

Julian David Camacho
Simon Guzman Anaya
Camilo Bonilla Belalcázar
David Mesa Martinez

Ingenieria de Software II

1. Public class Paragraph {

```
protected List<String> myLines;  
public Paragraph() { myLines = new ArrayList<>(); }  
public List<String> textAlignment() {  
    return myLines;  
}
```

```
public String getLine(int position) throws Exception {
```

```
    if (position >= 0 && position < myLines.size())  
        return myLines.get(position);  
    else throw new exception("out of range");  
}
```

```
public int getSize() { return myLines.size(); }
```

```
public void addLine(String line) { myLines.add(line); }
```

• Diagrama de clases

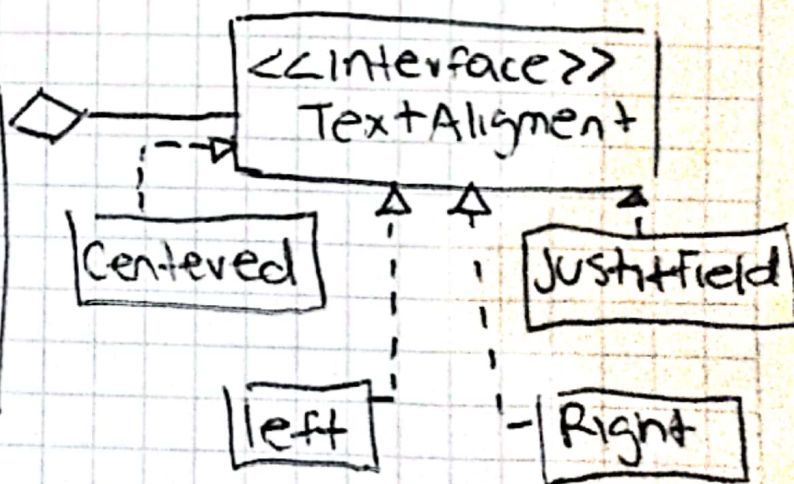
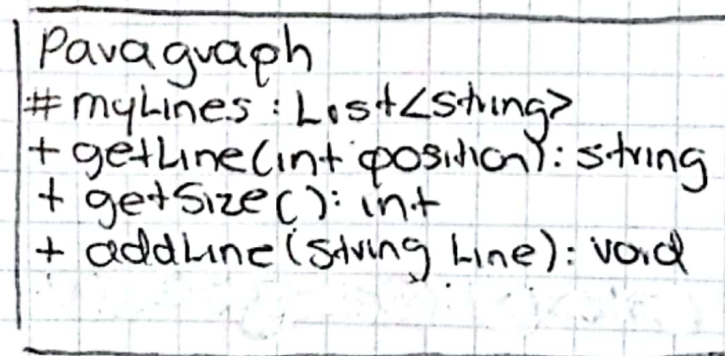
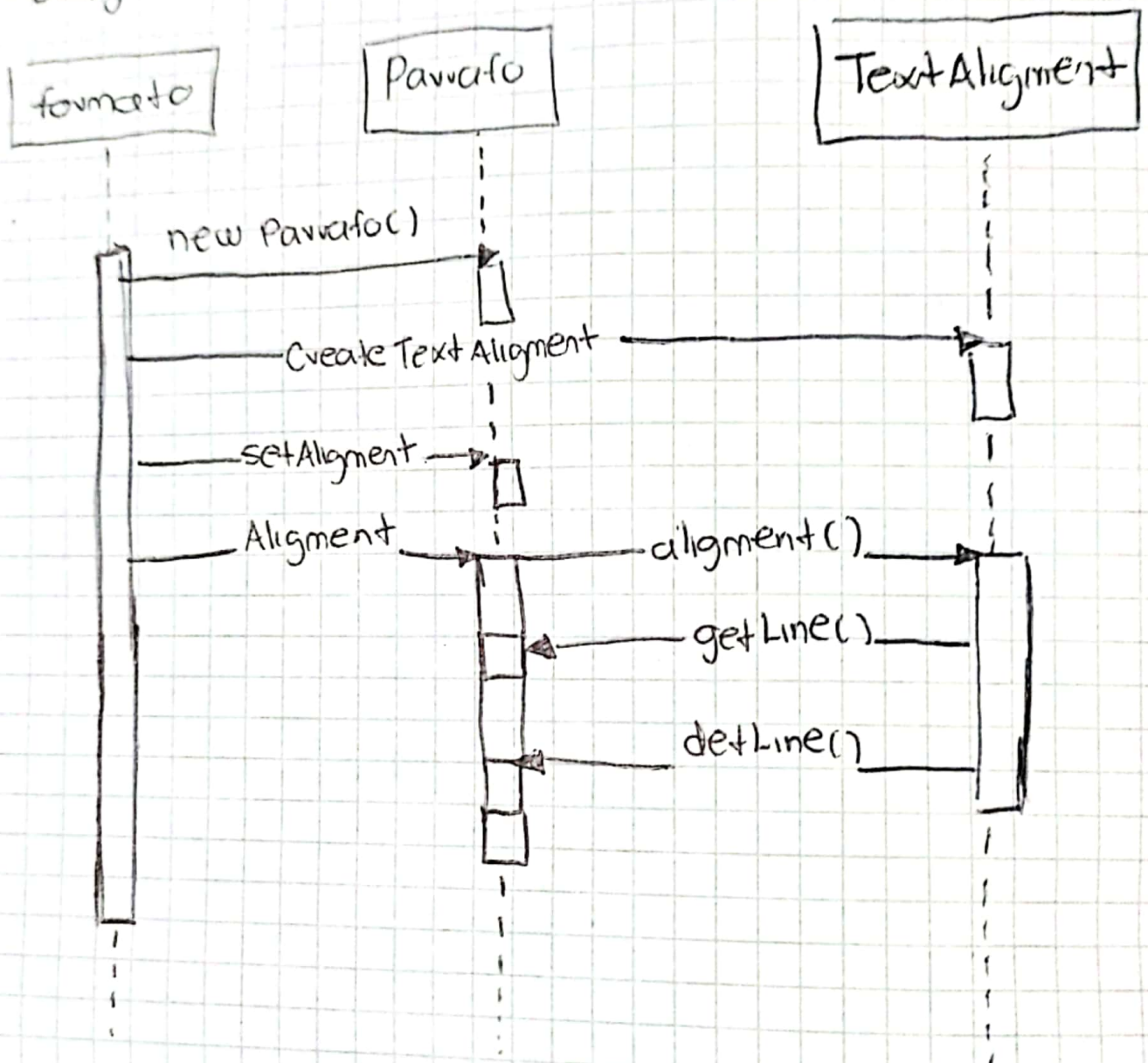
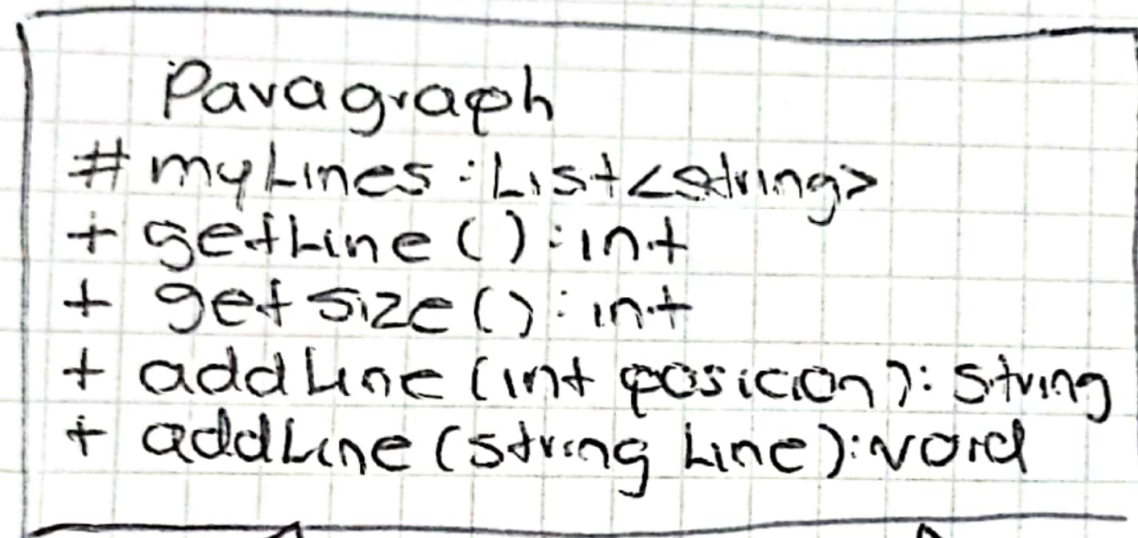


Diagrama de Secuencia



2. la clase sobre escrita es subconjunto de la clase padre o sea es covariante, pero la clase padre acepta como resultado 0, pero la sobre-escrita no, aqui podemos observar como se viola el principio de sustitucion de liskov

2. UML



Extends

Extends

