

Universidad del Valle de Guatemala
Facultad de ingeniería
Programación Orientada a Objetos



Laboratorio Sustituto

Nelson Eduardo García Bravatti 22434

Requisitos funcionales del programa:

El usuario podrá realizar las siguientes acciones en el programa:

- agregar participantes
- crear nuevos proyectos
- asignar 3 participantes a un proyecto, se puede repetir
- ver los participantes
- ver los proyectos creados

Análisis:

Principal

Clase main, se encarga de realizar todas las acciones, llamando métodos de otras clases.

Participante

Clase participante, representa las personas que trabajan en la empresa, pueden ser de tres tipos.

Atributos:

```
private String nombre; // nombre del participante
private String tipo; //practicante, ingeniero o administrador
private boolean liderazgo; // si tiene liderazgo
private boolean colaboración; // si tiene colaboración
private boolean compañeirismo; // si tiene compañeirismo
private boolean experiencia; // si tiene experiencia
private boolean conocimiento; // si tiene conocimiento
private boolean creatividad; // si tiene creatividad
private boolean barato; // si tiene barato
private boolean profesional; // si tiene profesional
private boolean tiempo_completo; // si tiene tiempo_completo
private ArrayList<participante> participantes_p; // los participantes del proyecto
```

Métodos:

```
participante(String nombre, String tipo, boolean liderazgo, boolean colaboracion, boolean
companeirismo, boolean experiencia, boolean conocimiento, boolean creatividad, boolean
barato, boolean profesional, boolean tiempo_completo) //constructor
```

```
gets() y sets() //gets y sets de los atributos de participantes
toString() //to string
```

Empresa

Clase empresa, tiene muchos trabajadores(participantes) y realiza proyectos.

Atributos:

```
private ArrayList<participante> participantes; //participantes  
private ArrayList<proyecto> proyectos; //proyectos de la empresa
```

Métodos:

```
gets() y sets() //gets y sets de los atributos de empresa  
empresa() //constructor  
agregar_participantes() // agregar un participante a la empresa  
nuevo_proyecto() // crea un nuevo proyecto de la empresa
```

```
mostrar_participantes() muestra a los participantes  
mostrar_proyectos() //muestra los proyectos
```

Proyecto

Clase proyecto, los proyectos son administrados por la empresa, cada proyecto tiene 3 participantes, los proyectos pueden ser de diferentes tipos.

Atributos:

```
private String nombre_proyecto; // nombre del proyecto  
private String tipo; //investigativo, nuevo cliente, mantenimiento, interno  
private boolean liderazgo; // si necesita liderazgo  
private boolean colaboración; // si necesita colaboración  
private boolean companeirismo; // si necesita compañerismo  
private boolean experiencia; // si necesita experiencia  
private boolean conocimiento; // si necesita conocimiento  
private boolean creatividad; // si necesita creatividad  
private boolean barato; // si necesita pago barato  
private boolean profesional; // si necesita profesionalidad  
private boolean tiempo_completo; // si necesita tiempo completo  
private ArrayList<participante> participantes_p; // los 3 participantes del proyecto
```

Métodos:

```
proyecto() // constructor  
gets() y sets() //gets y sets de los atributos de proyecto
```

toString() //to string

Principios de diseño:

Se piensa utilizar archivos para tener listos los participantes premeditados para probarlos en el programa. La empresa tendrá una lista de proyectos y una lista de practicantes. A parte de los participantes que ya están integrados a la empresa, se puede añadir nuevos. Los proyectos se crean desde cero, se elige qué tipo son y luego de manera aleatoria se eligen los participantes.

Diseño:

