	<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> Examen Practico Java	
<b>OBJETIVO:</b> Identificar los cambios importantes de Java Diseñar e Implementar expresiones regulares Entender la cada uno 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 programación genérica	
		3. Implementar y diseñar los nuevos componentes de programación genérica	
		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			
2. Diseñar e implementar las características de Java para generar una expresion regular.			
3. Probar su funcionamiento y rendimiento dentro de los equipos de cómputo de programación genérica.			
4. Realizar práctica codificando los codigos de las nuevas características de Java y su uso dentro de un sistema escolar.			
<b>Enunciado</b> Se desea generar un sistema que me permita extraer infomación del internet a traves de expresiones regulares, esta informacion permitira vincular actividades desarrolladas del los niños con aplicaciones mobiles que permitan apoyar en el desarrollo de las actividades planteadas ( <a href="https://play.google.com/store?hl=es&amp;gl=US">https://play.google.com/store?hl=es&amp;gl=US</a> ).  Adicionalmente, se debe realizar un sistema de gestion de alumnos y actividades planificadas por curso, dentro de este sistema se debe realizar un procesos de administracion de usuarios los mismo que son los docentes de cada curso escolar, en este sentido solo debemos tener un administrador (Rector) el encargado de crear docentes y el curso que se le asigna. Ejemplo Rector: Docentes: 1. Diego Quisi 2. Vladimir Robles 3. Etc.  Cursos: 1 de basica 2 de basica 3 basica			

	<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		

Asignacion de Curso – Docente

1 Basica -> Diego Quisi

2 Basica -> Vladimir Robles

Dentro de cada curso el docente gestionara los estudiantes y las actividades planificadas para el curso, estas actividades tendra una opcion de buscar aplicaciones moviles dentro de la tiendas de play store, obtenidas desde el internet, dentro de esta información lo importante es mostrar el link y una descripción para ello deberán utilizar expresiones regulares.

Ejemplo Docentes:

Alumnos

1. Juan Perez

2. Maria Peralta

3. .

Actividades:

1. Suma de numeros -> Obtener aplicaciones moviles (Link y Titulo)
2. Resta de numeros -> Obtener aplicaciones moviles
3. Oraciones compuestas -> Obtener aplicaciones moviles
4. Etc.

Toda esta infomación sera almacenada dentro de archivos y deberan tener aplicado al menos una patron de diseño y las nuevas características de programación de Java 8 o superior.

Al finalizar, generar el informe de la practica en formato PDF y subir todo el proyecto incluido el informe al repositorio personal.

**La fecha de entrega: 23:55 del 01 de diciembre del 2020.**

#### **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 y expresiones regulares.

Entender las funcionalidades adicionales de Java.

#### **CONCLUSIONES:**


Aprenden a trabajar en grupo dentro de plazos de tiempo establecidos, manejando el lenguaje de programación de Java.


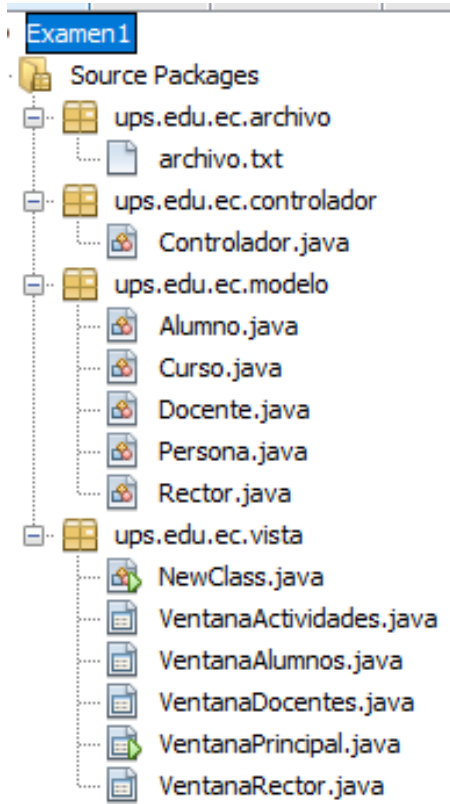
#### **RECOMENDACIONES:**

Realizar el trabajo dentro del tiempo establecido.

**Docente / Técnico Docente:** \_\_\_\_\_

**Firma:** \_\_\_\_\_

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

		<b>FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA ESTUDIANTES</b>	
<b>CARRERA:</b>		<b>ASIGNATURA:</b>	
<b>NRO. PRÁCTICA:</b>		<b>TÍTULO PRÁCTICA:</b>	
<b>OBJETIVO ALCANZADO:</b>			
<b>ACTIVIDADES DESARROLLADAS</b>			
<p>1.</p>  <p style="text-align: center;"><b>Creamos los paquetes con sus respectivas clases</b></p>			
2.			

```

public class Controlador {

    private Pattern patron;
    private Matcher corpus;

    public void ingresarRegex(String regex) {
        patron = Pattern.compile(regex);
    }

    public boolean validar(String texto) {
        corpus = patron.matcher(texto);
        return corpus.find();
    }

    public Set<String> obtenerURL(String paginaTexto) {
        Set<String> resultado = new HashSet();
        corpus = patron.matcher(paginaTexto);
        while (corpus.find()) {
            resultado.add(corpus.group(0));
        }
        return resultado;
    }

    public Pattern getPatron() {
        return patron;
    }

    public void setPatron(Pattern patron) {
        this.patron = patron;
    }
}

```

La clase controlador se encargara de controlar la expresión regular

3.

```

Controlador controlador = new Controlador();

String regex = "<a\\shref=\"\\\"/(.^[^\\\"\\\"\\s])+\"";
controlador.ingresarRegex(regex);

StringBuilder stringBuilder = new StringBuilder();
String textoBusqueda = "Lengua";
try {
    URL urlObject = new URL("https://play.google.com/store/search?q=" + textoBusqueda.replaceAll("\\s", "\\+"));
    URLConnection urlConnection = urlObject.openConnection();
    urlConnection.setRequestProperty("User-Agent", "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.11 (E
    BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(urlConnection.getInputStream()),
    String inputLine;
    while ((inputLine = bufferedReader.readLine()) != null) {
        stringBuilder.append(inputLine);
    }
}

```

Utilizando el método “ingresarRegex” de la clase controlador ingresaremos nuestra expresión regular la cual nos ayudara a buscar los resultados que pidamos en la pagina sugerida

4.

```
public class Persona {  
  
    private String nombre;  
    private String apellido;  
  
    public Persona() {  
    }  
  
    public Persona(String nombre, String apellido) {  
        this.nombre = nombre;  
        this.apellido = apellido;  
    }  
  
    public String getNombre() {  
        return nombre;  
    }  
  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
  
    public String getApellido() {  
        return apellido;  
    }  
  
    public void setApellido(String apellido) {
```


**Creamos una clase padre para las personas**

5.

```
public class Rector<T> extends Persona {  
  
    private List<T> docentes;  
  
    public Rector() {  
        docentes = new ArrayList<>();  
    }  
  
    public Rector(String nombre, String apellido) {  
        super(nombre, apellido);  
        docentes = new ArrayList<>();  
    }  
  
    public List<T> getDocentes() {  
        return docentes;  
    }  
  
    public void setDocentes(List<T> docentes) {  
        this.docentes = docentes;  
    }  
  
    public void agregarDocente(T objeto) {  
        docentes.add(objeto);  
    }  
}
```

**Podemos usar herencia tanto en el rector, docentes y alumnos.**

**Adicionalmente agregamos los métodos extra que necesita cada clase anteriormente mencionada**

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

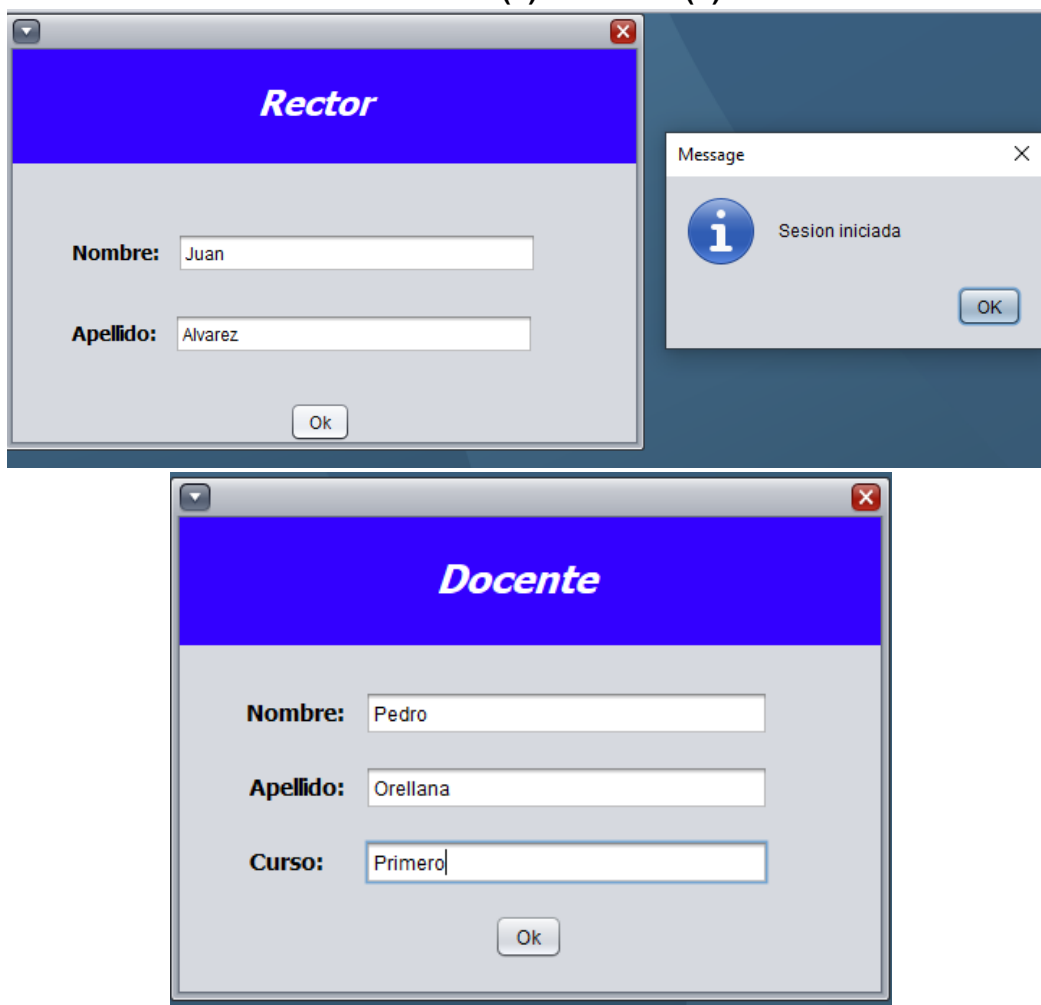
Como por ejemplo el agregar docentes que usamos en el rector y que al igual usamos programación generica

6.

```
// TODO add your handling code here:
try {
    String ruta = "archivo.txt";
    File file = new File(ruta);
    if (!file.exists()) {
        file.createNewFile();
    }
    FileWriter fw = new FileWriter(file);
    BufferedWriter bw = new BufferedWriter(fw);
    bw.write("Rector" + x.toString() + "\n");
    x.getDocentes().stream().forEach(n -> {
        try {
            bw.write(n + "\n");
        } catch (IOException ex) {
            Logger.getLogger(VentanaPrincipal.class.getName()).log(Level.SEVERE, null, ex);
        }
    });
    bw.close();
} catch (Exception e) {
    e.printStackTrace();
}
```

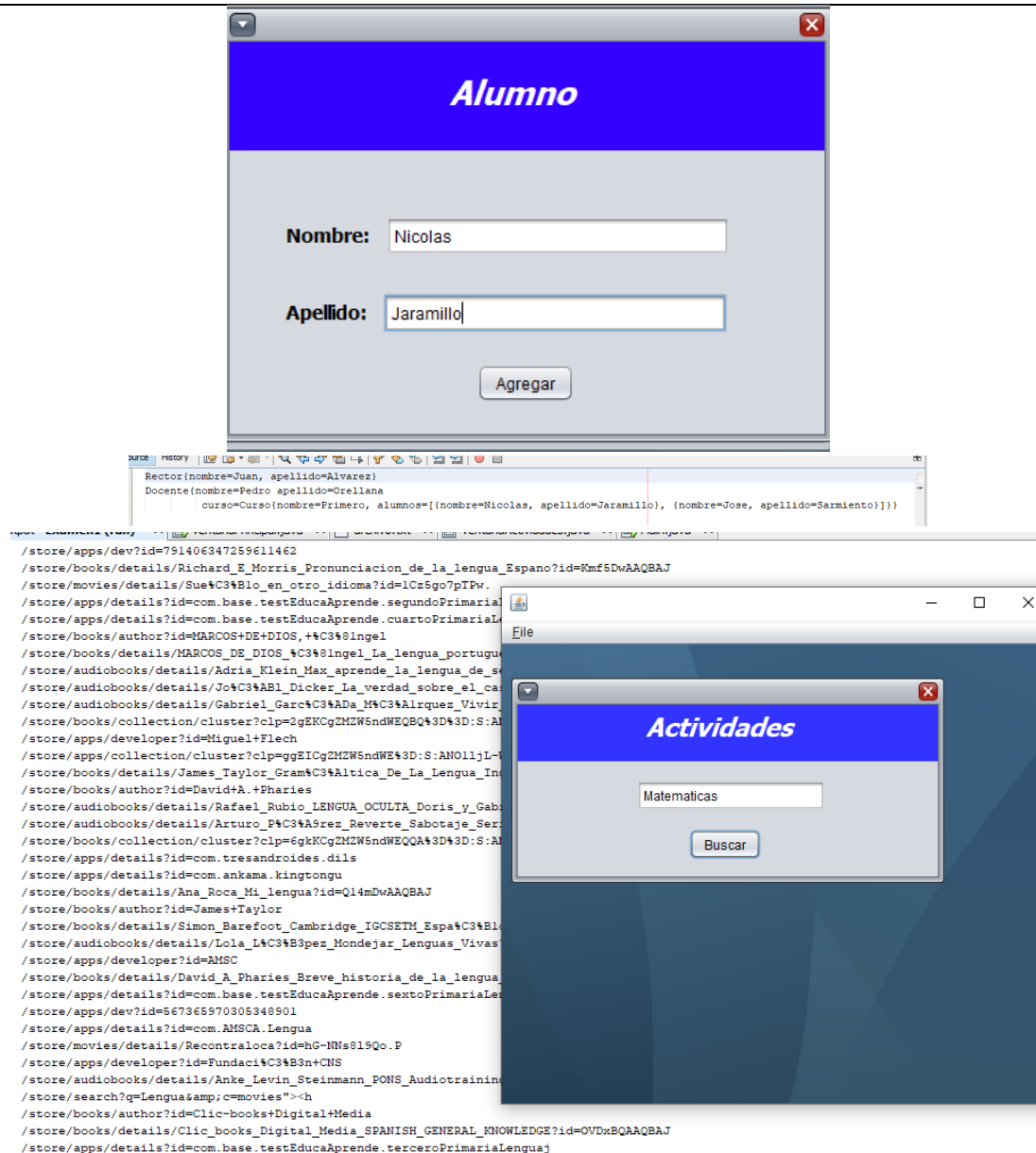
N.

RESULTADO(S) OBTENIDO(S):



The screenshot displays three windows from a Java application. The top-left window is titled 'Rector' and contains two text input fields: 'Nombre:' with the value 'Juan' and 'Apellido:' with the value 'Alvarez'. An 'Ok' button is at the bottom. To its right is a 'Message' dialog box with an information icon and the text 'Sesion iniciada', with an 'OK' button. Below these is a third window titled 'Docente' with three text input fields: 'Nombre:' with 'Pedro', 'Apellido:' with 'Orellana', and 'Curso:' with 'Primero'. An 'Ok' button is at the bottom of this window.






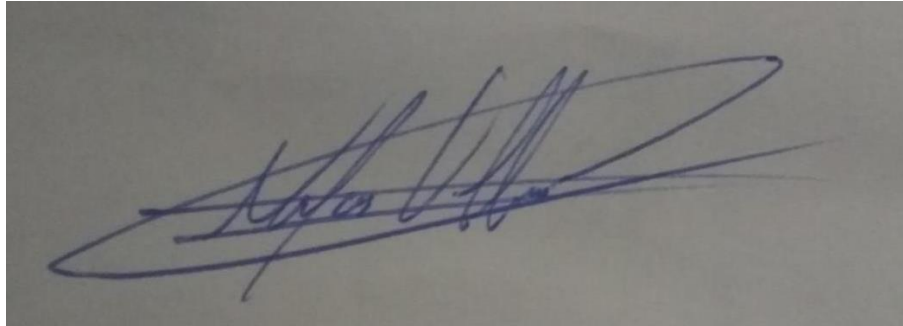
The screenshot displays a web application interface. At the top, there is a blue header with the word "Alumno" in white. Below this, there is a form with two input fields: "Nombre:" with the value "Nicolas" and "Apellido:" with the value "Jaramillo". A button labeled "Agregar" is positioned below the form. Below the form, there is a code editor showing a JSON array of student data: `curso=Curso(nombre=Primero, alumnos=[(nombre=Nicolas, apellido=Jaramillo), (nombre=Jose, apellido=Sarmiento)])`. To the right of the code editor, there is a window titled "Actividades" with a blue header. Inside this window, there is a text input field containing "Matematicas" and a button labeled "Buscar".

**CONCLUSIONES:**

**RECOMENDACIONES:**

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

**Nombre de estudiante:** Nicolas Orellana



**Firma de estudiante:**