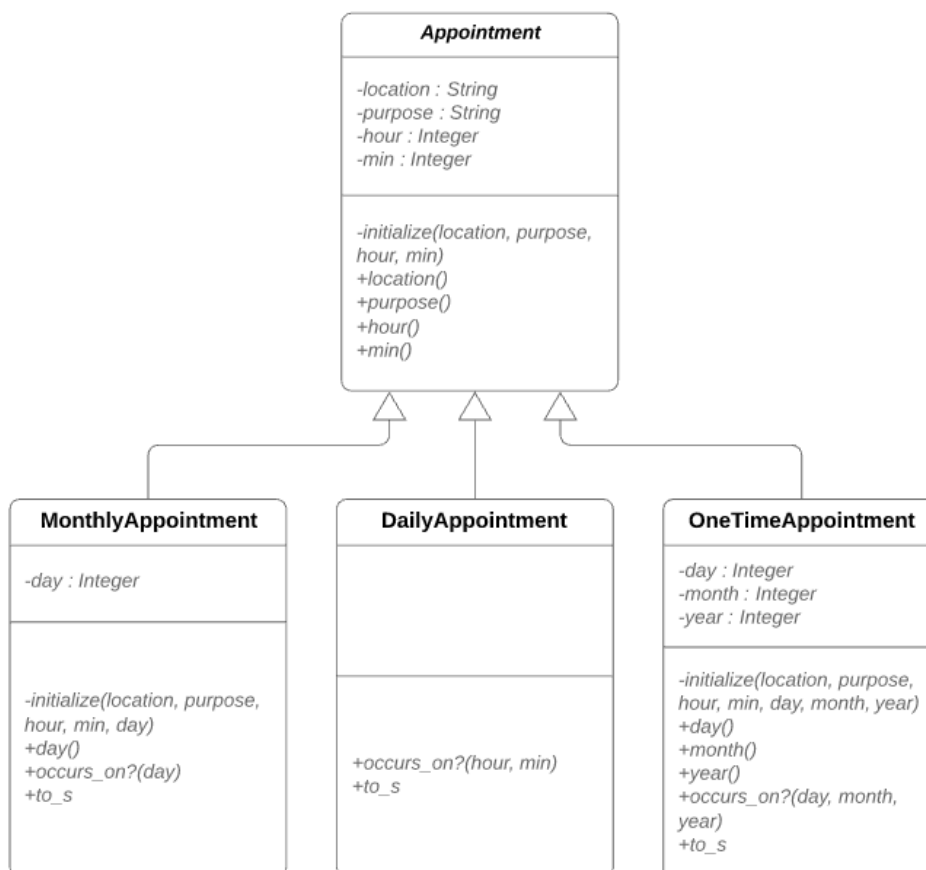


## Desafío - Herencia y UML

- Para realizar este desafío debes haber estudiado previamente todo el material disponible correspondiente a la unidad.
- Una vez terminado el desafío, comprime la carpeta que contiene el desarrollo de los requerimientos solicitados y sube el .zip en el LMS.
- Puntaje total: **10 puntos**.
- Desarrollo desafío:
  - El desafío se debe desarrollar de manera Individuall.

### Requerimientos

Se requiere implementar, aplicando los principios aprendidos en esta unidad, el siguiente diagrama UML en Ruby:





## Hints:

El método `occurs_on?` debe responder un `true` o `false` según corresponda recibiendo los argumentos correspondientes descritos.

¿Sabías que cuando utilizamos la instrucción `puts` sobre un `object` estamos llamando implícitamente al método `to_s`?

Este método se encuentra definido en la clase `Object` y lo podemos sobrescribir para modificar su comportamiento :)

- Cada clase tiene un puntaje de **2 puntos**.

El método `to_s` debe responder de forma distinta para cada objeto (polimorfismo):

```
puts OneTimeAppointment.new('Desafío Latam', 'Trabajo', 14, 30, 4, 6, 2019)
```

```
# Reunión única en Desafío Latam sobre Trabajo el 4/6/2019 a la(s) 14:30.
```

```
puts DailyAppointment.new('Desafío Latam', 'Educación', 8, 15)
```

```
# Reunión diaria en Desafío Latam sobre Educación a la(s) 8:15.
```

```
puts MonthlyAppointment.new('NASA', 'Aliens', 8, 15, 23)
```

```
# Reunión mensual en NASA sobre Aliens el día 23 a la(s) 8:15.
```