

DOCUMENTO DE CAMBIOS

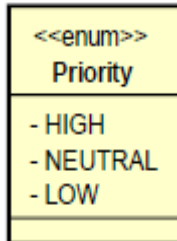
GRUPO 24

Contenido

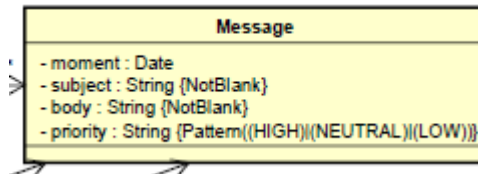
1. Cambios en UML	3
2. Cambios en JAVA.....	7

1. Cambios en UML

Antes:

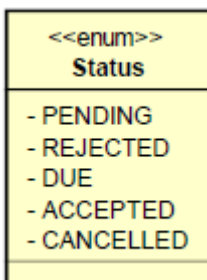


Después:

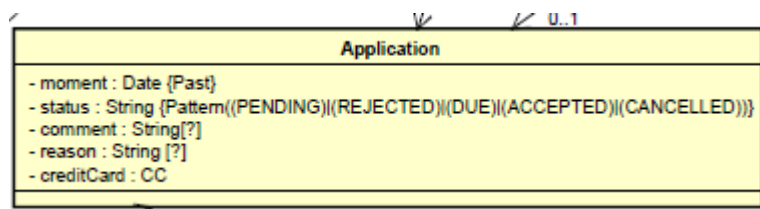


- Priority pasamos a tenerlo definido como un enum a introducirlo dentro de la clase Message como un atributo String.

Antes:

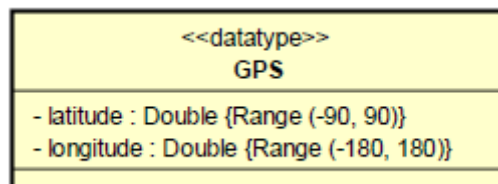


Después:

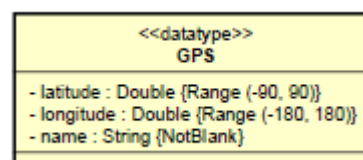


- El enum Status lo eliminamos y dentro de application introducimos un nuevo atributo Status de tipo String.

Antes:



Después:

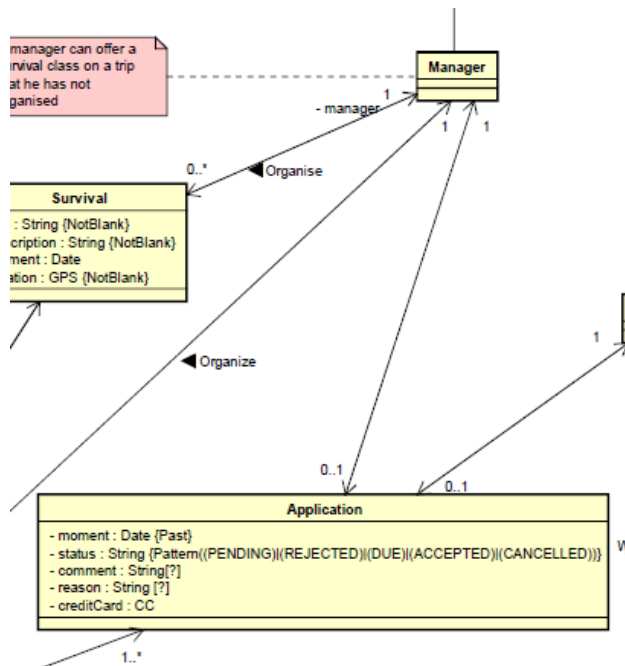


- Añadimos un nuevo atributo a GPS de tipo String llamado name.

Antes:

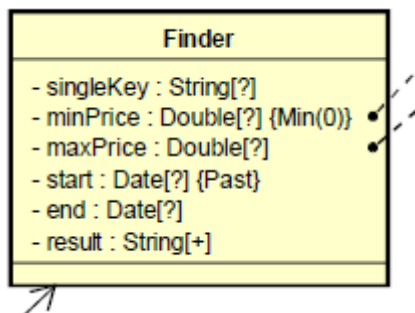
Sin asociación entre manager y Application.

Después:

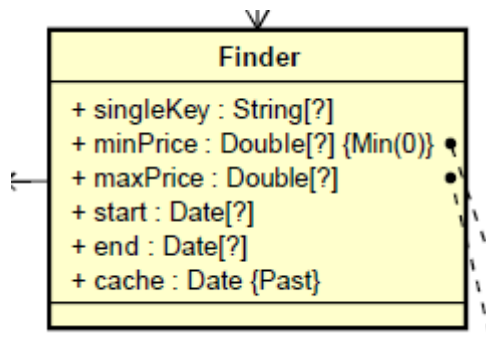


- Añadimos una asociación bidireccional entre Manager y Application

Antes:

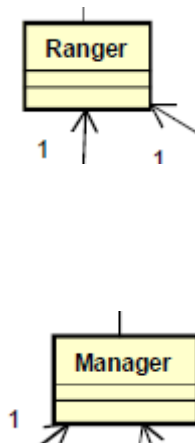


Después:

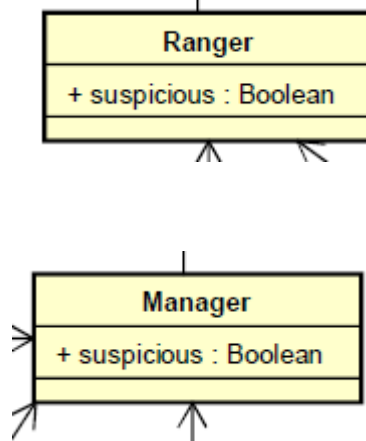


- Añadimos una relación más a Finder con Trip, de modo que queda conectado con Trip y Explorer.
- En Finder eliminamos el atributo result y añadimos chache de tipo date que sirve para mantener en caché durante 24 horas las búsquedas realizadas.

Antes:



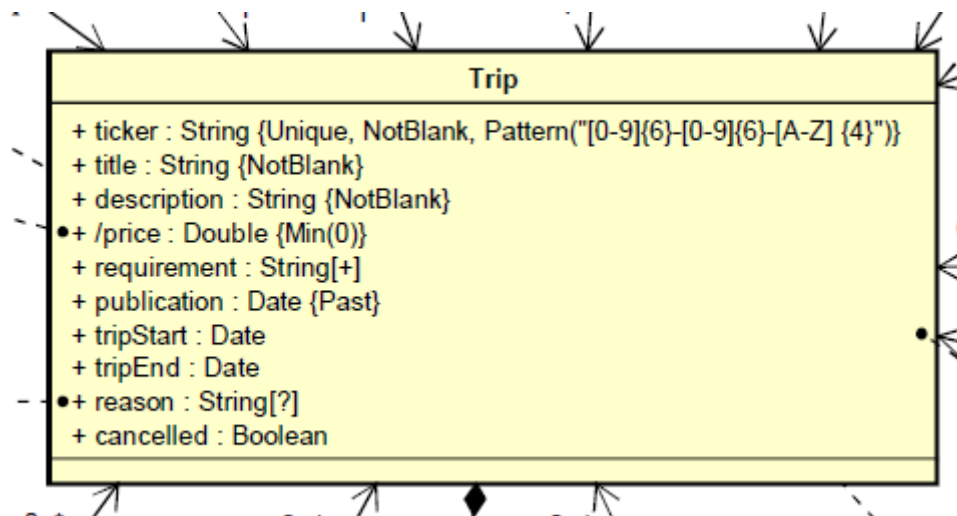
Después:



- Añadimos en Ranger y Manager un nuevo atributo llamado suspicious para sabes si estos son sospechosos.

Antes: Sin atributo cancelled

Después:



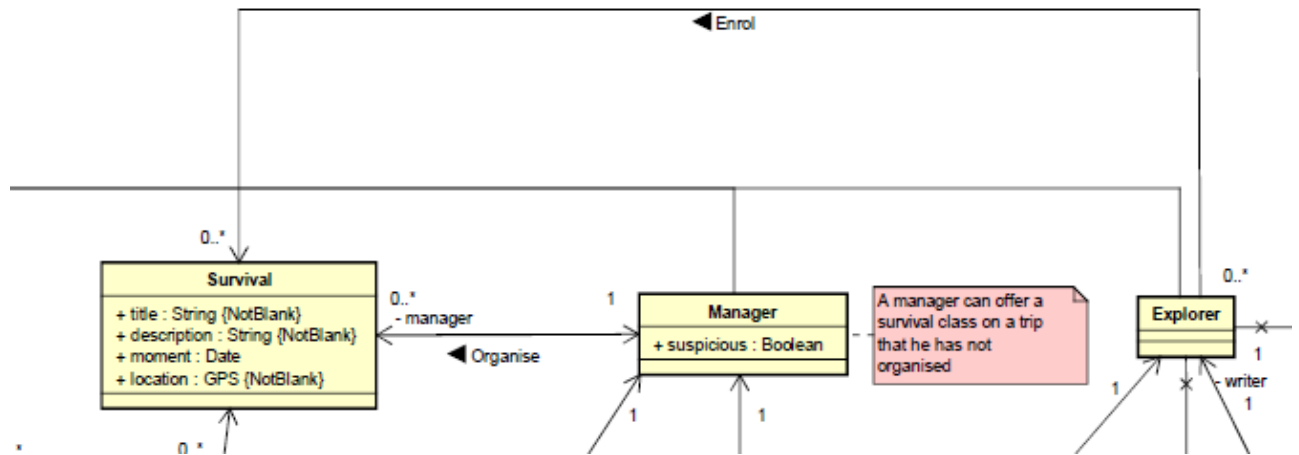
- Añadimos en Trip un nuevo atributo llamado cancelled (boolean) para saber si ha sido cancelado el Trip o no.

Antes: Todos los atributos de las clases empezaban con un guion (indica que son privados).

Después: Ponemos el comienzo de los atributos con + (indica que son públicos).

Antes: sin relación entre Explorer y Survival.

Después:



- Añadimos una nueva relación entre Explorer y Survival con direccionalidad hacia Survival. Esto nos sirve para una más fácil implementación de algunas queries.

2. Cambios en JAVA

Antes: Enum Priority

Después:

```
@Valid
@NotNull
@Pattern(regexp = "^((HIGH)|(NEUTRAL)|(LOW))$")
public String getPriority() {
    return priority;
}

public void setPriority(String priority) {
    this.priority = priority;
}
```

- Eliminamos el Enum priority y añadimos un nuevo atributo priority en la clase Message con sus respectivos atributos.

Antes: Enum status

Después:

```
@NotNull
@Pattern(regexp = "^((PENDING)|(REJECTED)|(DUE)|(ACCEPTED)|(CANCELLED))$")
public String getStatus() {
    return status;
}

public void setStatus(String status) {
    this.status = status;
}
```

- Eliminamos el Enum status y añadimos un nuevo atributo status en la clase Application con sus respectivos atributos.

Antes: Clase Gps sin atributo name.

Después:

```
@NotBlank
public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}
```

- Añadimos un nuevo atributo name en la clase Gps con sus respectivos atributos.

Antes: Sin asociación entre manager y Application.

Después: añadimos los respectivos métodos a la clase Manager y Application con respecto a la nueva relación.

Antes: Sin relación Trip con Finder y Finder con Atributo result.

Después: añadimos los respectivos métodos a la clase Trip y Finder con respecto a la nueva relación y añadimos un nuevo atributo a Finder llamado cache con sus respectivos métodos.

Antes: Ranger y Manager sin atributos que no sean heredados.

Después: Añadimos a Ranger y Manager un nuevo atributo suspicious (boolean) y sus respectivos métodos.

Antes: Trip sin atributo cancelled.

Después: Añadimos a Trip un nuevo atributo cancelled (boolean) con sus respectivos métodos.