

Análisis

El ejercicio consiste en seleccionar un conjunto de frutas de manera tal que al ser expulsadas de una estructura de tipo pila queden en el orden que se haya mostrado previamente.

Flujo del programa:

- **Primero:** se muestra al usuario una secuencia de imágenes de 6 frutas diferentes, y más abajo aparecen las 6 frutas de manera distribuida sin ningún orden específico.
- **Segundo:** El usuario selecciona las frutas, que se van apilando mas abajo.
- **Tercero:** El usuario presiona el botón verificar el cual desapiladora las frutas anteriormente ordenadas por el usuario.

Si después de desapilar loas frutas, estas quedan en el mismo orden que la secuencia original, el usuario gana el juego, de lo contrario pierde.

Casos de uso:

historia de usuario #	1	título	apilar frutas
Descripción	Como	usuario	
	quiero	poder seleccionar una fruta	
	para	intentar ordenar las frutas según la secuencia inicial	
Criterios de aceptación			

historia de usuario #	2	título	apilar frutas
Descripción	Como	usuario	
	quiero	distinguir las frutas elegidas	
	para	saber cuáles elementos ya fueron seleccionados	

Criterios de	
---------------------	--

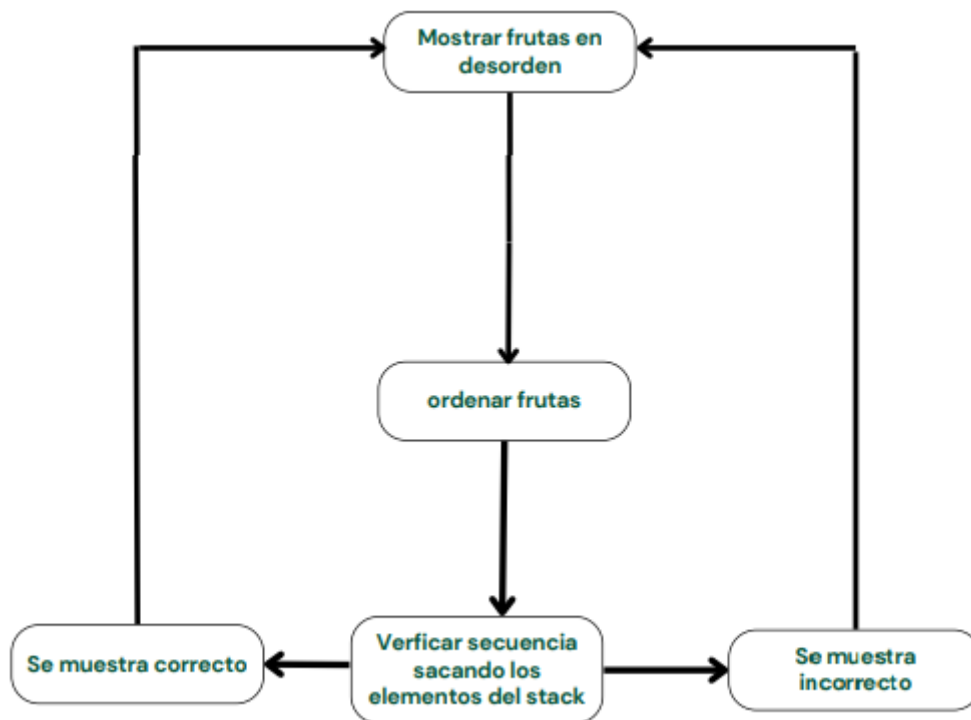
aceptación	
-------------------	--

historia de usuario #	3	título	apilar frutas
Descripción	Como	usuario	
	quiero	verificar el orden que entregué	
	para	intentar ordenar las frutas según la secuencia inicial	
Criterios de aceptación			

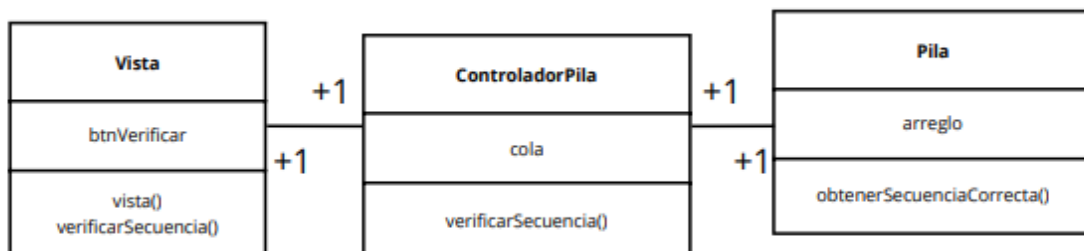
historia de usuario #	4	título	apilar frutas
Descripción	Como	usuario	
	quiero	poder reiniciar el programa	
	para	intentar de nuevo lo que se hay fallado	
Criterios de aceptación			

Diseño

- Diagrama de flujo:



- Diagrama de clases:



Desarrollo:

```
private void initMiddlePanel() {
    middlePanel = new JPanel(null);
    middlePanel.setPreferredSize(new Dimension(600, 350));
    middlePanel.setBackground(new Color(240, 255, 240));

    // Lista de frutas
    List<String> fruitsShuffled = new ArrayList<>(Arrays.asList(
        "manzana.png", "fresa.png", "uva.png",
        "naranja.png", "piña.png", "arandano.png"
    ));
    Collections.shuffle(fruitsShuffled); // desordena las frutas

    // 6 posiciones bien distribuidas
    Point[] posiciones = {
        new Point(50, 30),
        new Point(200, 30),
        new Point(350, 30),
        new Point(100, 150),
        new Point(250, 150),
        new Point(400, 150)
    };
    List<Point> posicionesDisponibles = new ArrayList<>(Arrays.asList(posiciones))
    Collections.shuffle(posicionesDisponibles); // para no tener patrones fijos

    for (int i = 0; i < fruitsShuffled.size(); i++) {
        String fileName = fruitsShuffled.get(i);
        Point p = posicionesDisponibles.get(i);

        URL url = getClass().getResource("/images/" + fileName);
        if (url == null) continue;
    }
}
```