarCursos():** método mediante una ruta relativa o absoluta anteriormente ingresada obtiene los cursos de una archivo .csv con limite de 30 cursos distintos
 - 4.2. **GuardarProfesores():** método mediante una ruta relativa o absoluta anteriormente ingresada obtiene a los profesores de una archivo .csv con límite de 25 profesores distintos
 - 4.3. **GuardarAlumnos():** método mediante una ruta relativa o absoluta anteriormente ingresada obtiene a los alumnos de una archivo .csv con límite de 150 alumnos distintos

- 4.4. **AsignarAA()**: método mediante una ruta relativa o absoluta anteriormente ingresada obtiene las asignaciones correspondientes cada alumno de una archivo .csv con límite de asignación de 3 cursos distintos por alumno
- 4.5. **AsignarAP()**: método mediante una ruta relativa o absoluta anteriormente ingresada obtiene las asignaciones correspondientes cada profesor de una archivo .csv con límite de asignación de 3 cursos distintos por profesor
- 4.6. **GuardarListaAA()**: método encargado de guardar la lista de alumnos para el reporte
- 4.7. **GuardarListaAP()**: método encargado de guardar la lista de profesores para el reporte
- 4.8. **GuardarListaAC()**: método encargado de guardar la lista de cursos para el reporte
- 5. **Reportes:**
 - 5.1. **ReporteStarrrt()**: método con finalidad de inicializar la parte del encabezado del reporte HTML.
 - 5.2. **ReporteMostrar()**: método que crea el archivo HTML y lo rescribe con todas las acciones realizadas durante la sesión en el sistema por usuario.
 - 5.3. **ReporteJuntar()**: método que junta todas las partes del HTML encabezado, cuerpo y parte final.
 - 5.4. **ReporteCrusos()**: método que crea el reporte con digito HTML para la seccion de cursos
 - 5.5. **ReporteProf()**: método que crea el reporte con digito HTML para la seccion de profesores.
 - 5.6. **ReporteAlum()**: método que crea el reporte con digito HTML para la seccion de alumnos.
 - 5.7. **Mayores()**: método encargado de obtener los 5 mejores alumnos de una clase anteriormente indicada según su ID.
 - 5.8. **Menores()**: método encargado de obtener los 5 peores alumnos de una clase anteriormente indicada según su ID.
 - 5.9. **ReporteMenores()**: método que crea el reporte con digito HTML para la seccion delos peores 5 alumnos de una clase.
 - 5.10. **ReporteMayores()**: método que crea el reporte con digito HTML para la seccion delos mejores 5 alumnos de una clase.
 - 5.11. **EjecucionReporteMayores()**: método encargado de tabular el código HTML para el reporte de los mejores 5 alumnos
 - 5.12. **EjecucionReporteMenor()**: método encargado de tabular el código HTML para el reporte de los peores 5 alumnos

III. Glosario

1. **Archivo .csv:** archivo (valores separados por comas) es un archivo de texto que tiene un formato específico que permite guardar los datos en un formato de tabla estructurada.
2. **Archivo .HTML:** archivo de Lenguaje de marcado de hipertexto y es el tipo de archivo de página web
3. **Arreglo:** en java es una estructura de datos que nos permite almacenar un conjunto de datos de un mismo tipo.
4. **Arreglo bidimensional:** Los arreglos bidimensionales son tablas de valores. Cada elemento de un arreglo bidimensional está simultáneamente en una fila y en una columna se maneja en eje x,y
5. **Arreglo tridimensional:** Los arreglos tridimensionales son aquellas que tiene filas, columnas y profundidad se manejan en eje x,y,z
6. **Clase:** Una clase es una plantilla que define la forma de un objeto; en ella se agrupan datos y métodos que operarán sobre esos datos.
7. **Método:** Los métodos son subrutinas que manipulan los datos definidos por la clase y, en muchos casos, brindan acceso a esos datos. En la mayoría de los casos, otras partes de tu programa interactuarán con una clase a través de sus métodos.
8. **Paquete:** Un Paquete en Java es un contenedor de clases que permite agrupar las distintas partes de un programa y que por lo general tiene una funcionalidad y elementos comunes, definiendo la ubicación de dichas clases en un directorio de estructura jerárquica
9. **Ruta absoluta:** Las rutas absolutas señalan la ubicación de un archivo o directorio desde el directorio raíz del sistema de archivos.
10. **Ruta relativa:** Una ruta relativa hace referencia a una ubicación que es relativa a un directorio actual.