

Pensamiento Computacional

Sistema de notificaciones.

Contexto: Un sistema que notifique a ciertos usuarios de eventos que ocurren.

- Clases:
 - NotificationSystem: Incluye el user y el mensaje.
 - User: Es una interfaz.
 - GuestUser: Implementa de User.
 - ClientUser: Implementa de User.
 - AdminUser: Implementa de User.
 - NotificationStrategy: Manda los mensajes.
 - SMSNotification: Implementa de NotificationSystem.
 - PushNotification: Implementa de NotificationSystem.
 - EmailNotification: Implementa de NotificationSystem.
 - EventListener: Actualiza a user.
 - EventManager: Suscribirse y quitar suscripción de eventos.
- Relaciones:
 - NotificationSystem contiene un NotificationStrategy
 - NotificationSystem contiene varios Users.
 - NotificationSystem contiene un EventManager

1. ¿Qué se solicita finalmente?

Un programa para enviar notificaciones dependiendo del usuario sobre eventos que ocurren.

¿Cómo se agrupa la información solicitada?

Sistema de Notificadores, Usuario, Notificación, Evento, Email, SMS, Push.

¿Qué funcionalidades se solicitan?

formatMessage(), send(), update(), subscribe(), unsubscribe().

2. Descomposición

¿Cómo se distribuyen las funcionalidades?

La clase notificador tiene el método "mandar mensajes".

La clase evento tiene "suscribir", "desuscribir" y "notificar".

Las clases PushNotification, SMSNotification y EmailNotification tienen el método "enviar"

3. Reconocimiento de patrones

Utilizamos los patrones Strategy, Template Method, Observer, Chain Of Responsibility, Command, Invoker.

4. Codificación

<https://github.com/holotwist/notification-system>

