



## Descripción

El auge de las comunicaciones móviles ha generado una gran demanda de servicios por parte de las empresas de telefonía celular. De esta forma, la empresa *LlamaYa* ha inaugurado recientemente un gran Centro de Atención al Usuario (CAU) para poder atender adecuadamente a todos sus suscriptores, y también a los nuevos usuarios que se quieran vincular. Este CAU tiene un conjunto de asesores, los cuales se encargan de realizar la atención personalizada a cada usuario. El usuario a su llegada solicita un turno, y espera que sea llamado por el asesor para realizar su operación.

El CAU ofrece atención a los usuarios en 3 temas fundamentales: ventas, servicio técnico y reclamos. De esta forma, al llegar al CAU cada usuario tiene asociado un número de identificación personal (único), un indicador de prioridad (si es una persona de la tercera edad, con alguna discapacidad o está en embarazo, por ejemplo), un tema sobre el cual va a consultar, representado por dos letras (VE, ST, RE), y un número de turno o número consecutivo de espera para el tema escogido. Por otro lado, cada asesor tiene asociado un número de identificación personal (único), un estado actual (que puede ser ocupado, disponible o fuera de servicio), un tema que puede atender (cada asesor se especializa en atender sólo un tipo de solicitudes), y un registro de los turnos que ha atendido hasta el momento.

Para implementar toda esta dinámica en el nuevo CAU, la empresa *LlamaYa* requiere el apoyo de un sistema de información que se encargue de la asignación de los turnos a los usuarios y de la asignación de usuarios a los asesores para realizar la atención. Además, el sistema de información debería permitir extraer algunas estadísticas u otra información relevante sobre el proceso de atención de usuarios. De esta forma, se han identificado inicialmente tres grandes operaciones que el sistema debería ofrecer para el nuevo CAU:

1. Asignar un turno. Dado un nuevo usuario que llega (con su identificación, prioridad y tema de consulta), el sistema debe generar un número de turno para el usuario, y debe almacenar la información del usuario en una estructura adecuada que permita la posterior atención bajo el esquema: primero que llega, primero en ser atendido. Si el usuario tiene alguna prioridad, debe ubicarse en la estructura de forma que sea el próximo (primero) en ser atendido.
2. Atender un usuario. El sistema debe identificar el tema con mayor cantidad de usuarios en espera para realizar ahí la atención. De esta forma, dentro del conjunto de asesores del CAU, el sistema debe encontrar el primer asesor que esté disponible y que además pueda atender el tema identificado anteriormente. La información del usuario debe retirarse de la estructura de atención y su número de turno debe almacenarse en el registro de atendidos del asesor seleccionado.
3. Conteo de usuarios atendidos. Para identificar la cantidad de usuarios atendidos hasta el momento en el CAU, el sistema debe revisar los registros de usuarios atendidos de cada asesor del CAU.

## Diseño del sistema

Para llevar a cabo una posterior implementación del sistema de apoyo a la empresa de telefonía celular *LlamaYa*, a continuación se describe el diseño del sistema y los TADs relacionados.

### TAD Asesor

#### Datos mínimos:

- id, entero largo, número único de identificación personal del asesor.

- estado, cadena de caracteres, representa el estado actual del asesor (ocupado, disponible, fuera-servicio).
- tema, cadena de caracteres, representa el tema que puede atender el asesor (VE, ST, RE).
- atendidos, lista de enteros, registra los turnos atendidos por el asesor hasta el momento.

#### Operaciones:

- obtenerId(), retorna el número de identificación del asesor.
- obtenerEstado(), retorna el estado actual del asesor.
- obtenerTema(), retorna el tema que atiende el asesor.
- obtenerAtendidos(), retorna la lista de turnos atendidos por el asesor.
- fijarId(n\_id), modifica la identificación del asesor.
- fijarEstado(n\_estado), modifica el estado actual del asesor.
- fijarTema(n\_tema), modifica el tema que atiende el asesor.
- fijarAtendidos(n\_turnos), modifica la lista de turnos atendidos por el asesor.
- agregarAtendido(turno), agrega el turno dado a la lista de atendidos.

#### TAD Usuario

##### Datos mínimos:

- id, entero largo, número único de identificación personal del usuario.
- prioridad, booleano, indica si el usuario debe tener prioridad en la atención.
- tema, cadena de caracteres, representa el tema sobre el cual el usuario requiere atención (VE, ST, RE).
- turno, entero, representa el turno asignado por el sistema al usuario.

#### Operaciones:

- obtenerId(), retorna el número de identificación del usuario.
- obtenerPrioridad(), retorna la prioridad del usuario.
- obtenerTema(), retorna el tema de consulta del usuario.
- obtenerTurno(), retorna el turno asignado al usuario.
- fijarId(n\_id), modifica la identificación del usuario.
- fijarPrioridad(n\_prioridad), modifica la prioridad del usuario.
- fijarTema(n\_tema), modifica el tema de consulta del usuario.
- fijarTurno(n\_turno), modifica el turno asignado al usuario.

#### TAD CAU

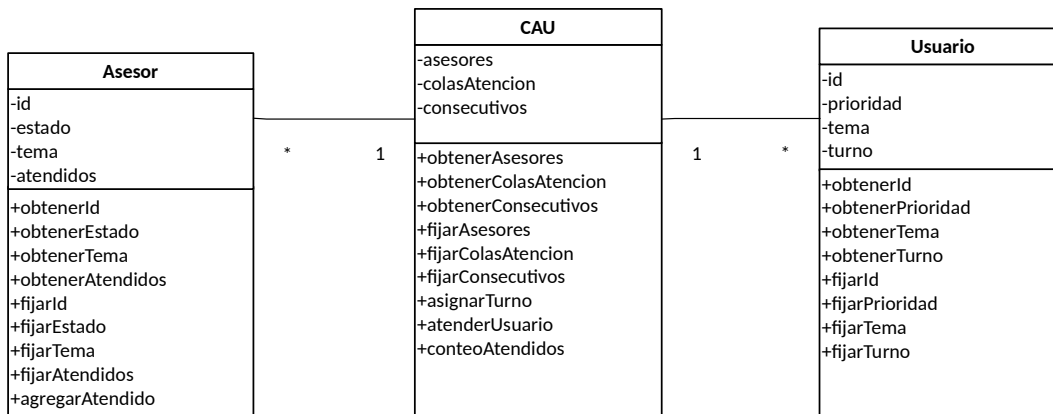
##### Datos mínimos:

- asesores, lista de Asesor, representa los asesores asignados al CAU.
- colasAtencion, vector de cola de Usuario, representa los usuarios en espera de ser atendidos por ventas, servicio técnico y reclamos.
- consecutivos, vector de entero, representa el turno consecutivo disponible para cada cola: ventas, servicio técnico y reclamos.

#### Operaciones:

- obtenerAsesores(), retorna la lista de asesores del CAU.
- obtenerColasAtencion(), retorna las colas de usuarios en espera en el CAU.
- obtenerConsecutivos(), retorna los turnos consecutivos disponibles para cada cola de usuarios.
- fijarAsesores(n\_asesores), modifica la lista de asesores del CAU.
- fijarColasAtencion(n\_colas), modifica las colas de usuarios.
- fijarConsecutivos(n\_cons), modifica los turnos consecutivos disponibles para cada cola.

- `asignarTurno(id,prioridad,tema)`, genera un usuario con la información dada, le asigna un turno y lo encola en la cola correspondiente al tema.
- `atenderUsuario()`, revisa la cola más larga, luego busca un asesor disponible que pueda atender ese tema, retira al usuario de la cola y registra el turno con el asesor.
- `conteoAtendidos()`, revisa los registros de turnos atendidos de cada asesor y genera el total de usuarios atendidos.



## Diseño e implementación de operaciones

Teniendo en cuenta el diseño presentado anteriormente, diseñe primero y luego complete con código C++ las siguientes implementaciones de los componentes ya descritos del sistema de apoyo a la empresa de telefonía celular *LlamaYa*. El diseño deberá especificar, en lenguaje natural, los pasos necesarios para completar cada una de las operaciones, teniendo en cuenta los datos y operaciones ya definidos en el (los) TAD(s). Recuerde que diseñar es un proceso previo a la implementación, por lo que no debería contener ninguna referencia a lenguajes de programación en el diseño. La implementación deberá tener en cuenta el NO uso de salidas/entradas por pantalla/teclado (i.e. paso/retorno correcto de valores y/o objetos), la coherencia y el correcto uso del diseño definido en los puntos anteriores, y la escritura de todo el código que pueda llegar a necesitar que no esté incluido en la STL.

### Diseño: Asignar un turno

```
// pasos a seguir:
//
```

### Implementación: Asignar un turno

```
string CAU::asignarTurno(int id, bool prioridad, string tema) {

}
```

### Diseño: Atender un usuario

```
// pasos a seguir:
//
```

### **Implementación: Atender un usuario**

```
bool CAU::atenderUsuario() {  
  
}
```

### **Diseño: Conteo de usuarios atendidos**

```
// pasos a seguir:  
//
```

### **Implementación: Conteo de usuarios atendidos**

```
int CAU::conteoAtendidos() {  
  
}
```