	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		


# Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación

---

Universidad Politécnica Salesiana

## Vicerrectorado Docente

Código del Formato:	GUIA-PRL-001
Versión:	VF1.0
Elaborado por:	Directores de Área del Conocimiento Integrantes Consejo Académico
Fecha de elaboración:	2016/04/01
Revisado por:	Consejo Académico
Fecha de revisión:	2016/04/06
Aprobado por:	Lauro Fernando Pesántez Avilés Vicerrector Docente
Fecha de aprobación:	2016/14/06
Nivel de confidencialidad:	Interno

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

## Descripción General

### Propósito


El propósito del presente documento es definir un estándar para elaborar documentación de guías de práctica de laboratorio, talleres o centros de simulación de las Carreras de la Universidad Politécnica Salesiana, con la finalidad de lograr una homogenización en la presentación de la información por parte del personal académico y técnico docente.


### Alcance

El presente estándar será aplicado a toda la documentación referente a informes de prácticas de laboratorio, talleres o centros de simulación de las Carreras de la Universidad Politécnica Salesiana.

### Formatos


- Formato de Guía de Práctica de Laboratorio / Talleres / Centros de Simulación – para Docentes
- Formato de Informe de Práctica de Laboratorio / Talleres / Centros de Simulación – para Estudiantes

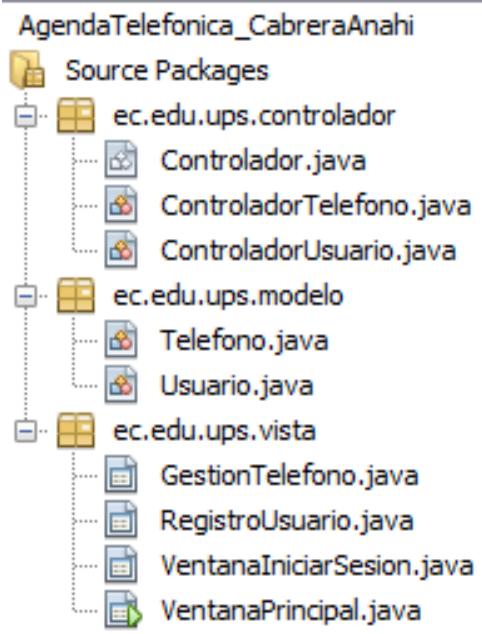
	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		


		<b>FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES</b>	
<b>CARRERA:</b> COMPUTACIÓN		<b>ASIGNATURA:</b> Programación Aplicada	
<b>NRO. PRÁCTICA:</b>	1	<b>TÍTULO PRÁCTICA:</b> Reflexión en Java	
<b>OBJETIVO:</b> Identificar los cambios importantes de Java Diseñar e Implementar las nuevas tecnicas de programación Entender cada una de las características nuevas en Java			
<b>INSTRUCCIONES</b> (Detallar las instrucciones que se dará al estudiante):		1. Revisar los conceptos fundamentales de Java	
		2. Establecer las características de Java en reflexión	
		3. Implementar y diseñar los nuevos componentes de reflexión	
		4. Realizar el informe respectivo según los datos solicitados.	
<b>ACTIVIDADES POR DESARROLLAR</b> (Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)			
1. Revisar la teoría y conceptos de Java 8, 9 ,10, 11, 12, 13, 14, 15			
2. Diseñar e implementar las características de Java para generar la impresión de cualquier lista, de los modelos que tengan el campo id generar automaticamente.			
3. Probar y modificar el metodo validar para que nos permita utilizar excepciones, ademas de modificar el buscar para controlar el nullpointerexception.			
4. Realizar práctica codificando los codigos de las nuevas características de Java y su uso dentro de una agenda telefónica.			
<b>RESULTADO(S) OBTENIDO(S):</b> Realizar procesos de investigación sobre los cambios importantes de Java Entender las aplicaciones de codificación de las nuevas características en base a la programación genérica Entender las funcionalidades adicionales de Java.			
<b>CONCLUSIONES:</b> Aprenden a trabajar en grupo dentro de plazos de tiempo establecidos, manejando el lenguaje de programación de Java.			
<b>RECOMENDACIONES:</b> Realizar el trabajo dentro del tiempo establecido.			

**Docente / Técnico Docente:** Ing. Diego Quisi

	<b>FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA ESTUDIANTES</b>
---	--

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

<b>CARRERA:</b> Computación		<b>ASIGNATURA:</b> Programación Aplicada
<b>NRO. PRÁCTICA:</b>	2	<b>TÍTULO PRÁCTICA:</b> Reflexión en Java
<b>OBJETIVO ALCANZADO:</b> Identificar los cambios importantes de Java Diseñar e Implementar las nuevas técnicas de programación Entender cada una de las características nuevas en Java		
<b>ACTIVIDADES DESARROLLADAS</b>		
1. Revisar la teoría y conceptos de Java 8, 9 ,10, 11, 12, 13, 14, 15		
2. Diseñar e implementar las características de Java para generar la impresión de cualquier lista, de los modelos que tengan el campo id generar automáticamente.		
3. Probar y modificar el método validar para que nos permita utilizar excepciones, además de modificar el buscar para controlar el nullpointerexception.		
4. Realizar práctica codificando los códigos de las nuevas características de Java y su uso dentro de una agenda telefónica.		
<b>Proyecto creado:</b>  <p><b>Paquetes y Clases</b>  <b>ec.edu.ups.modelo</b>  <b>Clase Usuario</b></p> <pre>package ec.edu.ups.modelo;  import java.util.List; import java.util.ArrayList; import java.util.Objects;</pre>		

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```

/**
 *
 * @author Anahi
 */
public class Usuario {
    private int id;
    private String cedula;
    private String nombre;
    private String apellido;
    private String correo;
    private String password;
    private List<Telefono> listaTelefonos;

    public Usuario(int id, String cedula, String nombre, String apellido, String
correo, String password) {
        this.id = id;
        this.cedula = cedula;
        this.nombre = nombre;
        this.apellido = apellido;
        this.correo = correo;
        this.password = password;

        listaTelefonos = new ArrayList<>();
    }


    public Usuario(String cedula, String nombre, String apellido, String correo,
String password,
        List<Telefono> listaTelefonos) {
        this.cedula = cedula;
        this.nombre = nombre;
        this.apellido = apellido;
        this.correo = correo;
        this.password = password;
        this.listaTelefonos = listaTelefonos;
    }

    public Usuario(int id, String cedula, String nombre, String apellido, String
correo, String password, List<Telefono> listaTelefonos) {
        this.id = id;
        this.cedula = cedula;
        this.nombre = nombre;
        this.apellido = apellido;
        this.correo = correo;
        this.password = password;
        this.listaTelefonos = listaTelefonos;
    }

    public Usuario(String apellido) {
        this.apellido = apellido;
    }

    public Usuario() {

```

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```

}

public int getId() {
    return id;
}

public void setId(int id) {
    this.id = id;
}

public String getCedula() {
    return cedula;
}

public void setCedula(String cedula) {
    this.cedula = cedula;
}

public String getNombre() {
    return nombre;
}

public void setNombre(String nombre) {
    this.nombre = nombre;
}

public String getApellido() {
    return apellido;
}

public void setApellido(String apellido) {
    this.apellido = apellido;
}

public String getCorreo() {
    return correo;
}


public void setCorreo(String correo) {
    this.correo = correo;
}

public String getPassword() {
    return password;
}

public void setPassword(String password) {
    this.password = password;
}

public List<Telefono> getListaTelefonos() {
    return listaTelefonos;
}

```

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```

}

public void agregarTelefono(Telefono telefono) {
    this.listaTelefonos.add(telefono);
}

public Telefono buscarTelefono(Telefono telefonoBuscar) {
    return this.listaTelefonos.stream().filter(t ->
t.getNumero().equals(telefonoBuscar.getNumero())).findFirst().get();
}

public void actualizarTelefono(Telefono telefonoNuevo) {
    var telefonoCambiar = listaTelefonos.stream().filter(t -> t.getCodigo() ==
telefonoNuevo.getCodigo()).findFirst().get();
    int index = listaTelefonos.indexOf(telefonoCambiar);
    listaTelefonos.set(index, telefonoNuevo);
}

public void eliminarTelefono(int codigo) {
    var telefonoEliminar = listaTelefonos.stream().filter(telefono ->
telefono.getCodigo() == codigo).findFirst().get();
    listaTelefonos.remove(telefonoEliminar);
}

@Override
public int hashCode() {
    int hash = 3;
    hash = 41 * hash + Objects.hashCode(this.apellido);
    return hash;
}

@Override
public boolean equals(Object obj) {
    if (this == obj) {
        return true;
    }
    if (obj == null) {
        return false;
    }
    if (getClass() != obj.getClass()) {
        return false;
    }
    final Usuario other = (Usuario) obj;
    if (!Objects.equals(this.apellido, other.apellido)) {
        return false;
    }
    return true;
}

@Override
public String toString() {
    return "Usuario\n" + "cedula=" + cedula + ", nombre=" + nombre
        + ", apellido=" + apellido + ", correo=" + correo + ", contraseña="

```

```
        + password + ",\nlistaTelefonos=" + listaTelefonos;\n    }\n}
```

### Clase Teléfono

```
package ec.edu.ups.modelo;\n\nimport java.util.Objects;\n\n/**\n * \n * @author Anahi\n */\npublic class Telefono {\n    private int codigo;\n    private String tipo;\n    private String numero;\n    private String operadora;\n\n    public Telefono(int codigo, String tipo, String numero, String operadora) {\n        this.codigo = codigo;\n        this.tipo = tipo;\n        this.numero = numero;\n        this.operadora = operadora;\n    }\n\n    public Telefono() {\n\n    }\n\n    public Telefono(String numero) {\n        this.numero = numero;\n    }\n\n    public int getCodigo() {\n        return codigo;\n    }\n\n    public void setCodigo(int codigo) {\n        this.codigo = codigo;\n    }\n\n    public String getTipo() {\n        return tipo;\n    }\n}
```



```
public void setTipo(String tipo) {
    this.tipo = tipo;
}

public String getNumero() {
    return numero;
}

public void setNumero(String numero) {
    this.numero = numero;
}

public String getOperadora() {
    return operadora;
}


public void setOperadora(String operadora) {
    this.operadora = operadora;
}

@Override
public boolean equals(Object obj) {
    if (this == obj) {
        return true;
    }
    if (obj == null) {
        return false;
    }
    if (getClass() != obj.getClass()) {
        return false;
    }
    final Telefono other = (Telefono) obj;
    if (!Objects.equals(this.numero, other.numero)) {
        return false;
    }
    return true;
}
}
```

ec.edu.ups.controlador

Controlador

package ec.edu.ups.controlador;

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```

import java.util.List;
import java.util.ArrayList;

/**
 *
 * @author Anahi
 */
public abstract class Controlador<T> {

    private List<T> listaGenerica;

    public Controlador() {
        listaGenerica = new ArrayList<>();
    }

    public boolean create(T objetoCrear) {
        if (!listaGenerica.contains(objetoCrear)) {
            return listaGenerica.add(objetoCrear);
        }
        return false;
    }

    public T read(T objetoBuscar) {
        if (listaGenerica.contains(objetoBuscar)) {
            return (T) listaGenerica.stream().filter(objeto ->
objeto.equals(objetoBuscar)).findFirst().get();
        } else
            return null;
    }

    public T update(T objetoAntiguo, T objetoNuevo) {
        int index = listaGenerica.indexOf(objetoAntiguo);
        return listaGenerica.set(index, objetoNuevo);
    }


    public boolean delete(T objetoEliminar) {
        return listaGenerica.remove(objetoEliminar);
    }

    public List<T> findAll() {
        return listaGenerica;
    }

    public abstract int cargarCodigo();
}

```

**ControladorUsuario**

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```

package ec.edu.ups.controlador;

import ec.edu.ups.modelo.Telefono;
import ec.edu.ups.modelo.Usuario;
import java.util.List;

/**
 *
 * @author Anahi
 */
public class ControladorUsuario extends Controlador {

    public ControladorUsuario() {

    }

    @Override
    public int cargarCodigo() {
        if (findAll().size() > 0) {
            return findAll().size() + 1;
        } else {
            return 1;
        }
    }

    public Usuario comprobarMayusculas(Usuario usuario) {
        String[] nombres = usuario.getNombre().split(" ");
        String[] apellidos = usuario.getApellido().split(" ");

        String nombreMayusculas = "";
        for (String nombre : nombres) {
            nombreMayusculas +=
String.valueOf(Character.toUpperCase(nombre.charAt(0)));
            nombreMayusculas += nombre.substring(1, nombre.length());
            nombreMayusculas += " ";
        }


        String apellidoMayusculas = "";
        for (String apellido : apellidos) {
            apellidoMayusculas +=
String.valueOf(Character.toUpperCase(apellido.charAt(0)));
            apellidoMayusculas += apellido.substring(1, apellido.length());
            apellidoMayusculas += " ";
        }

        usuario.setNombre(nombreMayusculas.trim());
        usuario.setApellido(apellidoMayusculas.trim());

        return usuario;
    }

    public Usuario iniciarSesion(String correo, String password) {
        for (var usuario : (List<Usuario>) findAll()) {

```

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```

        if (usuario.getCorreo().equals(correo) &&
usuario.getPassword().equals(password)) {
            return usuario;

        }

    }
    return null;
}

public Usuario readCedula(Usuario usuarioBuscar) {
    var copiaListaUsuarios = (List<Usuario>) List.copyOf(findAll());
    return copiaListaUsuarios.stream().filter(usuario ->
usuario.getCedula().equals(usuarioBuscar.getCedula())).findFirst().get();
}

public Usuario readNumero(Telefono telefonoBuscar) {
    for (var usuario : (List<Usuario>) findAll()) {
        if (usuario.buscarTelefono(telefonoBuscar) != null) {
            return usuario;
        }
    }
    return null;
}
}

```

### ControladorTelefono

```

package ec.edu.ups.controlador;


import ec.edu.ups.modelo.Telefono;
import java.util.List;

/**
 *
 * @author Anahi
 */
public class ControladorTelefono extends Controlador {

    public ControladorTelefono() {
    }

    @Override
    public int cargarCodigo() {
        var copiaListaTelefonos = (List<Telefono>) List.copyOf(findAll());
        if (copiaListaTelefonos.isEmpty()) {
            return 1;
        } else {
            return copiaListaTelefonos.get(copiaListaTelefonos.size() -
1).getCodigo() + 1;
        }
    }
}

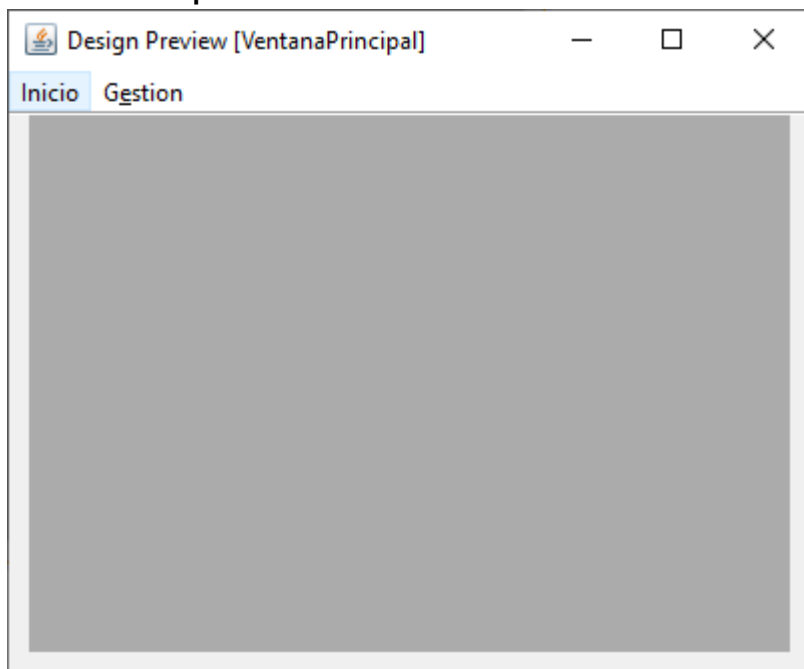
```

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

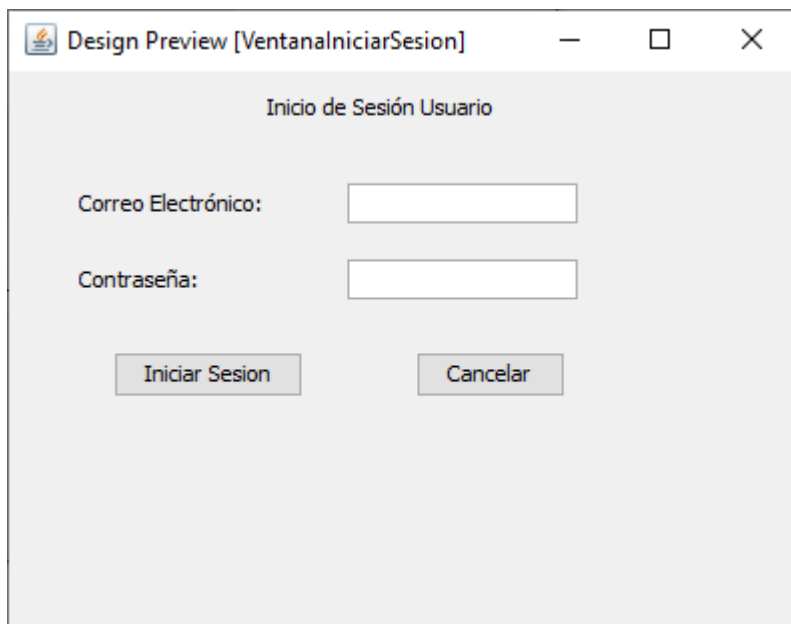
}

}

## ec.edu.ups.vista Ventana Principal



## Ventana Iniciar Sesión



## Ventana Registro Usuario

Design Preview [RegistroUsuario]

Registro Usuarios

Nombre:

Apellido:

Cedula:

Correo Electrónico:

Contraseña:

Registrar Cancelar

### Ventana Gestión Teléfonos

Design Preview [GestionTelefono]

Gestión Teléfonos

Código:

Tipo:

Número:

Operadora:

Crear Editar Eliminar Cancelar


Código	Tipo	Número	Operadora

### RESULTADO(S) OBTENIDO(S):

Realizar procesos de investigación sobre los cambios importantes de Java

Entender las aplicaciones de codificación de las nuevas características en base a la programación genérica

Entender las funcionalidades adicionales de Java.

	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

**CONCLUSIONES:**

Con la reflexión en java podemos observar que el código y la interacción con el usuario mejoran en el uso de la aplicación creada.

**RECOMENDACIONES:**

Utilizar buenas prácticas de programación.

**Nombre de estudiante:** Edith Anahí Cabrera Bermeo