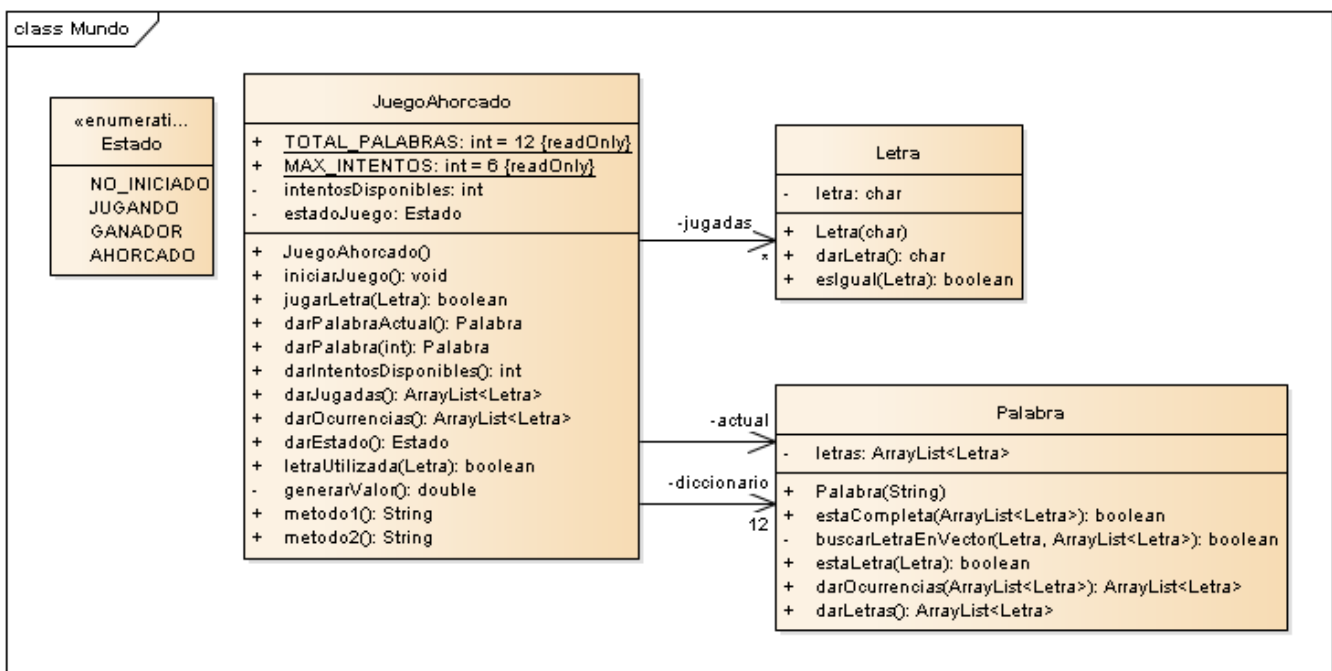


## Ahorcado

### Hoja de trabajo

Nombre:  
Código:

Considere el ejercicio del Ahorcado, cuyo modelo conceptual se muestra a continuación (puede utilizar cualquiera de los elementos definidos en dicho modelo para completar las actividades solicitadas en este taller):



1. Complete el siguiente método que retorna la palabra con la mayor cantidad de caracteres en el diccionario. En caso de que haya dos palabras con el mismo mayor número de caracteres se retorna la primera.

```
public class JuegoAhorcado
{
    //-----
    // Métodos
    //-----

    public Palabra darPalabraMasLarga( )
    {

    }
}
```

2. Complete el siguiente método que retorna si hay una palabra en el diccionario con una cantidad de caracteres mayor o igual al valor recibido como parámetro.

```
public class JuegoAhorcado
{
    //-----
    // Métodos
    //-----

    public boolean hayPalabraConNumeroCaracteres( int pNumeroCaracteres )
    {

    }
}
```

3. Complete el siguiente método en la clase Palabra, el cual cuenta el número de veces que aparece una letra en la palabra. Aplique el patrón de recorrido total sobre las letras de la palabra. El método recibe como parámetro una variable de tipo char con la letra que se está buscando.

```
public class Palabra
{
    //-----
    // Métodos
    //-----

    public int contarVecesLetra( Char pLetra )
    {

    }
}
```

4. Complete el siguiente método en la clase JuegoAhorcado, el cual busca y retorna la primera palabra del diccionario con más de 4 vocales. Si no hay ninguna palabra con más de cuatro vocales el método debe retornar null.

```
public class JuegoAhorcado
{
    //-----
    // Métodos
    //-----

    public Palabra palabraConMasDeCuatroVocales( )
    {

    }
}
```

5. Complete el siguiente método en la clase Palabra. Dicho método recibe una palabra como parámetro y cuenta cuántas letras tienen las dos palabras (la que ejecuta el método y la recibida como parámetro) en común.

```
public class Palabra
{
    //-----
    // Métodos
    //-----

    public int darCantidadLetrasIguales( )
    {

    }
}
```